This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410

On flow control and optimized back-off in non-saturated CSMA

Andrea Baiocchi, *Member, IEEE,* and Ion Turcanu, *Member, IEEE*

***Abstract*—Medium Access Control (MAC) main functions encompass contention for channel access, packet scheduling, error control, and data integrity. Channel contention is a collective function involving all stations in the network, while data integrity pertains to data flows of each individual station. We propose a design where contention related functions are separated from other data management functions. The hinge connecting contention and other data management functions is a flow control algorithm, aiming at guaranteeing stability of contention queues and load on the MAC channel. With reference to Carrier-Sense Multiple Access (CSMA), we define an analytical model of contention queues under non saturated traffic. An asymptotic analysis of the model for large number of stations yields a closed form of the optimal flow control rate. The insight gained from the model is used to design an adaptive flow control algorithm that guarantees throughput optimality for all values of the number of stations.**

***Index Terms*—MAC protocol design, CSMA, non-saturated traffic, throughput optimality, flow control, stability.**

I. INTRODUCTION

Carrier-Sense Multiple Access (CSMA) has dominated the stage of random multiple access techniques, boosted by the impressive success of Wi-Fi networks, whose core Medium Access Control (MAC) protocol algorithm is the so called CSMA/CA. The CSMA/CA algorithm is the foundation of essentially all variants of IEEE 802.11, from high speed local wireless networks (IEEE 802.11b/a/g/n/ac, partly also IEEE 802.11ax), to vehicular (IEEE 802.11p/bd) and sensor networks (IEEE 802.11ah).

On the theoretical side, the landmark works by Tobagi and Kleinrock [1]–[4] defined basic models for performance evaluation, investigated stability issues, and gave fundamental insight into the properties of CSMA. The seminal paper of Bianchi [5] opened the way to the analysis of CSMA/CA, the variant used in Wi-Fi MAC layer Distributed Coordination Function (DCF). This model applies under saturated traffic conditions. A notable re-visitation of that work is offered in [6], still for saturated stations.

Classic CSMA design refers consistently to the basic model, where re-transmissions are strictly connected to channel contention. While clearly impacting channel usage, re-transmission is a way of managing error control. Depending on context and application targets, error control can be pursued also with coding. Moreover, error control is less related to sharing the channel among the contending stations, and more with preserving end-to-end data integrity.

A. Baiocchi is with the Department of Information Engineering, Electronics and Telecommunications (DIET), University of Rome Sapienza, Italy, e-mail: andrea.baiocchi@uniroma1.it.

I. Turcanu is with the Luxembourg Institute of Science and Technology (LIST), Luxembourg, e-mail: ion.turcanu@list.lu.

As a matter of example, flows of update messages may not need re-transmissions, and real-time traffic flows could be protected by means of coding, instead of re-transmissions. Several instances of MAC protocols mix together these functions. As a matter of example, in Wi-Fi, re-transmissions affect directly the MAC procedure by modifying the contention window size through the Binary Exponential Back-off (BEB) algorithm.

We propose a clear-cut separation between channel contention related functions – referred to as Contention Management (CM) sublayer – and other tasks performed by the MAC layer – referred to as Data Management (DM) sublayer. Contending for channel is a collective function, involving all stations belonging to the given access network, whereas other data management functions pertain to individual traffic flows between end-points of the origin and destination devices at data-link level. We argue that decoupling data management functions from medium contention function implies an increased modularity in the design of MAC, giving potential flexibility to accommodate different requirements of traffic flows without affecting the basic channel contention algorithm.

The hinge connecting the CM and DM sublayers is a flow control function that aims at steering the system towards an optimized working point by acting on the packet flow offered to the shared channel. Optimization of performance (e.g., maximization of the achievable stable throughput), is usually pursued by adjusting parameters of the MAC protocol, such as the contention window size or the channel holding time. We show that optimization of performance can be achieved by flow control, for any given set of contention-related MAC protocol parameter values.

The flow control concept, proposed to manage the load on the shared channel, allows simpler design and increased modularity with respect to MAC contention procedure. It can work with any MAC contention algorithm, not only CSMA. The only requirement on MAC contention procedure is that it feeds back to flow control a measure of channel usage and the outcome of transmission attempts (success or fail). This approach adds modularity in the design of flow control, e.g., with respect to optimized queue-based CSMA [7], which is crafted on the CSMA contention algorithms, since it adapts the contention window size.

The main contributions of this work can be summarized as follows:

- We re-design the functional architecture of the MAC level, proposing a clear-cut separation between contention for channel access and other data management functions, introducing a flow control function that regulates the packet flow from the upper DM sublayer to the lower CM sublayer and guarantees stability of CM queues.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410

- We propose an analytical model of the MAC layer which allows the evaluation of several performance metrics in the general case of non-saturated traffic sources.
- Leveraging on the insight gained from the model, we present an adaptive flow control algorithm, showing that it achieves throughput optimality, i.e., DM queues are effectively stabilized under flow control as long as the load offered to the channel is smaller than the maximum achievable stable throughput. We also discuss the relationship of optimized flow control to traditional contention window optimization.

The workflow of the paper is as follows. Related works are discussed in Section II. The proposed architectural design is introduced in Section III, including a discussion on implementation. The relevant system model is defined and analyzed in Section IV. The model is validated against simulation of a CSMA network in non-saturated conditions in Section V. Leveraging on the model of Section IV, an adaptive flow control algorithm is defined in Section VI and it is shown to provide stability for any traffic load within the maximum achievable stable throughput. Conclusions and hints to generalizations of the modeling approach are given in Section VII. Mathematical details and proofs are collected in the Appendix.

II. Related works

Existing works that address modeling CSMA/CA under non-saturated traffic conditions are either based on Markov chains [8]–[11] or following a renewal process [12]–[25]. Most of these works modify Bianchi's saturation model to account for the fact that stations are not always backlogged, by scaling the probability of transmission of a station according to the utilization coefficient of the relevant packet queue. The critical point of those models is the difficult trade-off between complexity and accuracy. Another group of works [26]–[33] addresses broadcast traffic in non-saturated conditions, often with applications to vehicular and ad-hoc networking.

Optimization of CSMA has also been considered. For example, queue-based carrier-sense multiple access (Q-CSMA) [34] is designed to ensure throughput optimality in wireless networks, where the access probability depends on the local queue length information. Maatouk et al. [35] extend adaptive CSMA targeting throughput optimality to new adaptive schemes that account also for energy saving, by allowing sleep mode of idle stations. Though introducing the idea of queue-based admission control, these works still retain the approach to adjust MAC channel sharing parameters to achieve optimal throughput.

Optimal CSMA analysis and implementation is presented in [36]. It addresses a practically implementable, IEEE 802.11-compatible version of the theory developed in [7], [34], [37]–[39] for optimal queue-based CSMA. The basic idea is that back off probability and transmission time are adapted as a function of local station queue length. Still in the framework of queue-based CSMA, Xia et al. [40] introduce virtual queues to improve fairness and delay performance. This thread of works introduces a concept similar to flow control, i.e., the MAC interface queue is fed according to a backlog dependent control [36]. Per neighbor buffers are managed and the contention window size of IEEE 802.11 is adapted as a function of the control queue backlog. Queue-based control is therefore intimately connected to MAC contention parameters.

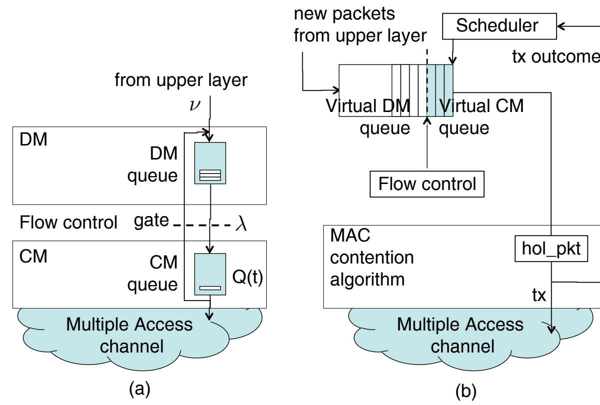


Figure 1. MAC layer architecture concept. (a) Functional split into DM and CM sublayers, connected through Flow Control. (b) Possible implementation scheme, based on virtual DM and CM queues.

Flow control is introduced in [41] for CSMA/CA-based multi-hop networks. However, the purpose is to reduce the interference between packets of a given flow that travel through the multi-hop end-to-end route. The techniques used to improve throughput include a hop-by-hop window-based flow control scheme that paces the transmission of frames such that competition between frames originating from the same flow is reduced.

In the approach proposed in this paper, back-off parameters (like transmission probability, contention window size, channel holding time) are left under the control of MAC contention algorithm. A flow control function that throttles packets to the MAC interface buffer is in charge of managing the shared channel load and of guaranteeing stability of station queues.

III. System description

The proposed functional architecture of the MAC layer is split into two sublayers (see Figure 1a): the *DM sublayer*, which is in charge of data integrity and scheduling (as well as possibly other data management functions, such as security), and the *CM sublayer*, which is responsible for sharing the communication channel with other stations. Data is moved from DM to CM sublayer through a flow control function.

Section III-A is devoted to a functional description of the DM sublayer and the flow control function. Section III-B provides a concise description of the slotted non-persistent CSMA algorithm, which is the channel sharing algorithm considered in this paper for the CM sublayer. A possible implementation of the functional split concept, based on virtual CM and DM queues, is outlined in Section III-C.

A. *DM sublayer and flow control*

Packets arrive at the station buffer at DM level from upper layer. Let us refer to this buffer as the DM queue. The head-of-line packet in the DM queue is moved to the CM level for transmission on the channel, provided it is released by flow control. The flow control algorithm acts as a gate. A countdown timer is initialized to I. When I hits 0, the gate opens up, i.e., the DM queue is inspected to check whether it has a packet ready. In that case, the head-of-line packet of the DM queue is transferred to the CM queue. If the DM queue is found empty, the opportunity is lost. Once the DM queue is checked, the timer is initialized again and it ticks away until the next timer expiry.

The time pace I can be drawn from a fixed distribution or it can be adapted so as to maintain the channel at an optimal working point (see Section VI-C). A simple implementation of the flow control countdown is obtained by dividing the time axis into time slots of duration Δ and opening up the gate in each slot with probability a. This corresponds to using a geometrically distributed timer I with mean $\overline{I} = \Delta/a$.

Packets in the CM queue are wrapped in MAC frames which are transmitted according to the MAC contention algorithm. We distinguish two types of frames: acknowledged and unacknowledged. The former type applies to unicast frames that require an ACK from the receiver. The latter type does not require any receipt confirmation (e.g., broadcast frames).

In the first case, the CM entity reports the outcome of transmission attempt to the DM entity. If an ACK is received, the packet has been delivered and no further action is required. If instead the transmission attempt fails, the packet is possibly re-scheduled into the DM queue, according to the data integrity policy managed by the DM sublayer. Maintaining packet ordering is not a concern of the CM sublayer. The CM queue adopts a First Come First Served (FCFS) policy. It accepts packets from the DM sublayer through the flow control gate and transmits them in the same order over the channel. To maintain packet ordering and ease the interaction of CM and DM sublayers, a *virtual queue* approach is proposed in Section III-C.

B. CM sublayer

Channel access is operated according to a well-known variant of CSMA, the so called slotted non-persistent CSMA [1]. When the channel is idle, the time axis is slotted in so called back-off slots. Let δ be the duration of a back-off slot. A station is said to be backlogged if its CM queue is not empty. A backlogged station senses the channel to assess whether it is busy. This operation requires a time δ. If the channel is sensed busy, the station waits until the channel goes back to idle. Upon sensing an idle back-off slot, a backlogged station takes a randomized decision whether to transmit or not. When the station eventually starts transmitting, it keeps the channel busy for a time interval θ, which accommodates overhead and payload transmission time. Additionally, the interval θ also allows time for an acknowledgement, if required.

The randomized decision can be described as follows. When taking on a packet to transmit from the CM queue, the station draws a positive, integer-valued random variable M. A counter is initialized to M and decremented each time an idle back-off slot is sensed. Transmission starts when the counter hits 0. As a matter of example, in Wi-Fi CSMA/CA, the random counter is drawn uniformly at random in the integer set $\{1, \dots, w\}$, where w is the current contention window size of the station.

Table I
MAIN NOTATIONS USED IN ALGORITHMS 1 AND 2

Symbol	Meaning
h	Head-of-line packet according to scheduler selection.
a	Binary variable: $a = 1$ means that the head-of-line packet requires an acknowledgment; otherwise, it is $a = 0$.
I	Flow control timer.
CCA	Clear Channel Assessment, a binary variable returned by the physical layer, assessing whether the channel is sensed idle or busy. The function CCA takes time δ to run.

C. Implementation

The proposed architecture decouples the lower MAC sublayer, which is responsible for channel contention and fair sharing of the communication channel, from other traffic related functions, namely data integrity (e.g., pursued by means of re-transmissions) and scheduling (e.g., priorities, support of quality of service requirements).

Flow control gates packets from the DM sublayer to the CM sublayer. As the flow control timer expires, a packet is moved from the DM queue to the CM queue. The choice of which packet in the DM queue is the eligible head-of-line is up to the DM queue manager process.

To implement this decoupling (see Figure 1b), we virtualize the CM queue, by means of a counter q_{CM}. Another counter, q_{DM}, holds the number of packets enqueued in the DM queue. Packets are physically stored in a unique buffer, whose serving order is dictated by the desired scheduling and re-transmission policies. By using counters, a packet is picked out of the physical buffer only when its contention process starts (see Algorithm 1).

The counter q_{CM} is incremented every time the flow control timer expires and the DM queue is found not empty, namely, it is $q_{\text{DM}} > 0$. It is decremented each time a packet is transmitted on the channel. Analogously, the counter q_{DM} is decremented each time a packet is moved (virtually) from the DM queue to the CM queue, thanks to a flow control grant. The counter q_{DM} is incremented each time (1) a new packet is handed over to DM by upper layer or (2) a packet must be retransmitted.

The latter event accounts for the fact that a "new" packet transmission is being required, therefore requiring a flow control grant. In other words, a packet being transmitted, for example, three times (one initial attempt and two subsequent re-transmissions) consumes three flow control grants.

The pseudo-code of an algorithm that implements the concept described above is shown in Algorithm 1. Algorithm 2 lists the code of the function `CMfun()`, called inside Algorithm 1. The variables used in the two algorithms are defined in Table I.

In Algorithm 1, upon each timer expiry, the DM queue length is checked. If positive (at least one packet waiting for a transmission opportunity), the length counters of DM and CM queues are updated, moving (virtually) one packet from the DM queue to the CM queue. The timer is re-initialized and starts running again. If no packet is standing in the DM queue, the opportunity goes unused.

As long as the virtual CM queue counter q_{CM} is positive, there are packets ready to be transmitted on the channel

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410

and authorized by flow control. The CM queue length is decremented and the packet selected by the scheduler is retrieved from the physical buffer (head-of-line packet). The head-of-line packet is described by a pointer h and an attribute a. The latter is a binary variable, equal to 1 if and only if an ACK is required for packet delivery confirmation.

The head-of-line packet is handed over to the function `CMfun()` (see Algorithm 2), which is in charge of making one transmission attempt on the channel and returning the outcome of the transmission. The function $\texttt{rand\_int}(W_0)$ is invoked to set the countdown k, by drawing an integer in the range $\{1, \dots, W_0\}$ uniformly at random. The parameter W_0 denotes the given contention window size. The transmission attempt outcome is DONE unless an ACK was required and it was not received, in which case the outcome is FAIL. Once the outcome of the transmission attempt is returned to the main procedure, the head-of-line packet is re-scheduled if it is of acknowledged type ($a = 1$), the transmission attempt has failed (outcome = FAIL), and a new transmission attempt is allowed (the number of already used re-transmission attempts is less than the maximum allowed). The DM queue length is incremented by one, since the re-transmission attempt creates a new packet transmission request. In any other case (no ACK required, no more re-transmission attempts left, or successful transmission), the transmitted packet pointed by h is cleared off the buffer, a new head-of-line packet is selected and is immediately processed by the CM function, as long as $q_{\text{CM}} > 0$.

Stability of CM queues is guaranteed by flow control. A discussion of the stability of DM queues is given in Section IV-C.

Algorithm 1 DM and CM virtual queues handling.

```
Initialization:
  q_CM ← 0
  q_DM ← 0
  I ← set_timer( )
Upon packet arrival at DM queue:
  q_DM ← q_DM + 1
Upon timer expiry:
  if q_DM > 0 then
      q_DM ← q_DM − 1
      q_CM ← q_CM + 1
  end if
  I ← set_timer( )
CM de-queue as long as q_CM > 0:
  q_CM ← q_CM − 1
  [h, a] ← get_new_pkt_from_scheduler( )
  outcome ← CMfun(h,a)
  if (outcome == FAIL) and (retx_required) then
      q_DM ← q_DM + 1
  else
      clear_pkt(h)
  end if
```

Algorithm 2 Function `CMfun` called in Algorithm 1.

```
def CMfun(p,a):
  k ← rand_int(W_0)
  while k > 0 do
      CCA ← check_channel(δ)
      if CCA == idle then
          k ← k − 1
      end if
  end while
  sendto_PHY_and_wait_for_eotx(p)
  if a == 0 then
      outcome ← DONE
  else
      ACKed ← wait_for_ack(T_ACK)
      if ACKed then
          outcome ← DONE
      else
          outcome ← FAIL
      end if
  end if
  return outcome
```

IV. System model

In this section we elaborate an analytical model of the MAC layer, consistent with the functional architecture outlined in Section III.

We consider a set of n stations sharing a broadcast communication channel. Stations can sense each other's transmissions, i.e., there are no hidden stations. Hence stations are "synchronized" and the time axis can be thought as split into *virtual slots*. A virtual slot consists of one back-off slot and possibly MAC packet transmission by one or more stations. We assume that reception fails, if more than one station transmits simultaneously, otherwise it is successful. The transmission time of a MAC packet includes any overhead (e.g., preamble, header and trailer, inter-frame spaces). We focus on the homogeneous case where the transmission time is the same for all stations. Notation is simpler and we maintain the full potential for insight on the interplay between flow control and random multiple access.

In the following we assume that the flow control timer is realized as a negative exponential random variable with mean $1/\lambda$, so that the input to the CM queue is a Poisson process with mean rate λ.[1] We will see in Section IV-C that this seemingly restrictive assumption does not impair throughput optimality.

A summary of main notations is given in Table II. A variable that refers to station i is denoted with superscript i, e.g., $x^{(i)}$. The superscript is dropped when there is no ambiguity, for the sake of simple notation. Time-dependent variables are denoted with an argument, e.g., $x(t)$.

The models of the CM and DM queues are developed separately. Section IV-A is devoted to the analysis of the CM queue, while the model of the DM queue is derived in Section IV-B. An in-depth discussion of stability of the DM queue and its relationship to flow control is provided in Section IV-C. Details of mathematical derivations are given in Appendix A.

A. Analysis of the CM queue

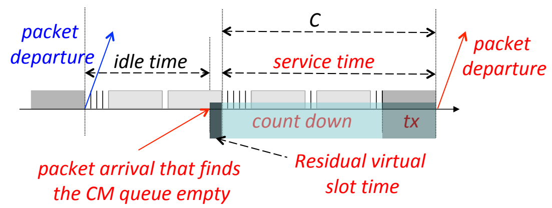
The model is developed from the point of view of a tagged station, all others being accounted for by means of their

[1] The actual input process to the CM queue is a Poisson process with "holes", corresponding to flow control opportunities not used, because the DM queue was empty. More in depth, as long as the DM queue is in a busy period, the input to the underlying CM queue is effectively a Poisson process. No input is realized instead when the DM queue is idle. Assuming the input to the CM queue is a stationary Poisson process with mean rate λ leads therefore to a bound on performance (e.g., an upper bound of the mean delay through the CM queue). The bound is closer to the actual performance the higher the traffic load on the DM queue, i.e., the smaller the probability that the DM queue is found empty.

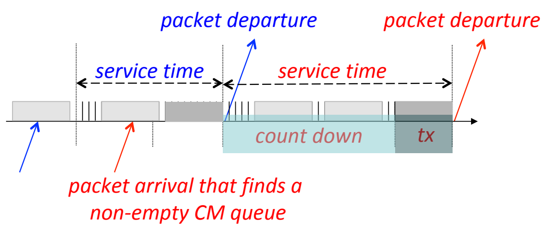
This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410

Table II
MAIN NOTATIONS USED IN THE ANALYTICAL MODEL

Symbol	Meaning
δ	Back-off slot time.
θ	Transmission time of a MAC frame.
τ	Probability of transmitting in a virtual slot time.
q	Probability that the tagged station sees an idle channel.
λ	Mean arrival rate of packets at the CM queue.
ν	Mean arrival rate of packets at the DM queue.



(a) Packet arrival that starts a new busy period



(b) Packet arrival during a busy period

Figure 2. Time evolution of the MAC channel as seen by the tagged station: (a) the residual virtual slot time, before countdown starts, is highlighted by a dark stripe; (b) no residual virtual slot time occurs. The service time C the countdown (during which other stations can transmit: light grey rectangles) and the transmission of the packet (dark grey rectangle).

probability of transmitting in a virtual slot time.[2] We define an Embedded Markov Chain (EMC) $Q_k = Q(t_k^+)$, where $Q(t^+)$ denotes the number of packets found in the CM queue immediately after time t, and t_k denotes the k-th packet departing time, $k \geq 0$. Therefore, Q_k is the CM queue length immediately after time t_k, i.e., just after the transmission of k-th packet.

Packets arrive at the CM queue according to a Poisson process with mean rate λ, thanks to the regulation of the flow control algorithm. A packet departs from the CM queue only at the end of a virtual slot time, the one where it is transmitted. If an arriving packet is the first one of a new busy period of the CM queue, it has to wait the end of the current virtual slot time. Only at that point can the countdown start. The time it takes to complete the MAC countdown plus the ensuing packet transmission time will be referred to as *service time*.

A timing scheme of the MAC channel as seen by the tagged station is shown in Figure 2. A packet arrival, triggering the start of a new busy period, is shown in Figure 2a. The time evolution of a packet transmitted within a busy period is shown in Figure 2b. Light grey rectangles represent transmissions of stations other than the tagged one, during the countdown of the tagged station. Dark grey rectangles stand for transmission times involving the tagged station.

The state of tagged station's CM queue at embedded times evolves according to the following recursion.

$$Q_{k+1} = \begin{cases} Q_k - 1 + A_{k+1} & Q_k > 0, \\ B_{k+1} - 1 + A_{k+1} & Q_k = 0 \end{cases} \tag{1}$$

where A_k is the number of arrivals during the k-th service time, and B_k is the number of arrivals in the residual virtual slot time at the beginning of a busy period. Service time is the time required to count down the back-off counter M and finally transmit the head-of-line packet, that is to say:

$$C = \begin{cases} \delta + \theta & \text{if } M = 1, \\ \sum_{j=1}^{M-1} X_j + \delta + \theta & \text{if } M > 1. \end{cases} \tag{2}$$

where X_j is the duration of the j-th virtual slot time during the countdown of the tagged station, and M is a positive discrete random variable, representing the initial value of the back-off counter of the tagged station.

Let q denote the probability that no other station transmits in a virtual slot time and hence the tagged station senses an idle channel. We can write

$$X = \begin{cases} \delta & \text{with probability } q, \\ \delta + \theta & \text{otherwise.} \end{cases} \tag{3}$$

The analysis developed in Appendix A yields the generating function of the steady state Probability Density Function (PDF) of the EMC Q_k, which exists if and only if $\lambda \mathrm{E}[C] < 1$. The first moment is given by

$$\mathrm{E}[Q] = \frac{\lambda \mathrm{E}[X^2]}{2\mathrm{E}[X]} + \frac{1}{2} + \frac{\lambda^2 \mathrm{E}[C^2] + \lambda \mathrm{E}[C]}{2\,(1 - \lambda \mathrm{E}[C])} + \lambda \mathrm{E}[C] \tag{4}$$

Expressions of the moments of X and C are given in Appendix A.

The probability π_0 that a departing packet leaves an empty queue behind is

$$\pi_0 = \frac{(1 - \lambda \mathrm{E}[C])(1 - \varphi_X(\lambda))}{\lambda \mathrm{E}[X]}, \tag{5}$$

where $\varphi_X(s) = \mathrm{E}[e^{-sX}]$ is the Laplace Trasform (LT) of the PDF of X.

Let τ be the steady-state probability that a station transmits in a virtual slot time. At equilibrium, the probability τ is the long-run average fraction of virtual time slots where a station transmits. It is proved in Appendix A that

$$\tau = \frac{1}{\mathrm{E}[M] + \frac{\pi_0}{1 - \varphi_X(\lambda)}} = \frac{\mathrm{E}[X]}{\frac{1}{\lambda} - q\theta}. \tag{6}$$

As the offered load $\lambda \mathrm{E}[C]$ tends to 1, the queue saturates, hence $\pi_0 \to 0$ and $\tau \to 1/\mathrm{E}[M] = \tau_{\text{sat}}$.

From the definition in Equation (3), it follows that $\mathrm{E}[X] = \delta + \theta - \theta q$. Hence, Equation (6) becomes:

$$\tau = \frac{\beta + 1 - q}{\frac{1}{\lambda\theta} - q} \tag{7}$$

where $\beta = \delta/\theta$.

[2]The approach is reminiscent of mean field approximation. The effect of all stations but the tagged station, say station i, is summarized in the probability $q^{(i)}$ that station i senses an idle channel, i.e., it sees no other station transmitting in a virtual slot time.

This is the master equation, yielding the transmission probability τ of the tagged station as a function of its mean arrival rate λ and of the probability q. This last probability in turn is a function of the transmission probabilities of other stations. Hence, writing the master equation for each station, we end up with a non-linear equation system, whose fixed point yields the values of the transmission probabilities of all stations, as a function of their respective mean arrival rates, provided CM queues are stable.

Let $\lambda^{(i)}$ and $\tau^{(i)}$ be the mean arrival rate of packets at the CM queue and the transmission probability of the i-th station, $i = 1, \dots, n$. Let also $q^{(i)} = \prod_{k=1, k\neq i}^{n} (1 - \tau^{(k)})$ be the probability that station i sees an idle back-off slot time, for $i = 1, \dots, n$. The product form is an approximation consistent with the kind of mean field approximation used in the analysis of CM queues. As shown in Section V, it leads to a highly accurate analytical solution when compared with simulations.

The non-linear equation system that yields the vector $\tau = [\tau^{(1)}, \dots, \tau^{(n)}]$ as a function of the mean arrival rate vector $\lambda = [\lambda^{(1)}, \dots, \lambda^{(n)}]$ can be written as

$$\tau = \mathbf{\Phi}(\tau). \tag{8}$$

Expliciting the individual equations, we get

$$\tau^{(i)} = \Phi_i(\tau^{(1)}, \dots, \tau^{(n)}) = \frac{\beta + 1 - q^{(i)}}{\frac{1}{\lambda^{(i)}\theta} - q^{(i)}}, \tag{9}$$

where $\beta = \delta/\theta$, and Φ_i is the i-th component of the vector $\mathbf{\Phi}$. It defines a continuous and differentiable mapping on the convex and compact subset $[0, 1]^n$ of $\mathbb{R}^n$.

The following theorem states that a solution of the system in Equation (8) exists and it is unique under mild conditions.

Theorem 1. *Let $\mathbf{\Phi} : [0, 1]^n \mapsto [0, 1]^n$ be the continuous and differentiable mapping defined by Equation* (9)*. Assume that the following conditions hold: (i) $\lambda^{(i)}\theta < \frac{1}{\beta+1}$ for all $i = 1, \dots, n$; (ii) $\sum_{i=1}^{n} \lambda^{(i)}\theta q^{(i)} < 1$. Then a solution of the equation system in Equation* (8) *exists in the interior of* $[0, 1]^n$ *and it is unique.*

Remark. The condition (i) forbids that the solution $\tau^{(i)}, i = 1, \dots, n$ may belong to the boundary of the region $[0, 1]^n$, i.e., that any of the transmission probabilities might be 1 ($\tau^{(i)} = 0$ is not a solution of the system, since $\beta + 1 - q^{(i)} \geq \beta > 0,\ \forall i$). More in depth, from Equation (7) it follows that $\tau^{(i)} < 1$ implies $\beta + 1 - q^{(i)} < 1/(\lambda^{(i)}\theta) - q^{(i)}$, hence $\lambda^{(i)}\theta < 1/(\beta + 1)$, which is just condition (i). Also the converse holds, i.e., condition (i) implies that it must be $\tau^{(i)} < 1$. The intuitive meaning of condition (i) is that the normalized load $\lambda^{(i)}\theta$ of CM queue i is *strictly less* than the theoretical maximum $1/(\beta+1)$, achievable in case only station i is part of the CSMA network (hence $n = 1$) and it transmits in every virtual slot time (hence $\tau^{(i)} = 1$). The condition (ii) means that the overall normalized throughput (coefficient of utilization) achieved by the set of n stations cannot attain 100 %.

Proof. See Appendix B. □

In the setting considered in this paper, flow control parameters are actionable, while MAC parameters are given. The transmission probability τ is therefore capped by the rules of CSMA. Specifically, it must be $\tau \leq \tau_{\text{sat}} = 1/\mathrm{E}[M]$. Let $\overline{\tau}(\lambda) \in [0, 1]^n$ denote the unique fixed point of Theorem 1, corresponding to given packet arrival rates $\lambda = [\lambda^{(1)}, \dots, \lambda^{(n)}]$. We distinguish two cases: (i) $\overline{\tau}(\lambda) \in [0, \tau_{\text{sat}}]^n$; (ii) $\overline{\tau}(\lambda) \in [0, 1]^n \setminus [0, \tau_{\text{sat}}]^n$. In the first case a feasible equilibrium solution exists, hence the vector λ leads to a stable system. In the second case, there is no feasible equilibrium point for the transmission probabilities under the given arrival rates λ. We can therefore define the following stability region for the arrival rates:

$$\Lambda = \{\lambda \in \mathbb{R}^n \mid \lambda \geq 0 \text{ and } \overline{\tau}(\lambda) \in [0, \tau_{\text{sat}}]^n\}, \tag{10}$$

where inequalities on vectors are meant to be entrywise.

In the rest of the paper we refer to the homogeneous case where mean arrival rates at DM and CM queues are same for all stations. In fact, we can derive more insightful relationships in case of homogeneous stations, having same mean arrival rates and hence same transmission probabilities, i.e., $\tau^{(i)} = \tau,\ \forall i$. In that case the probability q can be expressed as

$$q = (1 - \tau)^{n-1}. \tag{11}$$

Inverting Equation (7) we express λ as a function of τ:

$$\lambda = \frac{\tau}{\delta + \theta[1 - (1 - \tau)q]} = \frac{\tau}{\delta + \theta[1 - (1 - \tau)^n]} \tag{12}$$

We find out that the renewal reward approximation for τ in Equation (7) consists of balancing the CSMA throughput of the tagged station[3] to the mean arrival rate at the CM queue of the tagged station. This is nothing but the necessary steady-state input-output equilibrium of the CM queue.

Since the right-hand side of Equation (12) is monotonously increasing with τ, the upper bound of λ for stability is:

$$\lambda_{\text{sup}} = \frac{\tau_{\text{sat}}}{\delta + \theta[1 - (1 - \tau_{\text{sat}})^n]}. \tag{13}$$

It can be checked that $\tau \to 0$ as $\lambda \to 0$ (light traffic regime), while $\tau \to \tau_{\text{sat}}$ for $\lambda \to \lambda_{\text{sup}}$ (heavy traffic regime). It can be checked as well that $\lambda\mathrm{E}[C] \to 1$ as $\lambda \to \lambda_{\text{sup}}$, so that the mean length of the CM queue diverges (saturation regime).

Using the CM queue model, we calculate the following performance metrics:

- P_s : Probability of successful packet transmission;
- ρ : Channel Busy Ratio (CBR), the mean fraction of time that a station senses the channel busy (including times when the station itself transmits);
- D_{CM} : Delay through the CM queue, defined as the time elapsing since a packet joins the CM queue until its transmission is completed.

The probability of delivering a packet (successful transmission), given that the station transmits, is

$$P_s = q = (1 - \tau)^{n-1}. \tag{14}$$

The CBR is evaluated as the ratio of the time spent to transmit a packet to the sum of that time and the duration of the idle back-off slot times preceding the packet transmission. The mean number of idle back-off slot times preceding packet

[3]In the setting considered in this paper, a packet leaves the CM queue irrespective of whether its transmission was successful or not.

transmission is denoted with $\overline{N}_I$ and is expressed as $\overline{N}_I = 1/[1-(1-\tau)^n]$. Then, the CBR can be expressed as follows:

$$\rho = \frac{\theta}{\delta \overline{N}_I + \theta} = \frac{\theta[1-(1-\tau)^n]}{\delta + \theta[1-(1-\tau)^n]} = \lambda\theta \frac{1-(1-\tau)^n}{\tau}. \tag{15}$$

As for queueing delay D_{CM}, by Little's law, we have:

$$\text{E}[D_{\text{CM}}] = \frac{\text{E}[Q]}{\lambda}, \tag{16}$$

where $\text{E}[Q]$ is given in Equation (4).

B. Analysis of the DM queue

The input to the DM queue consists of packets arriving from upper layer. Service of the DM queue is triggered by flow control. When the flow control timer expires, a packet is taken from the DM queue and copied to the CM queue[4], where it is eventually transmitted on the channel. If reliable data transfer is required, a feedback from the CM queue is expected, to signal that the packet has been acknowledged. In case the transmission attempt fails (ACK timeout), the packet is re-inserted into the DM queue.

Let us focus on loss-sensitive packet flows. Each packet requires an ACK and it is retransmitted in case of failure, until it is eventually delivered with success. We neglect the event of packet dropping due to finite number of re-transmission attempts, since the probability of such an event is usually very small (e.g., Wi-Fi allows up to 7 re-transmissions; even with a failure probability of 0.5, the probability of having to drop a packet is $0.5^8 \approx 0.004$).[5]

We assume the flow control timer is set to a random value, drawn from a negative exponential PDF with mean $1/\lambda$. A packet is serviced out of the DM queue when it is eventually delivered with success, which occurs with probability P_s. Hence, we approximate the service process out of DM queue as a Poisson process with mean rate λ, thinned with the success probability P_s. According to this approximation, the service time is distributed according to a negative exponential PDF with mean $1/(\lambda P_s)$ and the DM queue for acknowledged traffic is modeled as a $G/M/1$ queue.

Let Q_k denote the number of packets in the DM queue immediately after an arrival. Then

$$Q_{k+1} = Q_k + 1 - D_k, \tag{17}$$

where D_k is the number of departures in an inter-arrival time T. By the theory of $G/M/1$ queue (e.g., see [42, Ch. 6]), we get the following PDF for the steady state number of customers Q random variable:

$$\mathcal{P}(Q = h) = (1-\xi)\xi^h, \quad h \geq 0. \tag{18}$$

The parameter ξ is the unique root in $[0,1]$ of the equation $z = \varphi_T(\lambda P_s - \lambda P_s z)$, where $\varphi_T(s)$ is the LT of the PDF of the inter-arrival time T.

The Cumulative Distribution Function (CDF) of the waiting time is expressed as follows:

$$F_W(t) = 1 - \xi e^{-\lambda P_s (1-\xi) t}, \quad t \geq 0. \tag{19}$$

The mean of the system time D_{DM} through the DM queue is given by

$$\text{E}[D_{\text{DM}}] = \frac{1}{\lambda P_s} + \text{E}[W] = \frac{1}{\lambda P_s (1-\xi)}. \tag{20}$$

C. Stability and flow control

For the DM queue to be stable, it must be $\nu \equiv 1/\text{E}[T] < \lambda P_s$. Using Equations (12) and (14), the stability condition is written as follows:

$$\nu < \frac{\tau q}{\delta + \theta - \theta(1-\tau)q} = \frac{\tau(1-\tau)^{n-1}}{\delta + \theta - \theta(1-\tau)^n}, \tag{21}$$

where the right-hand side is the usual expression of the saturation throughput of CSMA, holding in case of a homogeneous system. Note that the right hand side of Equation (21) is a function of λ, i.e., the mean rate of flow control grants. Given λ, τ is determined according to Equation (7).

Let us define

$$\nu_{\text{sup}} = \max_{0 \leq \tau \leq \tau_{\text{sat}}} \frac{\tau(1-\tau)^{n-1}}{\delta + \theta - \theta(1-\tau)^n}, \tag{22}$$

This is the upper limit of the mean packet arrival rate ν for which it is guaranteed that the DM queue can be stabilized for a suitable value of λ, as shown in the following theorem.

Theorem 2 (Throughput optimality of flow control)**.** *For any assigned* $\nu \in (0, \nu_{sup})$*, there exist* λ_{min} *and* λ_{max}*, with* $0 < \lambda_{min} < \lambda_{max} \leq \lambda_{sup}$*, where* λ_{sup} *is given in Equation* (13)*, for which the DM queue is stabilized provided that* $\lambda \in (\lambda_{min}, \lambda_{max})$*.*

Proof. See Appendix C. □

Intuitively, too low values of λ fail to provide enough transmission opportunities. On the contrary, too high values of λ induce a high congestion level on the shared channel, thus reducing the probability of success P_s, eventually impairing stability.

Let us dig this point further, to understand how flow control stabilizes the DM queue, as long as $\nu \leq \nu_{\text{sup}}$. Assume no flow control is in place and the packet flow at rate ν is directly fed to the channel. Assuming stability is achieved and taking re-transmissions into account, it must hold that

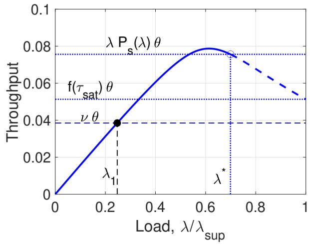
$$\nu = f(\tau) = \frac{\tau(1-\tau)^{n-1}}{\delta + \theta - \theta(1-\tau)^n}, \quad \tau \in (0, \tau_{\text{sat}}). \tag{23}$$

We consider the intersections of the curve $y = f(\tau)$ and the horizontal line $y = \nu$, as a function of the transmission probability $\tau \in [0, \tau_{\text{sat}}]$.

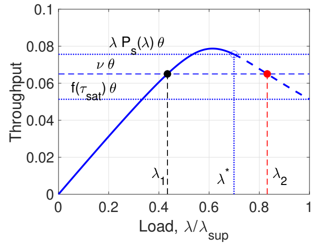
[4]Virtually copied into the CM queue, if the implementation approach outlined in Section III-C is followed.

[5]Limiting the maximum number of re-transmission does not impact the CM queue, since a packet is removed out of the CM queue after being transmitted irrespective of the outcome of its transmission attempt. The number of allowed re-transmissions does, however, impact the stability of DM queues, since it cuts a fraction of submitted packets, in case they fail all transmission attempts. Given the maximum allowed number of re-transmissions m (the parameter `max_retry` of WiFi), the stability condition of the DM queue becomes $\nu - \nu(1-P_s)^{m+1} < \lambda P_s$. If $m \to \infty$, we recover the stability condition analyzed in Section IV-B. In case no re-transmission is allowed ($m = 0$), the stability condition becomes $\nu < \lambda$, which is consistent with the fact that a packet is removed out of both DM and CM queues once they are transmitted, irrespective of the outcome of its transmission attempt.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410



(a) $\nu < f(\tau_{\text{sat}})$



(b) $\nu > f(\tau_{\text{sat}})$

Figure 3. Example sketch of stability induced by flow control. The solid/dashed line curve corresponds to the function $f(\tau)$ in Equation (23) with $n = 10$ and $\theta/\delta = 34$. (a) In this case it is $\nu < f(\tau_{\text{sat}})$, hence there exists only one intercept of the horizontal line at level ν with the curve $f(\tau)$. (b) In this case it is $\nu > f(\tau_{\text{sat}})$, hence there exist two intercepts of the horizontal line at level ν with the curve $f(\tau)$.

If $\nu < f(\tau_{\text{sat}})$, there is a single intersection point at τ_1, corresponding to a stable equilibrium point (see Figure 3a). The drain of the queue is larger than ν for every $\tau > \tau_1$ and less than ν in the opposite case. Hence, τ_1 is a stable equilibrium point.

If instead it is $\nu > f(\tau_{\text{sat}})$, the horizontal line $y = \nu$ intersects the curve $y = f(\tau)$ in two points, $\tau_1 < \tau_2$ (see Figure 3b). The smaller intercept (black marker) is a locally stable equilibrium point, but the bigger one (red marker) is an unstable point of equilibrium. If the queue fluctuations lead the system beyond the latter point, the average drift becomes positive and the queue backlog tends to increase. Eventually, the queue is driven into saturation and delay diverges.

Flow control forces the mean rate of packets leaking to the channel not to exceed λ, for $0 < \lambda < \lambda_{\text{sup}}$. This corresponds to reducing the effective sweeping interval of the transmission probability from $(0, \tau_{\text{sat}})$ to $(0, \tau(\lambda))$, where $\tau(\lambda)$ is the unique solution of Equation (12).

Let then $P_s(\lambda) = [1 - \tau(\lambda)]^{n-1}$. Even for $\nu > f(\tau_{\text{sat}})$, provided it is $\nu < \lambda P_s(\lambda)$, the DM queue remains stable, since the input packet rate to the CM queue cannot exceed λ. Only the locally stable intersection point τ_1 is attainable, while the unstable point τ_2 can never be attained, thanks to flow control. It is as if flow control "cuts" the curve $f(\tau)$ up to the abscissa $\tau(\lambda)$, leaving out the right-tail of the curve, and the unstable equilibrium point located there.

Figure 3 sketches an example of this situation. The solid line curve represents the achievable part of $f(\tau)$, while the part forbidden by flow control is shown as a dashed line curve. The horizontal dashed line marks the level of the input arrival rate ν (normalized by multiplying it by the transmission time θ). The empty circle marker shows the location of the achievable throughput under flow control for a value of λ chosen so that it is $\nu < \lambda P_s(\lambda)$. The two intersection points of the horizontal line at level ν with the curve $f(\tau)$ are marked by filled circles. The one on the left (black one) corresponds to a stable equilibrium point (the only achievable one, under flow control). The one on the right of the plot (red one), is the unstable equilibrium point, that would fatally attract the queue in the long run, without flow control.

Table III
NUMERICAL VALUES OF PARAMETERS USED FOR MODEL VALIDATION

Definition	**Value**	**Definition**	**Value**
Back-off slot time	9 µs	SIFS	16 µs
PLCP preamble	20 µs	AIFS (BE class)	43 µs
PLCP header	16 µs	ACK length	14 Byte
MAC header length	34 Byte	Basic bit rate	6 Mbit/s
MAC payload length	1500 Byte	Air bit rate	65 Mbit/s

V. Model validation

A simulation model of n CM queues interacting through the CSMA channel has been implemented in MATLAB. Physical layer parameters have been adjusted to represent the baseline functionality of IEEE 802.11ac DCF. The back-off random variable M is uniformly sampled over the integer set $\{1, 2, \dots, W_0\}$. Numerical values of parameters are listed in Table III.

The simulation model accounts for all details of the access protocol, including post back-off and immediate transmission.

Post-back-off: When a packet is transmitted and the station has an empty buffer, before going idle, it draws a back-off value and starts countdown. If no new packet arrives before the countdown expires, the station goes definitely back to idle state, waiting for new packets. Otherwise, the newly arrived packet hijacks the on-going countdown.

Immediate transmission: When a station is idle and a new packet is generated, the station checks if the channel is idle for an AIFS time. If that is the case, the frame containing the packet is transmitted immediately, without any countdown. If instead the channel becomes busy before the AIFS time is completed, the station falls back to the usual access procedure.

Simulations were run for a fixed packet size with channel holding time equal to $\theta = 34 \cdot \delta \approx 0.31$ ms (corresponding to a MAC payload length of 1500 Byte transmitted at air bit rate of 65 Mbit/s). If a collision event occurs, reception is assumed to fail with probability 1. To be consistent with the CM sublayer definition, re-transmissions are disabled.

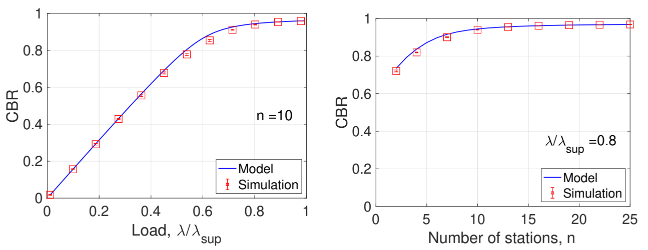
Performance results are plotted as a function of:

- the normalized packet arrival rate $\hat{\lambda} = \lambda/\lambda_{\text{sup}}$ for a fixed number of stations $n = 10$;
- the number of stations n, for a fixed ratio $\lambda/\lambda_{\text{sup}} = 0.8$,

where λ_{sup} is the maximum arrival rate that can be sustained by the CM queue (see Equation (13)). In this numerical example it is $\tau_{\text{sat}} = 2/(W_0 + 1) \approx 0.1176$, hence $1/\lambda_{\text{sup}} \approx 1.9$ ms. Simulations are displayed as square markers, along with the 95-level confidence intervals.

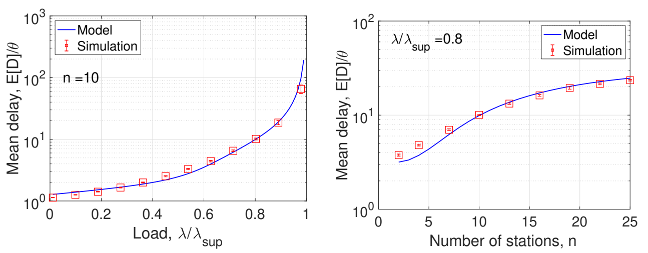
Figure 4 displays the CBR ρ as a function of the normalized packet arrival rate $\hat{\lambda}$ (Figure 4a) and of the number of stations n (Figure 4b). ρ grows almost linearly for small to moderate values of $\lambda/\lambda_{\text{sup}}$, and saturates for large values of the mean arrival rate. ρ grows with n, hovering on quite large values given the relatively high load level considered in Figure 4b. The agreement between model predictions and simulations is excellent. This is confirmed consistently by all subsequent performance metrics, as well as by many other comparison results, not shown here for space reasons.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410



(a) (b)

Figure 4. CBR for $\theta = 34 \cdot \delta$: (a) as a function of the normalized arrival rate $\lambda/\lambda_{\text{sup}}$, for $n = 10$ stations; (b) as a function of the number of stations for $\lambda/\lambda_{\text{sup}} = 0.8$.



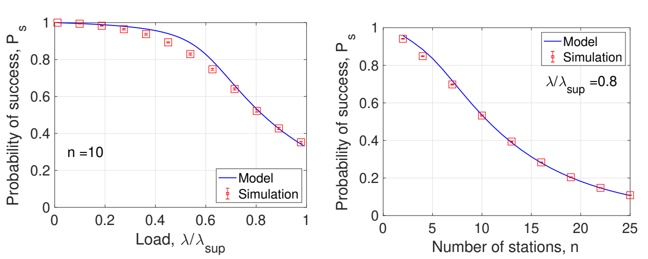
(a) (b)

Figure 5. Average CM queue delay normalized with respect to θ, for $\theta = 34 \cdot \delta$: (a) as a function of the normalized arrival rate $\lambda/\lambda_{\text{sup}}$, for $n = 10$ stations; (b) as a function of the number of stations for $\lambda/\lambda_{\text{sup}} = 0.8$.

The mean delay of the CM queue, $\mathrm{E}[D]$, is plotted in Figure 5. The behavior of the plot in Figure 5a is the typical delay-versus-throughput curve, slowly ramping up for light load levels and sharply increasing as the load gets close to saturation level. Also the curve in Figure 5b is monotonously growing with n, although with a concave shape that tends to saturate for large n.

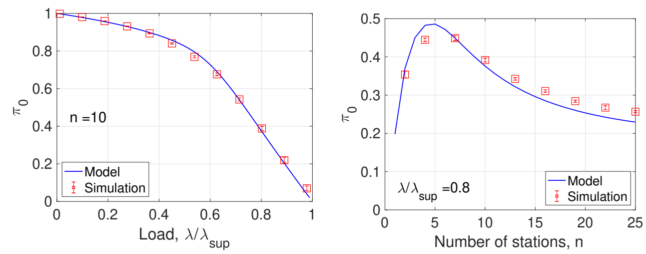
The probability of success P_s is plotted in Figure 6. From Figure 6a it appears that the probability of success is close to 1 at light load levels, dropping quickly to its value in saturation when the mean packet arrival rate λ tends to its upper limit. A similar pattern is found when P_s is plotted as a function of the number of stations in Figure 6b, except that it is never quite close to 1, even for low values of n, given the high considered load. Moreover, P_s decreases towards 0 as n grows.

The probability of having the CM queue empty at the end of a virtual slot time π_0 is plotted against $\hat{\lambda}$ in Figure 7a and as a



(a) (b)

Figure 6. Probability of success P_s for $\theta = 34 \cdot \delta$: (a) as a function of the normalized arrival rate $\lambda/\lambda_{\text{sup}}$, for $n = 10$ stations; (b) as a function of the number of stations for $\lambda/\lambda_{\text{sup}} = 0.8$.



(a) (b)

Figure 7. Probability of empty queue π_0 for $\theta = 34 \cdot \delta$: (a) as a function of the normalized arrival rate $\lambda/\lambda_{\text{sup}}$, for $n = 10$ stations; (b) as a function of the number of stations for $\lambda/\lambda_{\text{sup}} = 0.8$.

function of n in Figure 7b. In the former plot, as expected, π_0 decreases from 1 down to 0 as the load increases from 0 up to the saturation point. The decrease exhibits two rates. For low to moderate loads a slow decay dominates the behavior of the probability of empty queue. After a breakpoint around a load level of 60 %, the slope of the π_0 curve becomes definitely steeper and the curve falls rapidly towards 0.

In the plot of π_0 versus n a decreasing trend is noted after an initial growth. This is the outcome of two contrasting "forces". Requiring that $\lambda/\lambda_{\text{sup}} = a$, for a given constant a, implies that $\lambda = a\lambda_{\text{sup}} = \frac{a\tau_{\text{sat}}}{\delta+\theta-\theta(1-\tau_{\text{sat}})^n}$, which is a decreasing function of n. When n grows, but it is still at low values, the decrease of λ entails an increase of the probability of finding an empty CM queue.

As n grows further, the impact of collisions is stronger, hence congestion sets in, even if λ decreases. For large n values, π_0 is attracted to its asymptotic value as $n \to \infty$.

More in depth, it is easy to check that the following limits hold asymptotically as $n \to \infty$: $\tau \to a\tau_{\text{sat}}$, $\lambda \to a\tau_{\text{sat}}/(\delta+\theta)$, $q \to 0$, $\mathrm{E}[X] \to \delta+\theta$, $\mathrm{E}[C] = \mathrm{E}[X]/\tau_{\text{sat}} + q\theta \to (\delta+\theta)/\tau_{\text{sat}}$, $\varphi_X(\lambda) \to e^{-a\tau_{\text{sat}}}$. Hence,

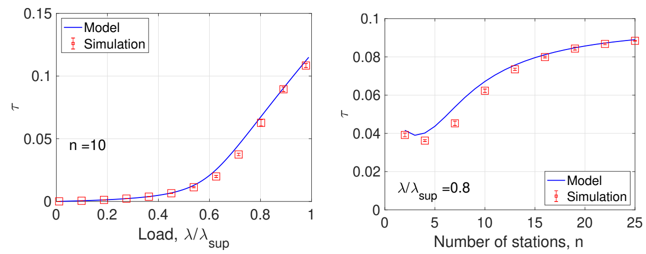
$$\pi_0 = (1-\lambda\mathrm{E}[C])\,\frac{1-\varphi_X(\lambda)}{\lambda\mathrm{E}[X]} \to (1-a)\frac{1-e^{-a\tau_{\text{sat}}}}{a\tau_{\text{sat}}} \quad (24)$$

The asymptotic value of π_0 for large n is always smaller than the value of π_0 for $n = 1$. It is easy to check that $\pi_0|_{n=1} = (1-a)\frac{1-e^{-\lambda\delta}}{\lambda\delta}$, with $\lambda\delta = a\tau_{\text{sat}}\frac{\delta}{\delta+\tau_{\text{sat}}\theta}$. Since $\frac{1-e^{-y}}{y}$ is a monotonously decreasing function of y, it follows that $\lim_{n\to\infty}\pi_0 < \pi_0|_{n=1}$. With the assumed numerical values of $a = 0.8$ and $\tau_{\text{sat}} \approx 0.1176$, we have $\pi_0 \to 0.191$ as $n \to \infty$ and $\pi_0|_{n=1} = 0.198$.

The probability of transmission τ is plotted against $\hat{\lambda}$ in Figure 8a and against n in Figure 8b. A two-regime behavior is evident also in this curve, as in the case of the probability of empty queue. For low to moderate load levels, the probability of transmission grows slowly. Once the load exceeds a critical region, the growth of the probability of transmission accelerates. As the load level tends to the saturation point the transmission probability tends to its saturation value $\tau_{\text{sat}} = 2/(W_0+1) \approx 0.1176$. A similar behavior is found for τ as a function of n, except that in this case $\tau \to 0.8 \cdot \tau_{\text{sat}} \approx 0.0941$ as $n \to \infty$.

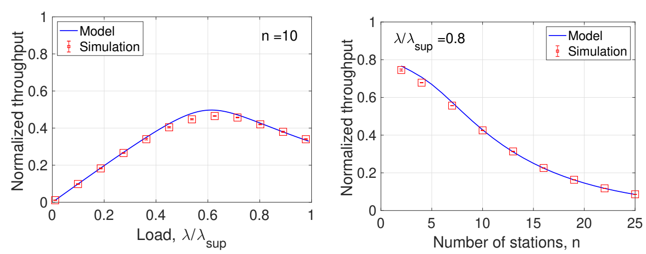
The normalized throughput $TH = \lambda P_s/\lambda_{\text{sup}}$ is shown in Figure 9. When plotted as a function of $\hat{\lambda}$, the normalized throughput exhibits a maximum. It is attained where offered

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410



(a) (b)

Figure 8. Probability of transmission τ for $\theta = 34 \cdot \delta$: (a) as a function of the normalized arrival rate $\lambda/\lambda_{\text{sup}}$, for $n = 10$ stations; (b) as a function of the number of stations for $\lambda/\lambda_{\text{sup}} = 0.8$.



(a) (b)

Figure 9. Normalized throughput $TH = \lambda P_s/\lambda_{\text{sup}}$ for $\theta = 34 \cdot \delta$: (a) as a function of the normalized arrival rate $\lambda/\lambda_{\text{sup}}$, for $n = 10$ stations; (b) as a function of the number of stations for $\lambda/\lambda_{\text{sup}} = 0.8$.

traffic and loss of packets due to collision strike an optimal balance. For smaller values of $\hat{\lambda}$, the channel is underloaded, For larger values of the load, the channel is congested. A monotonic decreasing behavior is seen when the normalized throughput is plotted against n, since in that case TH is proportional to P_s, which decreases as the number of stations grows.

Finally, Figure 10 plots the CCDF of the delay through the CM queue for $n = 10$ and $\lambda = 0.9 \cdot \lambda_{\text{sup}}$. The model matches the CCDF excellently, showing that its predictive power extends nicely to probability distribution, beyond moments.

VI. Design of adaptive flow control

The adaptive flow control aims at adjusting the parameter λ in an environment where the number of active stations can change over time, so as to drive the system towards an optimal working point, i.e., the one where the achievable stable throughput is maximized. We first characterize the optimal working point, then we state a simple adaptive updating of λ that drives the system to that working point.

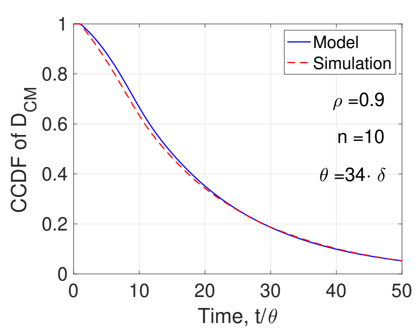


Figure 10. CCDF of the mean CM queue delay for $n = 10$ stations, $\theta = 34 \cdot \delta$ and queue load 0.9.

A. System optimization

The upper bound of the normalized maximum stable throughput $\eta = \nu\theta$ is (see Equation (22)):

$$\eta_{\text{sup}} = \theta\nu_{\text{sup}} = \max_{0 \le \tau \le \tau_{\text{sat}}} \frac{\tau(1-\tau)^{n-1}}{\beta + 1 - (1-\tau)^n}, \qquad (25)$$

where $\beta = \delta/\theta$. The maximum is achieved by setting τ equal to[6] $\min\{\tau_{\text{sat}}, \tau^*\}$, where τ^* is the unique root of the equation

$$(1-x)^n = (1+\beta)(1-nx), \quad x \in [0,1]. \qquad (26)$$

In the following we denote the solution of Equation (26) with $\tau^*(n)$ to stress its dependence on n. It can be verified that $\tau^*(n)$ is monotonously decreasing with n, from $\tau^*(1) = 1$ down to $\tau^*(\infty) = 0$.

An approximate solution of this equation can be found introducing the variable change $\alpha = nx$ and taking the limit for $n \to \infty$. Then, Equation (26) is approximated by:

$$e^{-\alpha} = (\beta+1)(1-\alpha) \qquad (27)$$

Since $\beta > 0$, it is easy to show that this equation has a unique positive solution belonging to the interval $(0,1)$. Let α^* denote the solution. Since $e^{-\alpha} > \left(1 - \frac{\alpha}{n}\right)^n$, $\forall n \ge 1$, it follows that $\tau^*(n) > \alpha^*/n$. It can also be verified $\lim_{n\to\infty} n\tau^*(n) = \alpha^*$, i.e., the approximation $\tau^*(n) \approx \alpha^*/n$ is asymptotically sharp as n grows.

Letting $\tau = \tau^*(n) \sim \alpha^*/n$ into Equation (12), we find for large n:

$$\lambda^* = \frac{\alpha^*/n}{\delta + \theta - \theta(1-\alpha^*/n)^n} \approx \frac{1}{n\theta}\frac{\alpha^*}{\beta + 1 - e^{-\alpha^*}} \overset{(*)}{=} \frac{1}{n\theta}\frac{1}{\beta+1} = \frac{1}{n(\theta+\delta)}$$

where we have used the fact that α^* is the root of Equation (27) to obtain the equality $(*)$.

The simple and intuitively appealing result is that *the optimal flow control rate to maximize the overall stable net throughput in Equation* (25) *is the inverse of* n *times the transmission time*, where n is the number of stations sharing the channel. Therefore, optimal flow control can be achieved by estimating n, the number of stations contending on the channel. This is known to be a feasible task, e.g., see [43], [44].

B. Classic versus flow control based optimization

The optimal working point of the MAC channel can be achieved by adjusting the back-off probability distribution. In case the back-off random variable is uniformly distributed over $\{1, 2, \dots, W_0\}$, the contention window size W_0 can be adjusted. This is the target of Idle Sense algorithm [45], [46], which is based on the observation of the number N_I of idle back-off

[6] For typical values of τ_{sat} and β, it turns out that $\tau_{\text{sat}} > \tau^*$ for all values of n greater than a small threshold, e.g., 2 or 3.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410

slots between two consecutive transmissions (a function of the number of contending stations) and uses the estimated mean value of N_I to adjust W_0.

In the proposed flow control approach, the optimal working point is pursued in a different way, namely by adjusting the flow control rate λ, while back-off parameters are fixed.

Let n_{eff} denote the number of effectively contending stations on the channel, i.e., stations that are backlogged. The fundamental theory developed originally in [5] suggests that throughput is maximized when contention window sizes are the same for all (re-)transmission attempts, and equal to the following optimal contention window size:

$$W^* = \frac{2n_{\text{eff}}}{\alpha^*} - 1\,, \tag{28}$$

where α^* is the unique root of Equation (27) in $[0,1]$. Here the control variable is the contention window size, while the number of contending stations n_{eff} is taken as input to optimization.

Flow control flips this perspective. It strikes the optimized balance in Equation (28), *for a given contention window size* W_0*, by making the average number of backlogged stations* $\overline{n}_{\text{eff}}$ *equal to the optimal level* n^* *such that* $2n^*/\alpha^* - 1 = W_0$. [7] In other words, λ is adjusted so that the level of contention on the channel, namely n_{eff}, hovers about the optimal level $n^* = \alpha^*(W_0+1)/2$ for the given contention window size W_0.

To prove this, we note that $\overline{n}_{\text{eff}} = nb$, where b is the probability that a station is backlogged in a virtual slot time. b is obtained as the ratio of the mean number of virtual slot times counted down by a station to the mean number of virtual slot times between two consecutive transmissions of the same station, namely:

$$b = \frac{\frac{W_0+1}{2}}{\frac{W_0+1}{2} + \frac{\pi_0}{1-\varphi_X(\lambda)}} = \frac{1}{1+\tau_{\text{sat}}\frac{1-\lambda \mathrm{E}[C]}{\lambda \mathrm{E}[X]}} = \frac{\lambda \mathrm{E}[X]/\tau_{\text{sat}}}{1-q\theta\lambda}. \tag{29}$$

Setting $\lambda = \lambda^* = 1/[n(\delta+\theta)]$, hence $\tau \approx \alpha^*/n$, and assuming n is large, so that $q = (1-\tau)^{n-1} \approx e^{-\alpha^*}$, we have

$$\begin{aligned}\overline{n}_{\text{eff}} = nb &= \frac{n\lambda^*\mathrm{E}[X]/\tau_{\text{sat}}}{1-e^{-\alpha^*}\theta\lambda^*} = \frac{\frac{W_0+1}{2}\,\frac{1}{\delta+\theta}\left(\delta+\theta-\theta e^{-\alpha^*}\right)}{1-e^{-\alpha^*}\frac{1}{n}\,\frac{\theta}{\delta+\theta}}\\ &\sim \frac{W_0+1}{2}\left(1-\frac{e^{-\alpha^*}}{\beta+1}\right) = \frac{W_0+1}{2}\alpha^*,\end{aligned}$$

where the last but one passage holds for large n (a regime where $q \sim e^{-\alpha^*}$ and $e^{-\alpha^*}\frac{1}{n}\,\frac{\theta}{\delta+\theta}$ is negligible with respect to 1), and the last passage stems from α^* being the solution of Equation (27).

C. Flow control algorithm

Let us consider the CBR metric, given in Equation (15). In the asymptotic regime for large n, with $\lambda = \lambda^* = 1/[n(\delta+\theta)]$, and hence $\tau \sim \alpha^*/n$, we have

$$\begin{aligned}\rho = \lambda^*\theta\frac{1-(1-\tau(\lambda^*))^n}{\tau(\lambda^*)} &\sim \frac{\theta}{n(\delta+\theta)}\frac{1-(1-\alpha^*/n)^n}{\alpha^*/n}\\ &\sim \frac{1-e^{-\alpha^*}}{(\beta+1)\alpha^*} = \frac{1-e^{-\alpha^*}}{\beta+1-e^{-\alpha^*}} = \rho^*.\end{aligned}$$

The expression of ρ^* depends only on the parameter $\beta = \delta/\theta$.

Let us refer to the time I between packet grants issued by the flow control algorithm. The mean of I is $\overline{I} = 1/\lambda$. The value of $\overline{I}$ that leads the system to its optimal working point, i.e., where the stable throughput is maximized, is given by $\overline{I}^* = 1/\lambda^* = n(\delta+\theta)$.

On the other hand, the CBR ρ is a monotonously increasing function of λ, hence monotonously decreasing with $\overline{I}$. At the optimal working point we have $\rho = \rho^*$. Then, the adaptive control strategy is to estimate the current ρ on the channel and adjust $\overline{I}$ so as to drive ρ towards ρ^*.

In the language of dynamical control systems, the mean flow control timer is governed by the dynamic law $\frac{d\overline{I}}{dt} = \kappa[\rho(t)-\rho^*]$, with κ a positive quantity and $\rho(t)$ the estimated CBR at time t. If the estimated CBR is smaller than the target one, the derivative is negative and we decrease the flow control timer, which is the right thing to do, since we are not using efficiently the channel due to too slack arrivals at the CM queue. If instead the estimated CBR is bigger than the target one, than stations are too aggressive, causing too many collisions, and we need to increase the flow control timer, which is what the dynamic equation dictates, since the derivative is positive.

Let us now translate the dynamical system statement into a discrete time algorithm for updating the mean flow control timer $\overline{I}$. To this end, we introduce the following notation:

- t_k is the k-th control update time. It corresponds to the end of channel transmission time[8];
- $\hat{\rho}_k$ is the estimated CBR at time t_k;
- $\overline{I}_k$ is the mean flow control timer value at t_k;
- Θ_k is the amount of time that the channel is sensed busy between t_{k-1} and t_k;
- m_ρ is the number of samples of CBR used in the moving average estimator of $\hat{\rho}$.

The estimated CBR is obtained as a moving average over the last m_ρ sampling time intervals, i.e.,

$$\hat{\rho}_k = \frac{\sum_{j=k-m_\rho+1}^{k}\Theta_j}{t_k - t_{k-m_\rho}} \tag{30}$$

The adaptation of $\overline{I}$ is defined as follows:

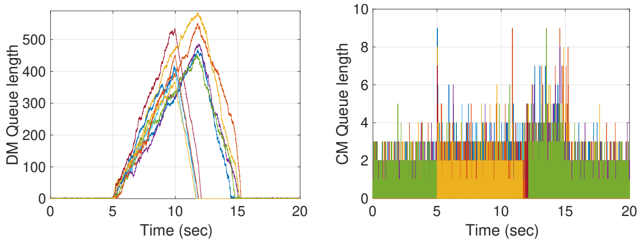
$$\overline{I}(t_{k+1}) = \begin{cases}\overline{I}(t_k)+\kappa(\delta+\theta) & \text{if } \hat{\rho}_k \geq \rho^*,\\ \max\{\overline{I}_{\min}, \overline{I}(t_k)-\kappa(\delta+\theta)\} & \text{otherwise.}\end{cases} \tag{31}$$

Correspondingly, we set $\lambda(t) = 1/\overline{I}(t_k)$, $\forall t \in [t_k, t_{k+1})$.

[7] Since $n_{\text{eff}} \leq n$, forcing n_{eff} to satisfy $n_{\text{eff}} = n^* = \frac{W_0+1}{2}\alpha^*$ implies that it must be $n \geq \alpha^*(W_0+1)/2$. With typical numerical values, it is $\alpha^* \ll 1$, so that this inequality is satisfied even for small n.

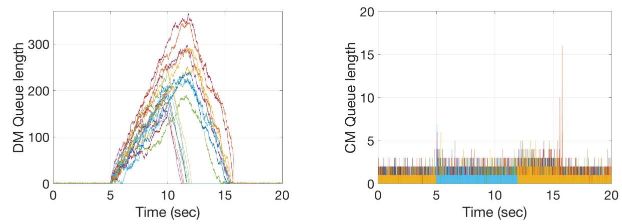
[8] As a matter of example, in the CSMA/CA protocol of Wi-Fi the end of channel activity time can be determined by observing an idle channel for a time equal to AIFS.

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410



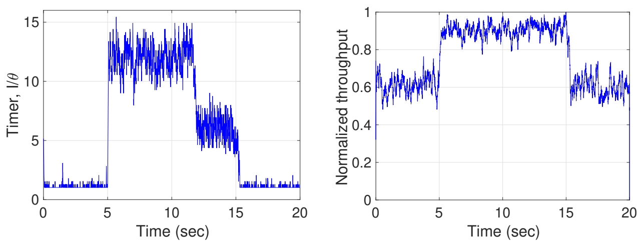
(a) Sample paths of DM queues. (b) Sample paths of CM queues.

Figure 11. Sample paths of (a) DM queues and (b) CM queues for the transient experiment described in Section VI-C with $n = 10$ stations.



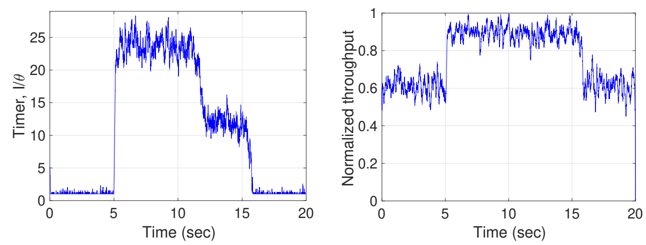
(a) Sample paths of DM queues. (b) Sample paths of CM queues.

Figure 13. Sample paths of (a) DM queues and (b) CM queues for the transient experiment described in Section VI-C with $n = 20$ stations.



(a) (b)

Figure 12. Two groups of $n = 5$ stations each, one of which is active through the whole simulation experiment, while the other one is active only between time 5 s and 12.5 s. (a) Sample path of timer $\overline{I}$ normalized with respect to activity time θ. (b) Sample path of normalized throughput.



(a) (b)

Figure 14. Two groups of $n = 10$ stations each, one of which is active through the whole simulation experiment, while the other one is active only between time 5 s and 12.5 s. (a) Sample path of timer $\overline{I}$ normalized with respect to activity time θ. (b) Sample path of normalized throughput.

D. Numerical examples

In the implementation of the algorithm, we set $\overline{I}_{\text{min}} = \delta + \theta$, $m_\rho = 5$, and $\kappa = 0.25$ as a default value. The mean flow control timer is initialized as $\overline{I}(t_0) = m_\rho(\delta + \theta)$, since in this time it is possible to collect enough data to estimate the CBR.

In a first experiment, we analyse the time behavior of the flow control adaptive algorithm by considering a set of $n = 10$ stations. Five stations start transmitting at time 0, generating an overall load equal to 60 % of the capacity of the system. After 5 s another group of five stations starts transmitting, offering an overall traffic equal to 60 % of the capacity of the channel. This second group of stations is active for 5 s. After that time, the five stations of the second group go back to idle, leaving only the initial five station to contend for the channel.

As a result, in the five seconds where both groups of stations share the channel, there is an overload, highlighted by the fast build-up of DM queues shown in Figure 11a. The flow control algorithms guarantees that the channel is not overloaded, causing a collapse of throughput as a consequence of massive collisions. This is clearly visibile from the steady and limited excursion of CM queues sample paths in Figure 11b.

The normalized timer $\overline{I}/\theta$ sample path is shown in Figure 12a. As long as the overall offered load equals 60 % of the channel capacity, the timer takes a low value, i.e., flow control is essentially letting packets move immediately to the CM queue as soon as they arrive. When the offered traffic surges up due to five new stations jumping on the channel, the normalized timer quickly raises up hitting the expected optimal value. As soon as the five newcomer stations stop transmitting, the normalized timer falls to an intermediate level, beacuse of the pressure caused by the backlog accumulated by the initial five sations during the overload time. As soon as the backlog is cleared, the normalized timer goes back to a baseline level corresponding to a low load on the system.

The time-varying channel coefficient of utilization, averaged over time intervals of 100 ms, is depicted in Figure 12b. The coefficient of utilization is obtained as the ratio of the achieved throughput and the maximum achievable throughput $n\nu_{\text{sup}}$. At the beginning it hovers around 60 %. As five more stations kick in, the utilization coefficient shifts close to 1, for the whole time where congestion is on. It lands quickly back to its baseline level of 60 % as soon as the additional five stations stop transmitting and the accumulated backlog is cleared. Thus, during overload, there is no throughput collapse. On the contrary, the capacity of the CSMA channel is used almost entirely (coefficient of utilization ≈ 1).

From Figure 11 it is also apparent that DM queue build up occurs at all stations, those that are already on the channel, and the newly arrived stations. This is evidence of fairness in channel sharing, i.e., stations using the channel back-off and give up shares of channel capacity to newcomers.

In a second experiment, we analyse the robustness of the proposed algorithm by doubling the number of stations, i.e., there are two groups of 10 stations each. A first group starts transmitting from the beginning of the experiment, while the second group of 10 stations sets on about 5 s after the beginning of the simulation.

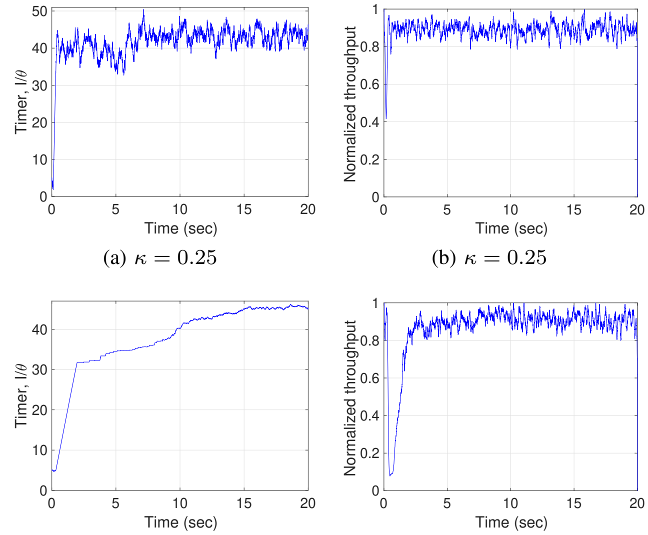
While the normalized throughput in Figure 14b is essentially the same as seen in Figure 12b, the flow control timer sample path in Figure 14a peaks at about twice the value it achieves in Figure 12a. This is expected, since the timer value targeted by the flow control algorithm is proportional to the overall

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410

number of contending stations, which is doubled in the second set of experiments as compared to the first set. As for queue dynamics, doubling the number of stations appears to reduce the fluctuations of DM queues. This effect, consistently observed in several other experiments (not shown for space reasons) may be attributed to a statistical smoothing effect when the system moves towards a large scale one, for the same overall load (i.e., each single station contributes less to the overall traffic load).

Regarding the flow control algorithm parameters, the gain coefficient κ affects the duration of transients and the accuracy of the control. Larger values of κ imply a faster convergence close to optimal values, but also larger fluctuations around the optimal value. To illustrate with an example, we consider $n = 40$ stations sharing the communication channel and always active throughout the simulation experiment. The mean arrival rate at DM queues is set so that the overall load is 90 % of the achievable maximum throughput.

The behavior of the station timer and of the overall throughput are shown in Figure 15. Figures 15a and 15b refer to $\kappa = 0.25$, while Figures 15c and 15d refer to $\kappa = 0.025$. The achieved normalized throughput hovers around 0.9 in both cases, consistently with the imposed traffic load. However, it is apparent that in case of $\kappa = 0.025$ there is a longer transient with respect to the more aggressive case where $\kappa = 0.25$. The achieved throughput drops initially due to the large number of contending stations and hence the large number of collisions. Then, the timer ramps up quickly, thus slowing down the flow of packets from DM queues to CM queues and relieving the congestion on the shared communication channel. After about 3 s a steady state is achieved. In case of $\kappa = 0.25$ the steady state is attained after less than 1 s, since the timer is adapted in a much faster way (compare Figure 15a and Figure 15c). In case $\kappa = 0.25$, the timer fluctuates significantly during steady state, while with $\kappa = 0.025$ the timer value maintains a smoother behavior, with essentially no fluctuations.



(a) $\kappa = 0.25$ (b) $\kappa = 0.25$

(c) $\kappa = 0.025$ (d) $\kappa = 0.025$

Figure 15. Analysis with $n = 40$ stations active through the simulation experiment: (a) and (c) sample path of timer $\overline{I}$ normalized with respect to activity time θ for $\kappa = 0.25$ and $\kappa = 0.025$, respectively; (b) and (d) sample path of normalized throughput for $\kappa = 0.25$ and $\kappa = 0.025$, respectively.

VII. Final remarks and model extensions

In this work we propose a modular design of the MAC layer, where the channel contention function is separated from data management functions such as data integrity and scheduling. We define a flow control function to connect the two functionalities. Flow control guarantees the stability of queues at the contention sublayer. It is based on a timed gate that opens up at randomized times, letting a packet from the upper DM sublayer pouring into the CM sublayer buffer. The mean value of the timer can be optimized adaptively as a function of the number of contending stations on the channel. The adaptive algorithm is based on the insight gained out of the presented analytical model and its asymptotic analysis for large numbers of stations. Numerical results show that the model provides highly accurate predictions for all values of the number of stations.

The second major contribution of the paper is an analytical model of the CM sublayer, leading to an accurate model of a non saturated channel run according to non-persistent CSMA. The model assumes symmetric stations (all having the same offered traffic statistics) and fixed transmission times. Those assumptions can be relaxed, still retaining the modeling approach based on taking the point of view of one tagged station and dealing with the effect of all others by means of a kind of mean field approach. Allowing variable transmission times as well as accounting for heterogeneous scenarios, where traffic loads offered by different stations are different, is part of future work.

The presented functional split concept, with adaptive flow control, can be applied to any underlying MAC channel sharing algorithm. Investigating the application of the flow control paradigm to MAC protocols other than CSMA is also part of future work.

Appendix

A. Analysis of the CM queue

1) Embedded Markov Chain model of the CM queue: From Equation (2), we derive the LT of the PDF of the service time C as:

$$\varphi_C(s) = e^{-(\theta+\delta)s} \frac{\phi_M(\varphi_X(s))}{\varphi_X(s)} \tag{32}$$

where $\phi_M(z) = \mathrm{E}[z^M]$ is the Generating Function (GF) associated with the random variable M and $\varphi_X(s)$ is the LT of the PDF of X. From the definition of X in Equation (3) we derive:

$$\varphi_X(s) = e^{-s\delta}\left[q + (1-q)e^{-s\theta}\right] \tag{33}$$

The first two moments of C and X can be calculated by deriving the LTs of their respective PDFs and setting $s = 0$:

$$\mathrm{E}[C] = \theta + \delta + (\mathrm{E}[M] - 1)\mathrm{E}[X] \tag{34}$$

$$\sigma_C^2 = \sigma_M^2(\mathrm{E}[X])^2 + (\mathrm{E}[M] - 1)\sigma_X^2 \tag{35}$$

and

$$\mathrm{E}[X] = \delta + (1-q)\theta \tag{36}$$
$$\sigma_X^2 = q(1-q)\theta^2 \tag{37}$$

where σ_v^2 is the variance of the random variable v.

In the special case where M is uniformly distributed over $\{1, \dots, W_0\}$, we have for the LT of the PDF of C:

$$\varphi_C(s) = e^{-(\theta+\delta)s} \frac{1 - [\varphi_X(s)]^{W_0}}{W_0[1 - \varphi_X(s)]} \tag{38}$$

and for the first two moments:

$$\mathrm{E}[C] = \theta + \delta + \frac{W_0 - 1}{2} \mathrm{E}[X] \tag{39}$$
$$\sigma_C^2 = \frac{W_0^2 - 1}{12} (\mathrm{E}[X])^2 + \frac{W_0 - 1}{2} \sigma_X^2 \tag{40}$$

By means of standard Markov chain analysis methods applied to Equation (1) (e.g., see [47, Ch. 5]), it can be found that the GF of the limiting probability distribution of Q_k at steady-state is given by

$$\phi_Q(z) = \pi_0 \frac{[\phi_B(z) - 1]\, \phi_A(z)}{z - \phi_A(z)} \tag{41}$$

where $\pi_0 = \mathcal{P}(Q = 0)$, and $\phi_A(z), \phi_B(z)$ are the GFs of the number A of arrivals in a service time and the number B of arrivals in the residual virtual slot time at the beginning of a busy period, respectively. The probability π_0 that a departing packet leaves an empty queue behind is found by requiring that $\phi_Q(1) = 1$ and applying de l'Hôpital's rule. It follows:

$$\pi_0 = \frac{1 - \phi'_A(1)}{\phi'_B(1)} \tag{42}$$

where $'$ denotes derivation. Steady-state exists provided the mean number of arrivals at the CM queue in a virtual slot time is less than 1, i.e., $\phi'_A(1) < 1$. The mean queue length is found as follows, again applying de l'Hôpital's rule:

$$\mathrm{E}[Q] = \phi'_Q(1) = \frac{\phi''_B(1)}{2\phi'_B(1)} + \frac{\phi''_A(1)}{2[1 - \phi'_A(1)]} + \phi'_A(1) \tag{43}$$

As for the GFs of A and B, let $N(u)$ be the number of arrivals of a Poisson process with mean rate λ in a time interval of duration u. Then

$$\begin{aligned}\phi_A(z) &= \mathrm{E}[z^{N(C)}] = \mathrm{E}[\mathrm{E}[z^{N(C)} | C = y]] \\ &= \int_0^\infty e^{\lambda y(z-1)} f_C(y)\, dy = \varphi_C(\lambda - \lambda z)\end{aligned}$$

As for B, it counts arrivals in a virtual slot time, given that there is at least one arrival (and hence a busy period of the queue can start):

$$\begin{aligned}\phi_B(z) &= \mathrm{E}[z^{N(X)} | N(X) > 0] \\ &= \sum_{k=1}^{\infty} z^k \mathcal{P}(N(X) = k \,|\, N(X) > 0) \\ &= \frac{\sum_{k=1}^{\infty} z^k \int_0^\infty \frac{(\lambda u)^k}{k!} e^{-\lambda u}\, dF_X(u)}{\mathcal{P}(N(X) > 0)} \\ &= \frac{\int_0^\infty \left(e^{-(\lambda - \lambda z)u} - e^{-\lambda u}\right) dF_X(u)}{1 - \mathcal{P}(N(X) = 0)} \\ &= \frac{\varphi_X(\lambda - \lambda z) - \varphi_X(\lambda)}{1 - \varphi_X(\lambda)}\end{aligned}$$

The first moments of A and B are

$$\begin{aligned}\mathrm{E}[A] &= \phi'_A(1) = \lambda \mathrm{E}[C] \\ \mathrm{E}[B] &= \phi'_B(1) = \frac{\lambda \mathrm{E}[X]}{1 - \varphi_X(\lambda)}\end{aligned}$$

The second derivatives of the GFs of A and B are also easily computed, yielding $\phi''_A(1) = \lambda^2 \mathrm{E}[C^2]$, and $\phi''_B(1) = \lambda^2 \mathrm{E}[X^2]/[1 - \varphi_X(\lambda)]$. Hence

$$\begin{aligned}\mathrm{E}[A^2] &= \phi''_A(1) + \phi'_A(1) = \lambda^2 \mathrm{E}[C^2] + \lambda \mathrm{E}[C] \\ \mathrm{E}[B^2] &= \phi''_B(1) + \phi'_B(1) = \frac{\lambda^2 \mathrm{E}[X^2] + \lambda \mathrm{E}[X]}{1 - \varphi_X(\lambda)}\end{aligned}$$

Substituting the expression of the first derivatives of $\phi_A(z)$ and $\phi_B(z)$ into Equation (42), we obtain the expression of π_0 in Equation (5).

Waiting times can be found by considering that

$$\phi_Q(z) = \varphi_C(\lambda - \lambda z)\varphi_W(\lambda - \lambda z) \tag{44}$$

where W denotes the waiting time random variable. It can be deduced that

$$\begin{aligned}\varphi_W(s) &= \frac{\pi_0 \lambda\, [1 - \phi_B(1 - s/\lambda)]}{s - \lambda + \lambda \varphi_C(s)} \\ &= \frac{1 - \lambda \mathrm{E}[C]}{\mathrm{E}[X]} \frac{1 - \phi_X(s)}{s - \lambda + \lambda \varphi_C(s)}\end{aligned}$$

2) Transmission probability: Let τ be the steady-state probability that a station transmits in a virtual slot time. At equilibrium, the probability τ is the long-run average fraction of virtual time slots where a station transmits. Let us observe the tagged station queue over busy cycles, a busy cycle being composed of an idle time followed by a busy period. Busy cycles form a renewal process, thanks to the Poisson arrival assumption. A renewal reward argument shows that

$$\tau = \frac{\mathrm{E}[K]}{\mathrm{E}[M]\mathrm{E}[K] + \mathrm{E}[J]} \tag{45}$$

where K is the number of packet transmissions performed in a busy period, and J is the number of virtual slot times in an idle time.

Using the result on the mean number of packets served in a busy period from standard $M/G/1$ theory, we have

$$\mathrm{E}[K] = \frac{\phi'_B(1)}{1 - \phi'_A(1)} = \frac{1}{\pi_0} \tag{46}$$

The idle time consists of a sequence of virtual time slots where the stations other than the tagged one possibly transmit their packets. Since arrivals at the tagged station follow a Poisson process, the number J of virtual slot times making up an idle time is geometrically distribution with ratio $\mathcal{P}(\text{no arrivals at the tagged station in a virtual slot time}) = \mathcal{P}(N(X) = 0) = \varphi_X(\lambda)$. Formally, we have

$$\mathcal{P}(J = k) = [1 - \varphi_X(\lambda)][\varphi_X(\lambda)]^{k-1}, \quad k \geq 1 \tag{47}$$

Hence the mean number of virtual slot times in an idle time is

$$\mathrm{E}[J] = \frac{1}{1 - \varphi_X(\lambda)} \tag{48}$$

This article has been accepted for publication in IEEE/ACM Transactions on Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNET.2023.3239410

Summing up, from Equations (45), (46) and (48), we get:

$$\tau = \frac{1/\pi_0}{\mathrm{E}[M]\frac{1}{\pi_0} + \frac{1}{1-\varphi_X(\lambda)}} = \frac{\tau_{\text{sat}}}{1+\tau_{\text{sat}}\frac{\pi_0}{1-\varphi_X(\lambda)}} \tag{49}$$

where $\tau_{\text{sat}} = 1/\mathrm{E}[M]$ is the limit of τ as the traffic load tends to saturation, i.e., $\pi_0 \to 0$. In case the back-off counter M is uniformly distributed over $\{1,\dots,W_0\}$, we have $\mathrm{E}[M] = (W_0+1)/2$.

Substituting π_0 from Equation (5) into the expression of τ in Equation (49), we find

$$\begin{aligned}\tau &= \frac{1}{\mathrm{E}[M] + \frac{1-\lambda\mathrm{E}[C]}{\lambda\mathrm{E}[X]}} \\ &= \frac{\mathrm{E}[X]}{\mathrm{E}[M]\mathrm{E}[X] + \frac{1}{\lambda} - [(\mathrm{E}[M]-1)\mathrm{E}[X]+\theta+\delta]} \\ &= \frac{\mathrm{E}[X]}{\frac{1}{\lambda} - q\theta} = \frac{\beta+1-q}{\frac{1}{\lambda\theta}-q}\end{aligned} \tag{50}$$

where $\beta = \delta/\theta$. This proves Equation (7).

B. Proof of Theorem 1

To prove the theorem, we appeal to the uniqueness Theorem stated in [48]. We need to prove the two conditions of the theorem statement, namely: (a) 1 is not an eigenvalue of the derivative matrix $\mathbf{\Phi}'$; (b) no solution exists on the boundary of the region where the map $\mathbf{\Phi}$ is applied.

Let us first prove that condition (b) above holds in our case. Assume $\tau^{(i)} = 1$. The i-th equation of the system $\tau = \mathbf{\Phi}(\tau)$ yields

$$1 = \tau^{(i)} = \frac{\beta+1-q^{(i)}}{\frac{1}{\lambda^{(i)}\theta} - q^{(i)}} \quad \Rightarrow \quad \lambda^{(i)}\theta = \frac{1}{\beta+1} \tag{51}$$

This last equality contradicts condition (ii) assumed in the statement of Theorem 1, hence it cannot be $\tau^{(i)} = 1$ for any i.

Let us now consider condition (a) of the theorem in [48]. The entry (i,k) of the derivative matrix $\mathbf{\Phi}'$ is

$$\frac{\partial\Phi_i}{\partial\tau^{(k)}} = \begin{cases} \frac{\frac{1}{\lambda^{(i)}\theta}-\beta-1}{\left(\frac{1}{\lambda^{(i)}\theta}-q^{(i)}\right)^2} & k \neq i \\ 0 & k = i \end{cases} \tag{52}$$

Re-arranging Equation (7), we get

$$\lambda^{(i)}\theta = \frac{\tau^{(i)}}{\beta+1-(1-\tau^{(i)})q^{(i)}} \tag{53}$$

Inserting this expression in the derivative in Equation (52) for $i \neq k$, we have

$$\frac{\partial\Phi_i}{\partial\tau^{(k)}} = \frac{(1-\tau^{(i)})\tau^{(i)}q^{(i)}}{(1-\tau^{(k)})(\beta+1-q^{(i)})} \tag{54}$$

We need to prove that there does not exist a non-null vector $\mathbf{v}$ such that $\mathbf{\Phi}'\mathbf{v} = \mathbf{v}$. We will prove this by contradiction. Assume that such a vector exists. Hence, we can write

$$v_i = \sum_{k\neq i} v_k \frac{\partial\Phi_i}{\partial\tau^{(k)}} = \frac{(1-\tau^{(i)})\tau^{(i)}q^{(i)}}{\beta+1-q^{(i)}} \sum_{k\neq i}\frac{v_k}{1-\tau^{(k)}} \tag{55}$$

Note that we have already proved that it must be $\tau^{(i)} < 1$, $\forall i$. Let $u_k = v_k/(1-\tau^{(k)})$. From Equation (55) we derive

$$u_i\frac{\beta+1-q^{(i)}}{\tau^{(i)}q^{(i)}} = \sum_{k\neq i} u_k \tag{56}$$

Summing u_i on both sides, we get

$$u_i\frac{\beta+1-(1-\tau^{(i)})q^{(i)}}{\tau^{(i)}q^{(i)}} = \frac{1}{\lambda^{(i)}\theta q^{(i)}}u_i = \sum_{k=1}^{n} u_k \tag{57}$$

for $i = 1,\dots,n$. This proves that all components of the vector $\mathbf{u} = [u_1,\dots,u_n]$ have the same sign or are equal to 0. Since we are assuming that $\mathbf{v}$, hence $\mathbf{u}$, is not identically null, all components of $\mathbf{u}$ must be either positive or negative. As a consequence, the sum on the rightmost-hand side of Equation (57) is non null. Multiplying both sides of Equation (57) by $\lambda^{(i)}\theta q^{(i)}$ and summing up over i, we get

$$\sum_{i=1}^{n} u_i = \sum_{i=1}^{n}\lambda^{(i)}\theta q^{(i)}\sum_{k=1}^{n}u_k \quad \Rightarrow \quad \sum_{i=1}^{n}\lambda^{(i)}\theta q^{(i)} = 1 \tag{58}$$

where the implication stems from the fact that $\sum_{i=1}^{n} u_i \neq 0$. The result in Equation (58) contradicts the condition (i) of Theorem 1, hence there cannot exist a nonnull vector $\mathbf{v}$ which is an eigenvector of $\mathbf{\Phi}'$ corresponding to the eigenvalue 1. This completes the proof.point (a) of the theorem in [48]. Then, the proof of Theorem 1 is complete.

C. Proof of Theorem 2

The DM queue is stable provided $\nu < \lambda P_s$, i.e., if and only if

$$\nu\theta < \frac{\tau(1-\tau)^{n-1}}{\beta+1-(1-\tau)^n} = g(\tau) \tag{59}$$

where $\beta = \delta/\theta$ and τ is determined from λ by inverting the following monotonous mapping:

$$\lambda\theta = \frac{\tau}{\beta+1-(1-\tau)^n}, \quad \tau \in [0, \tau_{\text{sat}}]. \tag{60}$$

Let $\hat{\nu} = \nu\theta$ and $\hat{\lambda} = \lambda\theta$ denote the normalized arrival rate at DM queue and CM queue, respectively. The feasible range of $\hat{\lambda}$ is $(0, \hat{\lambda}_{\text{sup}})$, where

$$\hat{\lambda}_{\text{sup}} = \frac{\tau_{\text{sat}}}{\beta+1-(1-\tau_{\text{sat}})^n} \tag{61}$$

The function $g(\tau)$ defined in Equation (59) is monotonously increasing from 0 up to ν_{sup} for $\tau \in [0,\tau^*]$, then it decreases monotonically from ν_{sup} down to 0 when $\tau \in [\tau^*, 1]$, where τ^* is the unique solution in $[0,1]$ of the equation $g'(\tau) = 0$, i.e.,

$$(1-\tau)^n = (\beta+1)(1-n\tau) \tag{62}$$

We distinguish two cases.

a) $\tau_{sat} \leq \tau^*$: . Since the range of feasible τ is restricted to $[0,\tau_{\text{sat}}]$, if follows that $g(\tau)$ is monotonously increasing for the whole range of feasible values of τ. Hence, for any given $\nu < \nu_{\text{sup}}$ there exists a unique intersection of $g(\tau)$ with the horizontal line at level $\hat{\nu}$. Correspondingly, a unique value of $\hat{\lambda}$ is determined, through the monotonous mapping in Equation (60), say it is $\hat{\lambda}_{\text{min}}$. The inequality $\hat{\nu} < g(\tau)$ is met for all values of $\hat{\lambda}$ such that $\hat{\lambda} \in (\hat{\lambda}_{\text{min}}, \hat{\lambda}_{\text{sup}})$. Hence, the statement of the theorem is proved with $\hat{\lambda}_{\text{max}} = \hat{\lambda}_{\text{sup}}$.

b) $\tau_{sat} > \tau^*$: . In this case, the location τ^* of the maximum of $g(\tau)$ is interior in the interval $[0, \tau_{\text{sat}}]$. For $\hat{\nu} < g(\tau_{\text{sat}})$ there is a single intersection of the function $g(\tau)$ with the horizontal line at level $\hat{\nu}$. Hence, the inequality $\hat{\nu} < g(\tau)$ is met for $\hat{\lambda}$ ranging in the interval $(\hat{\lambda}_{\text{min}}, \hat{\lambda}_{\text{sup}})$, as in the previous case.

If instead it is $\hat{\nu} \geq g(\tau_{\text{sat}})$, the equality $\hat{\nu} = g(\tau)$ has two solutions, one smaller and the other larger than τ^*. Let the two solutions be τ_1 and τ_2, with $\tau_1 < \tau_2$. Those two values of τ are mapped to two values of $\hat{\lambda}$ through Equation (60), say they are $\hat{\lambda}_{\text{min}}$ and $\hat{\lambda}_{\text{max}}$, where it is $\hat{\lambda}_{\text{max}} \leq \hat{\lambda}_{\text{sup}}$. Since $\hat{\nu} < g(\tau), \ \forall \tau \in (\tau_1, \tau_2)$, the DM queue is stable for any $\hat{\lambda} \in (\hat{\lambda}_{\text{min}}, \hat{\lambda}_{\text{max}})$, which completes the proof. $\square$

REFERENCES

[1] L. Kleinrock and F. Tobagi, "Packet Switching in Radio Channels: Part I - Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics," *IEEE Transactions on Communications*, vol. 23, no. 12, pp. 1400–1416, 1975.

[2] F. Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part II - The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution," *IEEE Transactions on Communications*, vol. 23, no. 12, pp. 1417–1433, 1975.

[3] F. Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part III - Polling and (Dynamic) Split-Channel Reservation Multiple Access," *IEEE Transactions on Communications*, vol. 24, no. 8, pp. 832–845, 1976.

[4] F. Tobagi and L. Kleinrock, "Packet Switching in Radio Channels: Part IV - Stability Considerations and Dynamic Control in Carrier Sense Multiple Access," *IEEE Transactions on Communications*, vol. 25, no. 10, pp. 1103–1119, 1977.

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

[6] A. Kumar, E. Altman, D. Miorandi, and M. Goyal, "New Insights From a Fixed-Point Analysis of Single Cell IEEE 802.11 WLANs," *IEEE/ACM Transactions on Networking*, vol. 15, no. 3, pp. 588–601, 2007.

[7] L. Jiang and J. Walrand, "A Distributed CSMA Algorithm for Throughput and Utility Maximization in Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 960–972, 2010.

[8] M. Garetto and C.-F. Chiasserini, "Performance analysis of 802.11 WLANs under sporadic traffic," in *International Conference on Research in Networking*, Springer, 2005, pp. 1343–1347.

[9] C. G. Park, H. S. Jung, and D. H. Han, "Queueing analysis of IEEE 802.11 MAC protocol in wireless LAN," in *International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06)*, IEEE, 2006, pp. 139–139.

[10] D. Malone, K. Duffy, and D. Leith, "Modeling the 802.11 Distributed Coordination Function in Nonsaturated Heterogeneous Conditions," *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, pp. 159–172, 2007.

[11] N. T. Dao and R. A. Malaney, "A New Markov Model for Non-Saturated 802.11 Networks," in *2008 5th IEEE Consumer Communications and Networking Conference*, 2008, pp. 420–424.

[12] O. Tickoo and B. Sikdar, "A queueing model for finite load IEEE 802.11 random access MAC," in *2004 IEEE International Conference on Communications (IEEE Cat. No.04CH37577)*, vol. 1, 2004, pp. 175–179.

[13] E. Felemban and E. Ekici, "Single Hop IEEE 802.11 DCF Analysis Revisited: Accurate Modeling of Channel Access Delay and Throughput for Saturated and Unsaturated Traffic Cases," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3256–3266, 2011.

[14] Q. Zhao, D. H. Tsang, and T. Sakurai, "Modeling nonsaturated IEEE 802.11 DCF networks utilizing an arbitrary buffer size," *IEEE Transactions on Mobile Computing*, vol. 10, no. 9, pp. 1248–1263, 2010.

[15] S. H. Nguyen, H. L. Vu, and L. L. H. Andrew, "Performance Analysis of IEEE 802.11 WLANs With Saturated and Unsaturated Sources," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 1, pp. 333–345, 2012.

[16] Q. Zhao, D. H. K. Tsang, and T. Sakurai, "A Scalable and Accurate Nonsaturated IEEE 802.11e EDCA Model for an Arbitrary Buffer Size," *IEEE Transactions on Mobile Computing*, vol. 12, no. 12, 2455?2469, Dec. 2013. [Online]. Available: https://doi.org/10.1109/TMC.2012.211.

[17] N. Mukta Gupta, "Transform Free Modeling of Finite Buffer Non-Saturated IEEE 802.11 DCF in Ad Hoc Networks," *Int J Wireless Inf Networks*, vol. 27, pp. 197–206, 2020.

[18] F. Daneshgaran, M. Laddomada, F. Mesiti, and M. Mondin, "On the Linear Behaviour of the Throughput of IEEE 802.11 DCF in Non-Saturated Conditions," *IEEE Communications Letters*, vol. 11, no. 11, pp. 856–858, 2007.

[19] Q. Zhao, D. H. K. Tsang, and T. Sakurai, "A simple model for nonsaturated IEEE 802.11 DCF networks," *IEEE Communications Letters*, vol. 12, no. 8, pp. 563–565, 2008.

[20] K. Ghaboosi, B. H. Khalaj, Y. Xiao, and M. Latva-aho, "Modeling IEEE 802.11 DCF Using Parallel Space?Time Markov Chain," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 4, pp. 2404–2413, 2008.

[21] O. Tickoo and B. Sikdar, "Modeling Queueing and Channel Access Delay in Unsaturated IEEE 802.11 Random Access MAC Based Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 4, pp. 878–891, 2008.

[22] F. Daneshgaran, M. Laddomada, F. Mesiti, and M. Mondin, "Unsaturated Throughput Analysis of IEEE 802.11 in Presence of Non Ideal Transmission Channel and Capture Effects," *IEEE Transactions on Wireless Communications*, vol. 7, no. 4, pp. 1276–1286, 2008.

[23] Q. Zhao, D. H. K. Tsang, and T. Sakurai, "A Simple and Approximate Model for Nonsaturated IEEE 802.11 DCF," *IEEE Transactions on Mobile Computing*, vol. 8, no. 11, pp. 1539–1553, 2009.

[24] K. Vijayasankar, A. Taufique, L. N. Kannan, M. Tacca, and A. Fumagalli, "An Analytical Model with Improved Accuracy of IEEE 802.11 Protocol Under Unsaturated Conditions," in *2010 IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2010, pp. 385–387.

[25] G. Prakash and P. Thangaraj, "Throughput Analysis of IEEE 802.11b WLAN under a Non-saturated Condition," in *2010 International Conference on Recent Trends in Information, Telecommunication and Computing*, 2010, pp. 65–69.

[26] X. Ma, X. Chen, and H. H. Refai, "Unsaturated Performance of IEEE 802.11 Broadcast Service in Vehicle-to-Vehicle Networks," in *2007 IEEE 66th Vehicular Technology Conference*, 2007, pp. 1957–1961.

[27] R. Oliveira, L. Bernardo, and P. Pinto, "Modelling Delay on IEEE 802.11 MAC Protocol for Unicast and Broadcast Nonsaturated Traffic," in *2007 IEEE Wireless Communications and Networking Conference*, 2007, pp. 463–467.

[28] N. Abdullah, R. Piechocki, and A. Doufexi, "Systematic raptor codes for safety broadcast in an unsaturated vehicular highway environment," *J. Wireless Com Network*, vol. 213, 2012.

[29] M. Ni, Z. Zhong, J. Pan, D. Zhao, and Ruifeng Chen, "Non-saturated performance analysis of IEEE 802.11 broadcast in 2-D mobile ad hoc networks," in *2012 IEEE Global Communications Conference (GLOBECOM)*, 2012, pp. 427–432.

[30] Jinsong Zhang and Xiaomin Ma, "Broadcast performance analysis of IEEE 802.11 networks under fading channels," in *2013 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, 2013, pp. 18–21.

[31] N. Lyamin, B. Bellalta, and A. Vinel, "Age-of-Information-Aware Decentralized Congestion Control in VANETs," *IEEE Networking Letters*, vol. 2, no. 1, pp. 33–37, 2020.

[32] A. Baiocchi and I. Turcanu, "Age of Information of One-Hop Broadcast Communications in a CSMA Network," *IEEE Communications Letters*, vol. 25, no. 1, pp. 294–298, 2021.

[33] A. Baiocchi, I. Turcanu, and A. Vinel, "Age of Information in CSMA-based Networks with Bursty Update Traffic," *IEEE Access*, vol. 10, pp. 44 088–44 105, Apr. 2022.

[34] J. Ni, B. Tan, and R. Srikant, "Q-CSMA: Queue-length-based CSMA/CA algorithms for achieving maximum throughput and low delay in wireless networks," *IEEE/ACM Transactions on Networking*, vol. 20, no. 3, pp. 825–836, 2011.

[35] A. Maatouk, M. Assaad, and A. Ephremides, "Energy efficient and throughput optimal CSMA scheme," *IEEE/ACM Transactions on Networking*, vol. 27, no. 1, pp. 316–329, 2019.

[36] J. Lee, H. Lee, Y. Yi, S. Chong, E. W. Knightly, and M. Chiang, "Making 802.11 DCF near-optimal: Design, implementation, and evaluation," *IEEE/ACM Transactions on Networking*, vol. 24, no. 3, pp. 1745–1758, 2015.

[37] J. Liu, Y. Yi, A. Proutiere, M. Chiang, and H. V. Poor, "Towards utility-optimal random access without message passing," *Wireless Communications and Mobile Computing*, vol. 10, no. 1, pp. 115–128, 2010.

[38] P. Marbach and A. Eryilmaz, "A backlog-based CSMA mechanism to achieve fairness and throughput-optimality in multihop wireless networks," in *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, IEEE, 2008, pp. 768–775.

[39] S. Rajagopalan, D. Shah, and J. Shin, "Network adiabatic theorem: an efficient randomized protocol for contention resolution," in *Proceedings of the Eleventh International Joint Conference on Measurement and Modeling of Computer Systems*, 2009, pp. 133–144.

[40] Q. Xia, W. Wang, L. Chen, and Z. Zhang, "VQ-CSMA: Throughput-optimal Low-delay Random Access," *IEEE Wireless Communications Letters*, vol. 11, no. 6, pp. 1303–1307, 2022.

[41] J. S. Pathmasuntharam, A. Das, and P. Mohapatra, "A flow control framework for improving throughput and energy efficiency in CSMA/CA based wireless multihop networks," in *2006 International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'06)*, IEEE, 2006.

[42] L. Kleinrock, *Queueing systems - Vol. 1: Theory*. John Wiley & Sons, 1975.

[43] G. Bianchi and I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network," in *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428)*, vol. 2, 2003, 844–852 vol.2.

[44] I. Kadota, A. Baiocchi, and A. Anzaloni, "Kalman filtering: estimate of the numbers of active queues in an 802.11 e EDCA WLAN," *Computer communications*, vol. 39, pp. 54–64, Feb. 2014.

[45] M. Heusse, F. Rousseau, R. Guillier, and A. Duda, "Idle Sense: An Optimal Access Method for High Throughput and Fairness in Rate Diverse Wireless LANs," *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 4, 121?132, Aug. 2005. [Online]. Available: https://doi.org/10.1145/1090191.1080107.

[46] Y. Grunenberger, M. Heusse, F. Rousseau, and A. Duda, "Experience with an Implementation of the Idle Sense Wireless Access Method," in *Proceedings of the 2007 ACM CoNEXT Conference*, ser. CoNEXT '07, New York, New York: Association for Computing Machinery, 2007. [Online]. Available: https://doi.org/10.1145/1364654.1364685.

[47] D. Gross, S. F. John, M. J. Thompson, and M. C. Harris, *Queueing theory*. Wiley, 2008.

[48] R. B. Kellogg, "Uniqueness in the Schauder fixed point theorem," *Proceedings of the American Mathematical Society*, vol. 60, no. 1, pp. 207–210, 1976.

Andrea Baiocchi received his "Laurea" degree in Electronics Engineering in 1987 and his PhD degree in Information and Communications Engineering in 1992, both from the University of Roma "La Sapienza". Since January 2005 he is a Full Professor in the Department of Information Engineering, Electronics and Telecommunications of the University of Roma Sapienza. His main scientific contributions are on telecommunications network traffic engineering, queuing theory, resource sharing and congestion control. His current research interests focus on massive multiple access and vehicular networking. Andrea's research activities have been carried out also in the framework of many national (CNR, MIUR, POR) and international (European Union, ESA) projects, also taking coordination and responsibility roles. Andrea has published more than 170 papers on international journals and conference proceedings. He has participated to the Technical Program Committees of eighty international conferences. He also served in the editorial board of the telecommunications technical journal published by Telecom Italia (currently TIM) for ten years. He is currently Associate Editor of Elsevier Vehicular Communications journal. He is the author of the book "Network Traffic Engineering – Stochastic models and Applications", Wiley, 2020.

Ion Turcanu received his Ph.D. degree in information and communications technologies and his M.Sc. degree in computer science from the University of Rome Sapienza in 2018 and 2014 respectively. He is currently heading the Edge Computing & Networks (EDGE) group at the Luxembourg Institute of Science and Technology (LIST). Previously, he was research & technology associate at LIST and postdoctoral researcher at the Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg. His research interests include cooperative networking systems with highly dynamic topologies, time-sensitive networking for in-vehicle communications, and next-generation cellular networks.