



# Inference of Wi-Fi busy time fraction based on Markov chains

Nour El Houda Bouzouita, Anthony Busson, Herve Rivano

## ► To cite this version:

Nour El Houda Bouzouita, Anthony Busson, Herve Rivano. Inference of Wi-Fi busy time fraction based on Markov chains. *Ad Hoc Networks*, 2022, pp.102963. 10.1016/j.adhoc.2022.102963 . hal-03762029

**HAL Id: hal-03762029**

**<https://inria.hal.science/hal-03762029>**

Submitted on 26 Aug 2022

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

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

## Journal Pre-proof

Inference of Wi-Fi busy time fraction based on Markov chains

Nour El Houda Bouzouita, Anthony Busson, Herve Rivano

PII: S1570-8705(22)00138-X

DOI: <https://doi.org/10.1016/j.adhoc.2022.102963>

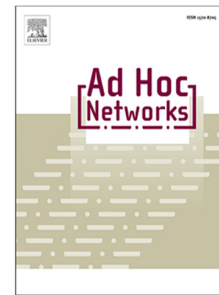
Reference: ADHOC 102963

To appear in: *Ad Hoc Networks*

Received date: 20 April 2022

Revised date: 22 July 2022

Accepted date: 6 August 2022



Please cite this article as: N.E.H. Bouzouita, A. Busson and H. Rivano, Inference of Wi-Fi busy time fraction based on Markov chains, *Ad Hoc Networks* (2022), doi: <https://doi.org/10.1016/j.adhoc.2022.102963>.

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2022 Elsevier B.V. All rights reserved.

# Inference of Wi-Fi Busy Time Fraction based on Markov Chains

Nour El Houda Bouzouita<sup>a,\*</sup>, Anthony Busson<sup>a</sup>, Herve Rivano<sup>b</sup>

<sup>a</sup>Univ Lyon, UCBL, EnsL, CNRS, Inria, LIP, F-69342, LYON Cedex 07, France.

<sup>b</sup>Univ Lyon, INSA Lyon, Inria, CITI, EA3720, 69621 Villeurbanne, France

---

## Abstract

IEEE 802.11 has emerged as a vital wireless network access technology for mobile devices. By providing the potential for high connectivity speeds, this technology has led to a huge rise in the number of access points (APs). In such environments, mobile devices have the choice to join several Wi-Fi networks. Despite its importance to user Quality of Experience (QoE), the AP selection is still trivial since it focuses at best on the received signal strength if not only the user's history. Crucial metrics that capture the overall dynamics of the AP load condition, such as the network load, are not taken into account. In this paper, we propose to use the Busy Time Fraction (BTF) as a metric to choose the best AP to attach to. The BTF level of a given channel is inferred based on the frame aggregation scheme proposed since the 802.11n standard. In this regard, we build a proof of concept system, FAM (Frame Aggregation based method) that leverages the theoretical frame aggregation levels of a probe traffic returned by two analytical Markovian models and the measured ones in order to estimate not only the BTF but also the nature of the traffic. We validate the accuracy of our proposed approach against ns-3 simulations under several scenarios.

*Keywords:* IEEE 802.11 networks, aggregated MAC protocol data unit (A-MPDU), Performance evaluation, Network load, Markov chain

---

## 1. Introduction

As one of the cornerstones for delivering flexibility and high speeds, Wireless-Fidelity (Wi-Fi) or IEEE 802.11 has become the most used technology for accessing the Internet. Nearly, all companies, universities, and homes use Wi-Fi to satisfy their connectivity needs, which has led to the explosion of wireless data usage and the colossal rise of Access Points (APs), smartphones, and various mobile devices. According to [1], Internet users will exceed 5.3 billion by 2023, and the number of public Wi-Fi hotspots will grow to nearly 628 million.

---

\*Corresponding author

Email addresses: `nour-el-houda.bouzouita@ens-lyon.fr` (Nour El Houda Bouzouita), `anthony.busson@ens-lyon.fr` (Anthony Busson), `herve.rivano@inria.fr` (Herve Rivano)

In dense environments, a device seeking connectivity must choose among all the APs that are within its radio range. This association is initiated by the device's operating system based on the best Received Signal Strength Indicator (RSSI) or the user's history or preferences. RSSI or equivalent metrics may help to estimate the physical transmission rate that will be used. But, these simple criteria do not take into account the load of the channels, APs, or any information relative to the network performance imposed by the other stations and the current traffic. If the network is managed by one authority, it is possible to implement specific functionalities such as the IEEE 802.11k/v amendment to obtain load estimations and, thus, distribute association with respect to load balancing objectives [2]. When the APs are managed by non-cooperative authorities or when these kinds of functionalities are not available, the station cannot obtain load estimation from the network and should infer it autonomously.

Aggressive achievable throughput tests such as Ookla speedtest and iperf have been widely adopted for network performance measurements such as TCP throughput. While throughput tests provide valuable insights into network state, they suffer from high and intrusive generated traffic and dependence on transport and application protocol. Thus, making them a poor choice for performance characterization.

In contrast to these metrics, the network load seems to be an excellent fit since it conditions the throughput and gives a broader view of the network state independently of the transmission conditions of the device. This latter can be expressed in many ways. In this work, we consider the Busy Time Fraction (BTF), defined as the fraction of time the wireless medium is sensed busy due to successful or unsuccessful transmissions. It captures concurrent transmissions, constituting the AP load, as well as inter-network interference.

This paper aims at studying the possibility to infer the BTF of the channels used by the AP in its vicinity locally in the user space of a non-rooted vanilla device, typically a smartphone, without modifications on network equipment (STAs and APs). We consider an active probing approach (a probe traffic is sent by a client to a server) to estimate this quantity between two devices connected to the same AP. A possible future application of this work (which is not considered in this paper) is to share these measurements publicly available to build a collective knowledge allowing other devices to choose their AP with respect to their loads and produce their own measures.

Classical channel load estimation tools are based on time dispersion but cannot be used in Wi-Fi [3, 4]. Indeed, in the most modern Wi-Fi versions (802.11n, ac, ax, and the most recent roll-up [5]), a frame aggregation scheme has been introduced. This scheme allows an AP or a station (STA) to send several frames through a unique access to the medium and thus making inter-packet arrival null. Hopefully, the load of the channel conditions the aggregation level, defined as the number of packets within the same aggregated frame.

The contributions of our paper are as follows:

- We propose analytical models based on Markov chains that demonstrate

how one could induce the theoretical frame aggregation levels of targeted probe traffic sent from a client to a server concurrent to the cross traffic (the competing traffic already present in the network).

- We reveal how the frame aggregation scheme embodies a rich set of Wi-Fi link properties that can be utilized to discern the network load. We illustrate through ns-3 simulations that frame aggregation levels correlate extremely well with the expected BTF across a variety of challenging scenarios. Moreover, we demonstrate that the throughput that a joining device could get depends on whether the competing traffic uses frame aggregation or not.
- We propose a method called Frame Aggregation based Method (FAM) that leverages both the theoretical frame aggregation levels (returned by the analytical models) and the measured ones to estimate not only the BTF but also the nature of the traffic (aggregated or not) which makes it possible to evaluate the available bandwidth that another device would have if it would join the network.

The proposed method constitutes a proof of concept that shows that the load is inferable locally when the server is close to the client, without any control on the AP. It is worth noting that it is not the definitive application and must potentially be supplemented by other methods depending on the context.

The paper is organized as follows. Section 2 describes in detail the background of this work. Section 3 presents our proposed analytical models to estimate the DownLink frame aggregation levels of the probe traffic and their validation. Section 4 exposes the proposed method. Section 5 is dedicated to the performance evaluation and numerical results of the method. Papers that relate to the studied problem are discussed in Section 6. We conclude this paper in Section 7.

## 2. Background

Before plunging into the details of our proposed system, we present several key background concepts: frame aggregation scheme and how the AP queue handles packets under the frame aggregation.

### 2.1. Frame aggregation overview

Ever since their first introduction in 1997, IEEE 802.11 network has considerably enhanced its throughput by coming with new technological advances to improve the physical (PHY) and the Medium Access Control (MAC) layers. At the MAC layer, one of the significant steps forward was the introduction of the frame aggregation scheme by the IEEE 802.11n standard. Since then, this latter has become the default mechanism of sending data on modern Wi-Fi (802.11n, ac, ax).

The general concept of frame aggregation is to allow several frames, that are

destined to the same destination, to be clustered and sent out as one aggregated frame, and thus reducing the access time overhead (Distributed Inter Frame Space (DIFS), Back-off, etc.) and the PHY overhead as all the aggregated sub-frames share one PHY header. Frame aggregation can be achieved at two levels: at the Aggregate Mac Protocol Data Unit (A-MPDU) and the Aggregate MAC Service Data Unit. Since A-MSDU is rarely implemented, we focus on the A-MPDU aggregation.

Figure 1 depicts the structure of an aggregated frame. The A-MPDU aggregation consists in sending several sub-frames (A-MPDU) with a common Wi-Fi physical header (PHY Header). Each sub-frame is composed of an MPDU delimiter, followed by the MPDU, consisting of its own MAC header, MAC payload (a packet or any layer 3 Protocol Data Unit), and Frame Check Sequence (FCS). Padding bytes may be appended at the end.

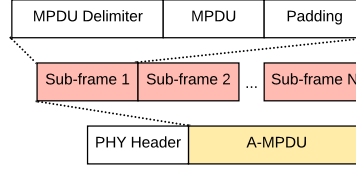


Figure 1: Illustration of A-MPDU frame aggregation.

## 2.2. AP queuing system under frame aggregation

Without frame aggregation and for a given enhanced distributed channel access (EDCA) class (an extension of the Distributed Coordination Function scheme that includes Quality of Service traffic prioritization for data packets), the AP pursues a predefined per-packet scheduling method (e.g., First In First Out (FIFO)). It is important to note that building a queuing algorithm is up to product designers that decide how to manage a transmission queue.

With frame aggregation, the aggregated transmissions act as a batch scheduler which alters the timing characteristics of received packets. To explain this approach, we show an example in Figure 2. It consists of an AP queue with two interleaved sequences of packets sent to two different clients, A and B, on a Wi-Fi link.  $A_i$  defines the  $i^{th}$  packet targeted to client A while  $B_i$  is the  $i^{th}$  packet destined to client B. In the case depicted in Figure 2, the packet at the head of the AP queue is addressed to A. All A's packets (within the limit of the maximum number of aggregated sub-frames) will be aggregated and sent out as an A-MPDU. Then, an aggregated frame with B's packets is sent. In this paper, we consider this approach in order to build the first Markovian model (detailed in Section 3).

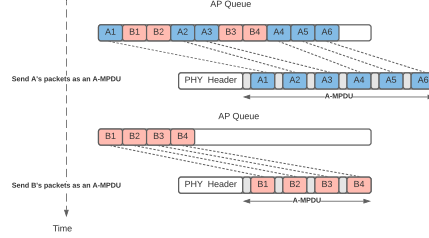


Figure 2: An Example of scheduling under frame aggregation.

### 3. Analytical models and their validation

In this section, we describe the proposed analytical models that help infer the frame aggregation behavior of the probe traffic as a function of the competing traffic. This competing traffic, called cross traffic, can aggregate its frames or not.

#### 3.1. System model description

We consider a general Wireless Local Area Network (WLAN) based on IEEE 802.11 Distributed Coordination Function (DCF) composed of two different entities: an AP and a user station. The station is assumed to be associated with the AP. Each station or AP has its own physical transmission rate. It is a combination of the following parameters: the modulation type and the coding rate which is defined by the Modulation and Coding Scheme index (MCS), the number of spatial streams, the channel width, and the guard interval length. The two proposed analytical models are Discrete-Time Markov chains (DTMC). They appraise the frame aggregation levels of the DownLink probe traffic for a given cross traffic load. As the probe frame aggregation levels depend on the nature of cross traffic (aggregated or non-aggregated traffic), the first chain considers that the cross traffic uses the frame aggregation scheme, whereas the second is based on non-aggregated cross traffic. Table 1 summarizes the principal notations used in the models. Our models rely on the following assumptions:

- **probe traffic:** we distinguish two types of probe traffics: 1) the UpLink (UL) probe traffic which is the traffic sent from the probe traffic node to the AP. It is a Constant Bit Rate (CBR) traffic, sent at regular interval  $d_p$ ; and 2) the DownLink (DL) probe traffic forwarded to the probe server. Frame aggregation scheme is always enabled for this flow.
- **Cross traffic:** the cross traffic is modeled by a CBR source sending packets at regular interval  $d_c$  managed by a unique queue. The cross traffic can be either aggregated or not. This flow is coming from the distribution system (a wired network connected to the AP, which is not represented in Figures 3 and 5) and sent to the cross traffic node.

Table 1: Principal notations

Parameters (unit)	Definition
$T_{DIFS} (\mu s)$	Distributed Inter Frame Space duration
$T_{SIFS} (\mu s)$	Short Inter Frame Space duration
$T_{PHY} (\mu s)$	The preamble and PHY header duration
$FCS (Bytes)$	Frame Check Sequence
$T_{BlockACK} (\mu s)$	Required time to send the block acknowledgment
$T_{backoff} (\mu s)$	Average backoff time
$T_{slot} (\mu s)$	Slot time
$CW_{min}$	Minimum size of the contention window
Models notation	Definition
$P_{(i,j,k,u)(l,m,q,v)}$	Transition probability from state $(i, j, k, u)$ to state $(l, m, q, v)$
$T_{AP}(k) (\mu s)$	Required time to send k DL probe aggregated packets
$T_{SP}(k) (\mu s)$	Required time to send k UL probe aggregated packets
$T_{AC}(K) (\mu s)$	Required time to send k cross traffic packets by the AP
$T_{AC} (\mu s)$	Required time to send a single cross traffic frame
$d_c (\mu s)$	inter-arrival time of cross traffic packets
$d_p (\mu s)$	Inter-arrival time of probe traffic packets
$AMPDU_{AP}$	Maximum A-MPDU size for the AP
$AMPDU_P$	Maximum A-MPDU size for the probe traffic node

- **buffers:** the probe traffic node, the cross traffic node, and the AP buffers are assumed to have a finite size. More precisely, we assume that when an aggregated frame is sent for a given destination, the corresponding buffer becomes empty for this destination. This assumption simplifies drastically the models and their computation. However, the simulation results show that it does not introduce a significant error.

The cross and probe traffics are modeled as CBR. The two DTMC are embedded chains taken at the end of each transmission. The chains describe the number of frames in the different buffers (AP, cross traffic node, probe traffic node) and the current transmission (which node is transmitting). A new state is then a function of the transmission duration. This time sets the number of frames arriving in the different buffer (as traffics are CBR). To keep the model tractable, we neglect the remaining time for the next arrival after the end of a transmission. The new state also depends on the node that wins access to the medium for the next transmission. As we neglect collisions, the successive accesses are independent. With these assumptions, a new state depends only on the previous one and the chains are Markovian.

### 3.1.1. Model based on aggregated cross traffic

Figure 3 depicts the scenario considered in the model based on aggregated cross traffic. An AP and three stations operate on the same Wi-Fi channel. The AP is connected to a distribution system.

We now derive a mathematical model that estimates the frame aggregation levels of the DL probe traffic when we consider an aggregated cross traffic. The model aims to be compared with the aggregation levels as measured by the probe traffic server to deduce the cross traffic load (load of the AP or channel). The frame aggregation level of the DL probe traffic depends on the competition



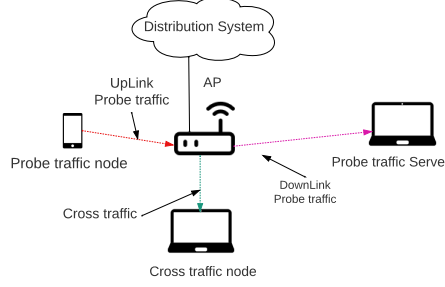


Figure 3: Aggregated cross traffic scenario. The probe traffic node and server estimate the current load. The cross traffic represents the current load of the system.

for medium access. In particular, when the cross traffic increases, the DL probe traffic has to wait longer. More packets may hence accumulate in the queue of the AP and be aggregated when the next DL probe transmission occurs.

We consider the Markov chain defined as  $(X_n, Y_n, Z_n, S_n)_{n \geq 0}$ . The process  $X_n$  describes the number of probe packets contained in the AP queue at the moment of the  $n^{th}$  frame transmission departure. The possible states are  $\{0, \dots, AMPDU_{AP}\}$ .  $Y_n$  defines the number of cross traffic packets in the AP queue. The possible states are  $\{0, \dots, AMPDU_{AP}\}$ . The process  $Z_n$  represents the number of packets at the probe traffic client. The possible states are  $\{0, \dots, AMPDU_P\}$ .  $S_n$  describes the  $n^{th}$  transmission. It takes three possible values  $\{APC, APP, SP\}$ .  $APP$  (Access Point Probe) is a DL transmission of probe traffic from the AP.  $APC$  (Access Point Cross) denotes a cross traffic transmission from the AP to the cross server, while  $SP$  (Station Probe) corresponds to a UL transmission from the probe traffic client.

*Transition probabilities.* Having defined the set of possible states for each process, we shall now derive the transition probabilities. The transition probabilities are denoted  $P_{(i,j,k,u)(l,m,q,v)}$ , and represent the probability to go from state  $(X_n, Y_n, Z_n, S_n) = (i, j, k, u)$  to state  $(X_{n+1}, Y_{n+1}, Z_{n+1}, S_{n+1}) = (l, m, q, v)$ . For convenience, we also define the following function where  $A$  and  $B$  take their values in the set  $\{APC, APP, SP\}$ :

$$\beta_{(B,l,m,q)}(A) = P(S_{n+1} = APC | S_n = APP, X_{n+1} = l, Y_{n+1} = m, Z_{n+1} = q)$$

The transition probabilities depend on the time between two successive transmissions. As both UL probe and UL cross traffics are deterministic, this time sets the number of packets that arrived in the AP buffer and the probing client buffer between two transmissions and thus the number of frames that will be sent in the next aggregated frame. We therefore analyze the events that

may occur between two successive transmissions. The next stage of our modeling is to decide when a transition from one state to another is allowed and to compute its probability. Note that impossible transitions have zero probability and that non-null transition probabilities are computed assuming that all concerned nodes are equally likely to access the channel. For ease of illustration, we categorize the state transitions into the following three classes.

**Class I: Transition from state APP.** For the sake of clarity, we recourse to Figure 4a to describe the possible transition probabilities and their computations. It illustrates an example of possible events between two successive transmissions when the current state is APP (an aggregated frame containing probe packets is transmitted by the AP). In this example, at time  $t = 0$ , we start from state  $(i, j, k, APP)$  (i.e. (3,2,2,APP) in Figure 4a). The AP sends the  $i$  corresponding probe frames currently in its buffer ( $i = 3$  in this example), aggregating them in a single A-MPDU using its current MCS. Its transmission duration is denoted  $T_{AP}(i)$  which is given in [6]. Note that the AP has a single buffer that contains at once the probe and the cross traffics but in order to clarify the explanation of the model, we distinguish between them in Figure 4a.

During this transmission, the probe traffic node buffer receives  $\lfloor \frac{T_{AP}(i)}{d_p} \rfloor$  packets and the AP buffer receives  $\lfloor \frac{T_{AP}(i)}{d_c} \rfloor$  cross packets. As a result, at the end of the APP state, the probe traffic node buffer and the AP will contain respectively,  $k + \lfloor \frac{T_{AP}(i)}{d_p} \rfloor$  of probe packets and  $j + \lfloor \frac{T_{AP}(i)}{d_c} \rfloor$  of cross traffic packets ((4,4) in the Figure 4a).

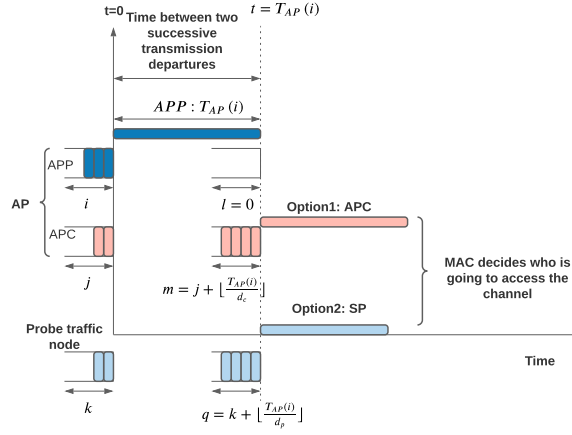
At the end of this transmission, the buffer of the AP does not have probe frames to send ( $X_{n+1} = 0$  almost surely), and another APP transmission is impossible ( $S_{n+1} \neq APP$  almost surely). So from this state, only two transitions are allowed: to APC or SP with  $X_{n+1} = l = 0$ . It can occur only if  $Y_{n+1} = m > 0$  and  $Z_{n+1} = q > 0$  respectively.

We derive the non-null transition probabilities as follows. If the AP gains access to the channel, it will send the cross traffic currently in its buffer. For  $m > 0$ , the next transmission will be APC with probability:

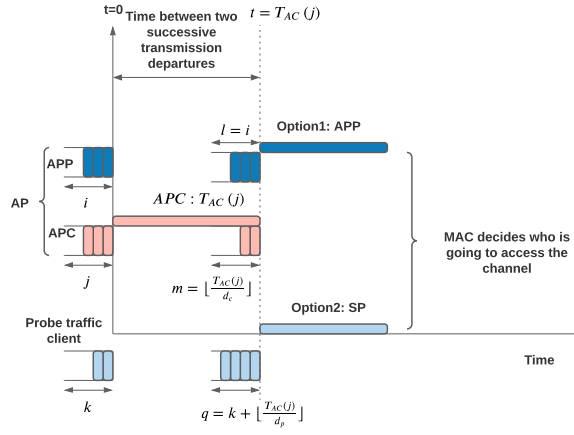
$$P_{(i,j,k,APP)(0,m,q,APC)} = \beta_{(APP,0,m,q)}(APC) \cdot \mathbb{1}_{k+\lfloor \frac{T_{AP}(i)}{d_p} \rfloor=q} \cdot \mathbb{1}_{j+\lfloor \frac{T_{AP}(i)}{d_c} \rfloor=m} \quad (1)$$

where  $\mathbb{1}_{condition}$  is the indicator function that equals to 1 if *condition* is true and 0 otherwise.  $P(S_{n+1} = APC | S_n = APP, X_{n+1} = l = 0, Y_{n+1} = m > 0, Z_{n+1} = q)$  denotes the probability that the event APC will occur given that the event APP has already occurred. The computation of these probabilities is detailed in [6]. Now, if the probe station gains access, the next event will be SP and the transition probability from APP to SP for  $q > 0$  is given by:

$$P_{(i,j,k,APP)(l,m,q,SP)} = \beta_{(APP,0,m,q)}(SP) \cdot \mathbb{1}_{k+\lfloor \frac{T_{AP}(i)}{d_p} \rfloor=q} \cdot \mathbb{1}_{j+\lfloor \frac{T_{AP}(i)}{d_c} \rfloor=m} \quad (2)$$



(a) Transition from APP



(b) Transition from APC

Figure 4: Possible transitions

**Class II: Transition from state APC.** Once again, when presenting the transition probabilities of this class, we resort to Figure 4b. It depicts a timeline of feasible events between two successive transmission departures when we start from the state APC.

The AP sends the  $j$  cross traffic frames currently in its buffer as an A-MPDU using the transmission rate associated to the cross traffic server. Its transmission duration is denoted  $T_{AC}(j)$ . During the period  $T_{AC}(j)$ , the probe traffic client buffer receives  $\lfloor \frac{T_{AC}(j)}{d_p} \rfloor$  packets and the AP buffer receives  $\lfloor \frac{T_{AC}(j)}{d_c} \rfloor$  cross traffic packets. At the end of the APC state, the probe traffic client buffer and the AP will thus contain respectively,  $k + \lfloor \frac{T_{AC}(j)}{d_p} \rfloor$  of probe packets and  $\lfloor \frac{T_{AC}(j)}{d_c} \rfloor$  of cross traffic packets.

For this class of transitions, there are three possible transitions:  $S_{n+1}$  can be APP, APC, or SP. First, we suppose that there will be another APC, the transition from APC to APC is deemed possible if and only if  $X_n = i = 0$  and  $Y_n = l = 0$ : it is impossible to have two successive APC transmissions if  $i > 0$  or  $l > 0$  due to the AP queuing system detailed in Section 2.2. The transition probability from APC to APC with  $Y_{n+1} = m > 0$  is:

$$P_{(0,j,k,APC)(0,m,q,APC)} = \beta_{(APC,0,m,q)}(APC) \cdot \mathbb{1}_{\lfloor \frac{T_{AC}(j)}{d_c} \rfloor = m} \cdot \mathbb{1}_{k + \lfloor \frac{T_{AC}(j)}{d_p} \rfloor = q} \quad (3)$$

Now if we suppose that the AP gains the medium access to transmit the DL probe flow, the transition probability from APC to APP is possible only if  $X_n = i = X_{n+1} = l$  since the AP cannot receive probe frames in its buffer during the transmission of the cross traffic and  $X_{n+1} = l > 0$ . The corresponding probability is given by:

$$P_{(i,j,k,APC)(l,m,q,APP)} = \beta_{(APC,l,m,q)}(APP) \cdot \mathbb{1}_{\lfloor \frac{T_{AC}(j)}{d_c} \rfloor = m} \cdot \mathbb{1}_{k + \lfloor \frac{T_{AC}(j)}{d_p} \rfloor = q} \quad (4)$$

If we assume that the probe traffic client gains the medium access, the transition from APC to SP is also allowed only if  $X_{n+1} = l = X_n = i$  and  $Z_{n+1} = q > 0$ .

$$P_{(i,j,k,APC)(l,m,q,SP)} = \beta_{(APC,l,m,q)}(SP) \cdot \mathbb{1}_{\lfloor \frac{T_{AC}(j)}{d_c} \rfloor = m} \cdot \mathbb{1}_{k + \lfloor \frac{T_{AC}(j)}{d_p} \rfloor = q} \quad (5)$$

**Class III: Transition from state SP.** In order to derive the transition probabilities of the last class, we apply the same principle. Due to space limitations, we do not describe the details of the derived transition probabilities.

From the SP state, there are three possible transitions APP, APC, or SP depending on the competition for the channel resource. First, if the next transition is APP with  $X_{n+1} = l > 0$ , the transition probability is given by:

$$P_{(i,j,k,SP)(l,m,q,APP)} = \beta_{(SP,l,m,q)}(APP) \cdot \mathbb{1}_{l=i+k} \cdot \mathbb{1}_{j + \lfloor \frac{T_{SP}(k)}{d_c} \rfloor = m} \cdot \mathbb{1}_{\lfloor \frac{T_{SP}(k)}{d_p} \rfloor = q} \quad (6)$$

Second, if the next transition is APC with  $Y_{n+1} = m > 0$ , the transition probability is defined as:

$$P_{(i,j,k,SP)(l,m,q,APC)} = \beta_{(SP,l,m,q)}(APC) \cdot \mathbf{1}_{l=i+k} \cdot \mathbf{1}_{j+\lfloor \frac{T_{SP}^{(k)}}{d_c} \rfloor = m} \cdot \mathbf{1}_{\lfloor \frac{T_{SP}^{(k)}}{d_p} \rfloor = q} \quad (7)$$

Finally, if the next transition is SP with  $Z_{n+1} = q > 0$ , the transition probability is thus formulated as:

$$P_{(i,j,k,SP)(l,m,q,SP)} = \beta_{(SP,l,m,q)}(SP) \cdot \mathbf{1}_{l=i+k} \cdot \mathbf{1}_{j+\lfloor \frac{T_{SP}^{(k)}}{d_c} \rfloor = m} \cdot \mathbf{1}_{\lfloor \frac{T_{SP}^{(k)}}{d_p} \rfloor = q} \quad (8)$$

### 3.1.2. Model based on aggregated cross traffic

Let us now derive the second Markov chain where the frame aggregation scheme is disabled for the cross traffic.

Figure 5 breaks down the network topology simulated by this model that consists of two co-located WLANs. The first one enable the frame aggregation scheme (e.g., IEEE 802.11n in Figure 5) and the second one disable the frame aggregation (e.g., IEEE 802.11g). The WLAN where aggregation is enabled is composed of an AP and two nodes. A probe traffic node to send the UL probe traffic, and a server node to receive the forwarded DL traffic. The other WLAN is composed of an AP and a node that receives the non-aggregated cross traffic.

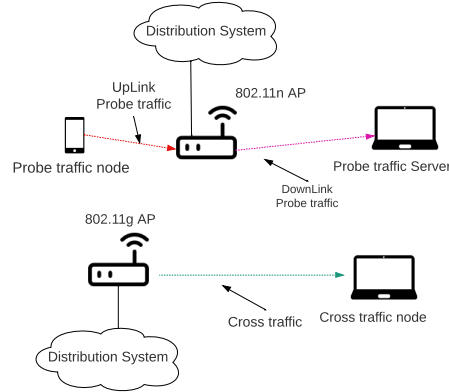


Figure 5: Scenario modeled by the model based on non-aggregated cross traffic.

The rationale behind this model is the same as the model described in the previous section except that each cross traffic transmission consists only of a single frame. Like the aggregated version of this model, we divide the state transitions into the following three classes for the non-aggregated version as follows.

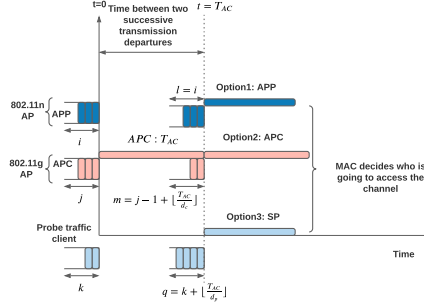


Figure 6: Possible events between two successive transmissions for the *Wireless server* model based on non-aggregated cross traffic when the current transmission is APC.

**Class I: Transition from APP.** We first consider the transition probabilities when the current transmission is APP. Since during this transmission, the 802.11n AP sends an aggregated probe frame to the probe traffic server, the transitions probabilities from state APP are the same. The computation of the conditional probabilities related to these three classes are given in [6]. If the 802.11g AP gains access to the channel, the transition probability from state APP to state APC with  $Y_{n+1} = m > 0$  is:

$$P_{(i,j,k,APP)(0,m,q,APC)} = \beta_{(APP,0,m,q)}(APC) \cdot \mathbb{1}_{k + \lfloor \frac{T_{AP}^{(i)}}{d_p} \rfloor = q} \cdot \mathbb{1}_{j + \lfloor \frac{T_{AP}^{(i)}}{d_c} \rfloor = m} \quad (9)$$

Now if the probe traffic node gains the competition for the channel resource, the transition from APP to SP with  $Z_{n+1} = q > 0$  is given by:

$$P_{(i,j,k,APP)(l,m,q,SP)} = \beta_{(APP,0,m,q)}(SP) \cdot \mathbb{1}_{k + \lfloor \frac{T_{AP}^{(i)}}{d_p} \rfloor = q} \cdot \mathbb{1}_{j + \lfloor \frac{T_{AP}^{(i)}}{d_c} \rfloor = m} \quad (10)$$

**Class II: Transition from APC.** We consider the transition probabilities when the current transmission is APC. When presenting the probabilities of this class, we resort to Figure 6. It exposes a set of possible events between two successive transmission departures when we start from the state APC.

The AP for which aggregation is disabled sends a single frame from the  $j$  cross traffic frames currently in its buffer. Its transmission duration is denoted  $T_{AC}$ . During this period, the probe traffic node receives  $\lfloor \frac{T_{AC}}{d_p} \rfloor$  of probe packets, and the AP receives  $\lfloor \frac{T_{AC}}{d_c} \rfloor$  of cross traffic. Here, there are three possible transitions. The process  $S_{n+1}$  can be APP, APC, or SP. It should be noted that all the following transitions are deemed possible if  $X_n = i = X_{n+1} = l$  since the AP that aggregates cannot receive probe frames in its buffer during the

transmission of the cross traffic by the other AP. If we assume that there will be another APC, the corresponding probability with  $Y_{n+1} = m > 0$  is defined as follows:

$$P_{(i,j,k,APC)(i,m,q,APC)} = \beta_{(APC,i,m,q)}(APC) \cdot \mathbb{1}_{j-1+\lfloor \frac{T_{AC}}{d_c} \rfloor = m} \cdot \mathbb{1}_{k+\lfloor \frac{T_{AC}}{d_p} \rfloor = q} \quad (11)$$

315 The transition probability from APC to APP with  $X_{n+1} = i > 0$  is given by:

$$P_{(i,j,k,APC)(i,m,q,APP)} = \beta_{(APC,i,m,q)}(APP) \cdot \mathbb{1}_{j-1+\lfloor \frac{T_{AC}}{d_c} \rfloor = m} \cdot \mathbb{1}_{k+\lfloor \frac{T_{AC}}{d_p} \rfloor = q} \quad (12)$$

Finally, we establish the transition probabilities from APC to SP. This transition is allowed if  $Z_{n+1} = q > 0$  and given by:

$$P_{(i,j,k,APC)(l,m,q,SP)} = \beta_{(APC,i,m,q)}(SP) \cdot \mathbb{1}_{j-1+\lfloor \frac{T_{AC}}{d_c} \rfloor = m} \cdot \mathbb{1}_{k+\lfloor \frac{T_{AC}}{d_p} \rfloor = q} \quad (13)$$

320 **Class III: Transition from SP.** From the state SP, there are also three possible transitions APP, APC, or SP depending on the competition for the channel resource. First, if the next transition is APP with  $X_{n+1} = l > 0$ , the transition probability is given by:

$$P_{(i,j,k,SP)(l,m,q,APP)} = \beta_{(SP,l,m,q)}(APP) \cdot \mathbb{1}_{l=i+k} \cdot \mathbb{1}_{j+\lfloor \frac{T_{SP}(k)}{d_c} \rfloor = m} \cdot \mathbb{1}_{\lfloor \frac{T_{SP}(k)}{d_p} \rfloor = q} \quad (14)$$

Second, if the next transition is APC with  $Y_{n+1} = m > 0$ , the transition probability is formulated as:

$$P_{(i,j,k,SP)(l,m,q,APC)} = \beta_{(SP,l,m,q)}(APC) \cdot \mathbb{1}_{l=i+k} \cdot \mathbb{1}_{j+\lfloor \frac{T_{SP}(k)}{d_c} \rfloor = m} \cdot \mathbb{1}_{\lfloor \frac{T_{SP}(k)}{d_p} \rfloor = q} \quad (15)$$

325 Finally, if the next transition is SP with  $Z_{n+1} = q > 0$ , the transition probability is thus formulated as:

$$P_{(i,j,k,SP)(l,m,q,SP)} = \beta_{(SP,l,m,q)}(SP) \cdot \mathbb{1}_{l=i+k} \cdot \mathbb{1}_{j+\lfloor \frac{T_{SP}(k)}{d_c} \rfloor = m} \cdot \mathbb{1}_{\lfloor \frac{T_{SP}(k)}{d_p} \rfloor = q} \quad (16)$$

### 3.2. Stationary probability

330 The last phase consists in deriving the stationary probability and computing the frame aggregation levels. Let us remind that a Markov chain is irreducible if and only if every state can be reached by any other state through one or several transitions. As our Markov chain is irreducible and has a finite number of states, hence it exists a unique stationary distribution. We solve this Markov chain through a numerical method. We compute the stationary probabilities and we denote by  $\pi$  the vector containing the corresponding values. The mean DL probe traffic aggregation levels, denoted *MeanAgg* are then given by:  $MeanAgg = \sum_{n=1}^{AMPDU_{AP}} n \cdot \pi_n$ .

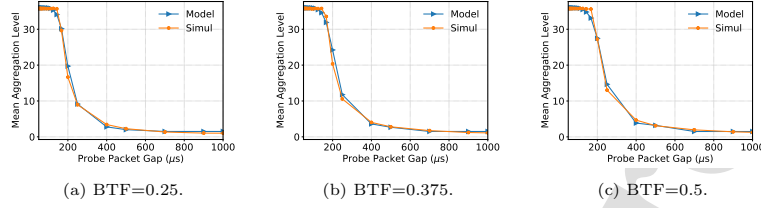


Figure 7: Mean aggregation levels, aggregated cross traffic, model versus NS-3 - MCS 15.

### 3.3. Numerical results

In this section, we evaluate the accuracy of the proposed models in estimating the frame aggregation levels of the probe traffic. We do so under several scenarios with different network parameters, such as the topology and its size, different IEEE 802.11 amendments, and different traffic patterns. In this regard, we compare the aggregation levels given by the models with those delivered by the discrete-event network simulator 3 (ns-3 version 3.30). For simplicity reasons, we assume for all the scenarios that all nodes have a random but fixed position during the whole simulation duration. We compute the two models for six cross traffic loads/BTF levels: 0, 0.125, 0.25, 0.375, 0.5, and 0.625 ranging from low to high levels of BTF. The maximum numbers of aggregated packets  $AMPDU_P$  and  $AMPDU_{AP}$  are set to 36. Our Markov models are solved through simulation. We start by fixing the initial state (the number of packets in each buffer and the first possible transition, e.g., APP) and the number of iterations (10000 in our case). Based on the buffer states and the computed transitions probabilities, we then ran the model in order to compute the probe mean aggregation levels.

#### 3.3.1. Validation of the model based on aggregated cross traffic

We examine the accuracy of the first model where the frame aggregation scheme is enabled for both probe and cross traffics under the 802.11n standard amendment.

**Scenario #1 - Same MCS.** We start by examining the performance details of this model when the same MCS index (HT-MCS15 with a physical transmission rate of 144.4 Mbps) is used for the probe traffic node and the AP. Figure 7 shows the mean aggregation level for the probe traffic as a function of the probe packet gap for the model and simulations. We let the probe packet gap gradually varies from  $50\mu s$  to  $1000\mu s$ . According to these results, it appears that the model performs well since it follows closely the pattern of the ns-3 simulations for the three levels of the network loads (0.25, 0.375, and 0.5). It is also the case for the three other BTFs (0, 0.125, and 0.625) that are not shown here. We can observe that the aggregation levels depend on the loads of the network. Indeed, when the BTF increases, the probe traffic has to wait longer, and more packets are received between two successive probe transmissions.



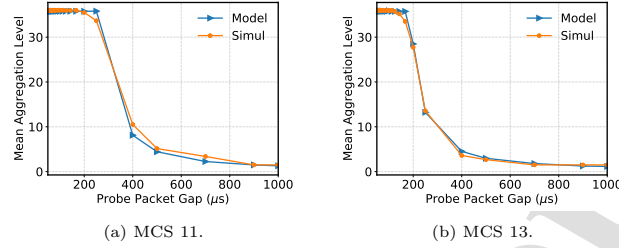


Figure 8: Mean aggregation levels, aggregated cross traffic, model versus NS-3 - BTF = 0.375.

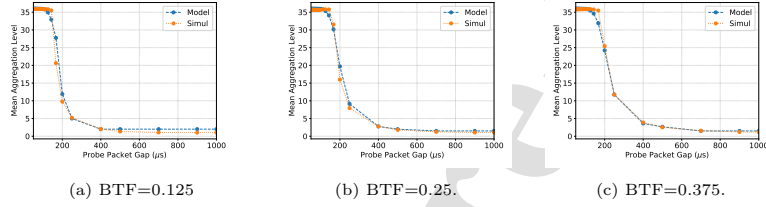


Figure 9: Mean agg levels, agg cross traffic, model vs NS-3 - Exp On/Off cross traffic.

**Scenario #2 - Different MCS indexes.** We now evaluate the performance details of the first model when different MCS indexes are used for the probe traffic node and the AP. Figure 8 shows the corresponding results. Figure 8a plots the results when we use the MCS 15 (144.4 Mbps) for the AP to send the DL probe traffic and the cross traffic and the MCS11 (57.8 Mbps) for the probing node. Figure 8b provides the corresponding results when we use MCS15 for the AP and MCS13 (115.6 Mbps) for the probe traffic node. The results show that considering different MCS indexes for the AP and the probe traffic node does not impact the accuracy of our approach.

**Scenario #3 - Exponential On/Off cross traffic.** To provide a broader overview of the accuracy reached by the proposed modeling approach, this scenario simulates another traffic pattern for the cross traffic which is exponential On/Off traffic, reflecting some of the kinds of cross traffic that would occur in practice. In ns-3, we use exponentially distributed On/Off periods thanks to the ns-3 OnOff Application class. This latter mainly relies on an *OffTime* and *OnTime* duration attributes that represent respectively the duration during which the data transfer is switched off and the duration of the continuous data transfer. Figure 9 shows the mean aggregation levels for the BTFs 0.125, 0.25 and 0.375. These results show that the estimations made by our model fit those delivered by ns-3 for the exponential On/Off aggregated cross traffic. We conclude that the proposed models still perform well under exponential On/Off traffic pattern.

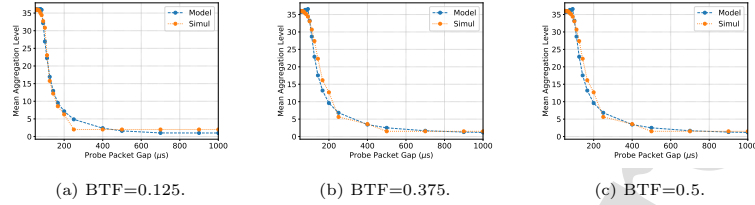


Figure 10: Mean agg levels, agg cross traffic, model vs NS-3 - 802.11ax traffic.

**Scenario #4 - 802.11ax traffic.** Here, we run the first model when both the probe and cross traffics use the 802.11ax standard. All the nodes use the data rate=286.8Mbps. It is noticeable from Figure 10 that the mean aggregation levels obtained with simulations fit the ones from the model.

#### 3.4. Validation of the model based on non-aggregated cross traffic

Extensive comparisons between the model based on non-aggregated cross traffic under several scenarios and simulations are detailed in [6]. The model is able to reproduce the DL probe traffic aggregation behavior with a satisfying degree of precision. The number of stations composing the DL competing traffic coming from the same AP does not influence the estimations for the model.

Figures 11a and 11b show the probe mean aggregation levels for six BTFs depending on the frame aggregation scheme. Varying the level of cross traffic barely affects the probe aggregation levels when the  $BTF > 0.25$ . In this case, each cross traffic frame is sent independently, with short transmission times. Consequently, the probe traffic receives less packets to aggregate between two consecutive medium accesses. Also, cross traffic reaches saturation faster (as it sends less frames on average). As soon as it has always a frame to send, its access time does not depend on its buffer state. Consequently, the aggregation level of the probe traffic becomes insensitive to the level of congestion of the cross traffic. On the contrary (Figure 11a), when cross traffic aggregates its frames, the state of its buffer has a deeper impact on probe traffic aggregation since the cross traffic buffer state determines the transmission duration. Indeed, varying the level of the aggregated network load significantly affects the probe aggregation levels. Consequently, the results are sufficiently separated to be used to infer the load level (Section 4).

## 4. Description of the method FAM

The results presented in the previous section indicates a strong correlation between the aggregation levels of the probe traffic and the network load. In this section, we hence introduce a network load inference method called Frame Aggregation based Method (FAM). It exploits the rich information embedded in frame aggregation to infer both the BTF of an AP and the type of traffic. Note that no changes are required, neither to the device nor to the AP. The designed

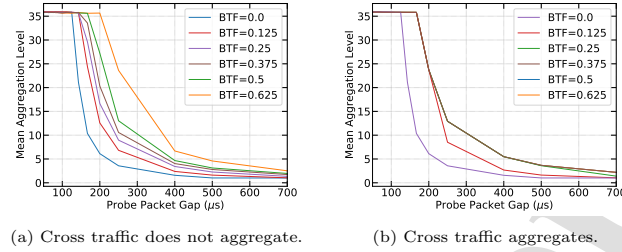


Figure 11: Simulated Mean Aggregation levels depending on cross traffic aggregation.

system aims at achieving this estimation for unmodified mobile devices. The foundation of FAM is mainly built on three steps:

- Measurement of the mean aggregation levels for different traffic flows. (Section 4.1)
- Detection of the type of the cross traffic (Section 4.2)
- Estimation of the wireless channel load (Section 4.3)

We unfold the details of the proposed approach by exposing the algorithmic representation of each stride. Each step relies on one or many algorithms. We note that all the algorithms detailed below are implemented on the server (the client procedure is not shown under its algorithm form).

#### 4.1. Mean aggregation level

Initially, we start with the measurement of the aggregation levels of the probe traffic. Probing nodes (the client) begins by sending successive batches of probe packets. For each batch, the gap interval between packets,  $d_p$ , is increased from  $d_{p_{min}}$  to  $d_{p_{max}}$ . In order to evaluate the mean aggregation levels accurately while keeping the batch size as small a possible, the number of packets in a batch  $np$  is not fixed. This number is determined in the server procedure (Algorithm 1). Upon receipt of this batch, the server responds with a packet that indicates to the probe node to stop sending packets. It is the *convergence* variable in the algorithm.

**Algorithm 1** Server Procedure

---

**Input:**  $E$  : acceptable standard error of the mean,  $Z$  : Value of the distribution function

```

1: MeanAgg= $\emptyset$ , D= $\emptyset$ 
2: while campaign in progress for this client do
3:    $n = 0$ , convergence = False,  $meanAgg = 0$ 
4:   while !convergence do
5:     server.receive(1)  $\triangleright$  Reception of the first packet that contains the parameters  $(d_p, np)$ 
6:     server.receive( $np - 1$ ) and store  $agg_{i+n}$ ,  $\forall i \in \{1, np\}$ 
7:      $n = n + np$ 
8:      $meanAgg = \frac{\sum_{j=1}^n agg_j}{n}$ 
9:      $S^2 = \frac{1}{n-1} \sum_{i=1}^n (agg_i - meanAgg)^2$ 
10:    convergence =  $\left( n \geq \frac{Z^2 \times S^2}{E^2} \right)$ 
11:    server.send(convergence)
12:  end while
13:  Add  $meanAgg$  to vector MeanAgg
14:  Add  $d_p$  to vector D
15:  server.send( $meanAgg$ )
16: end while

```

---

**Output:** (MeanAgg, D)  $\triangleright$  Returns two vectors: the “MeanAgg” and “D”

---

Algorithm 1 runs on the server. It receives and processes the incoming probe packets from the probe traffic node. During a given batch, the mean aggregation is computed on the fly at the reception of probe packets. We use the CLT to evaluate the accuracy of the mean aggregation levels. When the error is lower than the expected error  $E$ , the server sends the *convergence* notification to the probe node (that stops its batch). The use of the CLT allows the application to appropriately measure the mean aggregation level with a minimal amount of time and bandwidth cost.

**4.2. Cross traffic nature**

Having already computed the mean aggregation levels of the probe traffic, we now proceed with the cross traffic type estimation. Our intuition and reasoning to assess the cross traffic nature is based on the idea that, if the cross traffic does not aggregate, its channel access time becomes constant when the channel load increases. This time is defined here as the time the cross traffic uses the channel between two successive probe traffic accesses. It corresponds to  $T_{AC} + T_{AC}$  in Figure 6. The cross traffic access time depends only on the successive number of times it accesses the channel and the time to transmit a single frame. When the two buffers (cross and probe) are non-empty, it becomes independent of the number of frames/packets in these buffers and thus on the probe and traffic loads. On the contrary, when the cross traffic aggregates its frames, this time is increasing with the load as A-MPDU contains more aggregated frames.

We use this observation to detect the nature of the cross traffic. The mean cross traffic access time is denoted  $T_C$ . If this time is constant with  $d_p$  and with the load, we can express the mean aggregation level for the probe traffic,  $meanAgg$ , through a fixed point equation.

$$meanAgg = \min \left( K_{max}, \frac{f(meanAgg)}{d_p} + \frac{T_C}{d_p} \right) \quad (17)$$

The rational of this equation is that the mean number of aggregated frames ( $meanAgg$  on the left-hand side of the equation) is equal to the number of frames received at the buffer since the last probe transmission. In Figure 6, we would get  $X_{n+1} = \frac{T_{AP}(X_n)}{d_p} + \frac{T_{AC}+T_{AC}}{d_p}$ . Substituting  $(X_n, X_{n+1})$  by the mean aggregation ( $X_n = X_{n+1} = meanAgg$ ) and assuming that the cross traffic access time is constant,  $(T_{AC} + T_{AC})$  becomes  $T_C$  for this example), we obtain the Equation 17.

This approach is described in Algorithm 2. It takes as input the mean aggregation level for the probe and computes  $T_C$ , according to equation 17, for each probe packet gap  $d_p$ . It keeps only values of  $T_C$  for which  $\frac{T_{AP}(meanAgg)}{d_p} + \frac{T_C}{d_p}$  is less than  $K_{max}$ . Then, it returns the percentage of variation between the maximum and minimum of these values. In Algorithm 4, this percentage is compared to a given threshold to determine if  $T_C$  can be considered as constant (cross traffic does not aggregate) or not (cross traffic aggregates).

---

**Algorithm 2** Percentage Increase Algorithm

---

**Input:** MeanAgg=( $meanAgg(d_{pmin}), \dots, meanAgg(d_{pmax})$ ),  $D=(d_{pmin}, \dots, d_{pmax})$   
 1:  $TC\_VECTOR = \emptyset$   
 2: **for** all  $d_p \in D$  **do**  
 3:   Compute  $T_C$ :  
 4:    $T_C(d_p) = d_p * meanAgg(d_p) - f(meanAgg(d_p))$   
 5:   **if**  $\frac{f(meanAgg)}{d_p} + \frac{T_C}{d_p} < K_{max}$  **then**  
 6:     Add  $T_C(d_p)$  to vector  $TC\_VECTOR$   
 7:   **end if**  
 8: **end for**  
 9:  $\%increase \leftarrow \frac{max(TC\_VECTOR) - min(TC\_VECTOR)}{min(TC\_VECTOR)} * 100$   
**Output:**  $\%increase$

---

### 4.3. BTF estimation

Having already computed the mean aggregation levels and discussed how to determine the type of the concurrent traffic, we now detail how to estimate the network of a wireless channel via its BTF. Two methods are proposed to estimate the BTF. They are both based on the comparison between the mean aggregation levels given by our Markov chains and the measures made for different probe traffic batches. First, the mean error can be computed as the sum of the difference between the model and the measures as follows.

$$Error_{(bt, case)} = \frac{1}{Size(MeanAgg(.))} \sum_{d_p=d_{pmin}}^{d_{pmax}} |model_{(case, d_p, bt)} - meanAgg(d_p)|$$

The BTF that minimizes this error, denoted in the following as Error computation, is the argmin of the above formula over  $bt$  and is computed for both models, with aggregation ( $w/A$ ) and without frame aggregation ( $wo/A$ ). Alternatively, Algorithm 3 considers a notion of score to infer the BTF. For each probe packet gap  $d_p$  and a given model, the BTF that minimizes the error scores one. For each model, the final BTF is the argmax of this score.

**Algorithm 3** BTF Score Based Method Algorithm

---

**Input:** MeanAgg=( $meanAgg(d_{p_{min}}), \dots, meanAgg(d_{p_{max}})$ ),  $D=(d_{p_{min}}, \dots, d_{p_{max}})$ )  
1: **for**  $d_p \in D$  **do**  
2:   **for each**  $case \in \{w/A, wo/A\}$  **do**  
3:     **for**  $bt \in$  all levels of BTF **do**  
4:        $Error(case, d_p, bt) = \frac{1}{Size(meanAgg(.))} | model(case, d_p, bt) - meanAgg(d_p) |$   
5:     **end for**  
6:   **end for**  
7:    $Score(argmin_{bt, case} Error(case, d_p, bt), case) += 1$   
8: **end for**  
**Output:** ( $argmax_{bt} Score(bt, w/A)$ ,  $argmax_{bt} Score(bt, wo/A)$ )

---

Algorithm 4 gives the final output of our method. First, it evaluates if the BTF is less or equals to 0.25. We have observed that, for such a low load, it is difficult to distinguish precisely the nature of the traffic since the cross traffic tends to have very small mean aggregation levels. In this case, no precision on the aggregation is given. If the BTF is greater than 0.25 at least for one of the two models, it computes the percentage of variation of  $T_C$  with Algorithm 2 and compares it to a threshold  $T$ . It determines if the cross traffic access time can be considered as constant and consequently if the cross traffic aggregates or not. For both cases, the corresponding BTF is returned with the cross traffic nature.

**Algorithm 4** BTF Estimation Algorithm

---

**Input:** T:Threshold, E: acceptable standard error of the mean  
1: (MeanAgg, D) = Server Procedure(E, Z)  
2:  $PI = PercentageIncrease(MeanAgg, D)$   
3:  $BTF_{w/A}, BTF_{wo/A} = Error\ Computation(MeanAgg, D)$   
4:  $BTF1_{w/A}, BTF1_{wo/A} = Score\ Computation\ Procedure(MeanAgg, D)$   
5: **if** ( $BTF_{w/A} \leq 0.25$  or  $BTF1_{w/A} \leq 0.25$ ) and ( $BTF_{wo/A} \leq 0.25$  or  $BTF1_{wo/A} \leq 0.25$ ) **then**  
6:   **return**  $\{wo/A, BTF \leq 0.25\}$   
7: **else if**  $0 < PI < T$  **then**  
8:   **return**  $\{wo/A, BTF > 0.25\}$   
9: **else**  
10:   **return**  $\{w/A, BTF_{w/A}\}$   
11: **end if**

---

**5. Performance Evaluation**

In this section, we evaluate the accuracy of the proposed method FAM by relying mainly upon the same ns-3 scenarios used in the Section 3 that expose different network parameters, different IEEE 802.11 amendments, and different traffic patterns. The parameters of our method are as follow. Similar to the analytical models, we compute the four Markovian models for six cross traffic loads/BTF levels: 0, 0.125, 0.25, 0.375, 0.5, and 0.625 ranging from low to high levels of BTF. In algorithm 1, the FAM method uses a 95% confidence interval ( $Z = 1.96$ ) and an error  $E = 0.05$ . The CLT results indicated that the relevant number of packets to send increases proportionally to the probe rate. The number of sent packets ranges between 600 and 3800. In algorithm 4, the threshold  $T$  is chosen according to an empirical method. Our simulations and

experiments have shown that a value of  $T = 200\%$  offers good performances. For instance, we take different values for the parameter  $T$  that depend on the device's transmission rate. These values are known by FAM.

### 5.1. FAM validation for aggregated cross traffic scenarios

In this section, we assess the effectiveness of FAM when relying on the theoretical aggregation levels returned by the models when the cross traffic uses or not the frame aggregation feature. We want to evaluate the ability of FAM to infer the BTF and the cross traffic type when we consider aggregated cross traffic scenarios.

**Scenario #1 - Same MCS.** We now study the case when all the nodes (AP and STAs) use the same MCS index (MCS15 corresponding to 144.4 Mbps). We reuse the three-STAs scenario of Figure 3. Table 2 presents the estimated BTF and cross traffic nature for the considered scenario. The ground truth column gives the real value of the BTF. In the table, 'S' indicates the BTF and cross traffic nature that were set in the simulations, and 'A' presents the value returned by our algorithm FAM. We note that FAM finds the good result when the two letters ('S'/'A') are in the same box. As FAM does not return the cross traffic nature when the load is lower than 0.25, we set 'S' and 'A' in the two corresponding boxes (aggregated and non-aggregated cross traffic) in this case. Results presented in this table show that the method finds the good results for all the levels of BTF.

Table 2: BTF and cross traffic nature estimations, NS-3, cross traffic is aggregated - MCS 15

Ground Truth BTF	Estimated BTF and cross traffic nature					
	Cross traffic aggregates				Cross traffic does not aggregate	
	$\leq 0.25$	0.375	0.5	0.625	$\leq 0.25$	$> 0.25$
0	S/A				S/A	
0.125	S/A				S/A	
0.25	S/A				S/A	
0.375		S/A				
0.5			S/A			
0.625				S/A		

**Scenario #2 - Different MCS indexes.** Once again, we consider the scenario depicted in Figure 3 to study the accuracy of our approach under nodes (AP and STAs) that have different data rates. We use the MCS15 (144.4 Mbps) for the AP to send the DL probe traffic and the cross traffic and the MCS11 (57.8 Mbps) for the probing node (scenario 1). Then, we use MCS15 for the AP and MCS13 (115.6 Mbps) for the probe traffic node (scenario 2). Table 3 shows the results of scenario 2. We can observe that the method finds the good results for the five BTFs 0, 0.25, 0.375, 0.5, and 0.625 and slightly overestimates the BTF for the case 0.125. This error is only of 0.125 (approximately 10%) and is due to the fact that we neglect some protocol aspects, such as the beacons sent

550 by the AP. We note that we obtain similar results as 3 for the scenario 2. Overall, we notice that despite having STAs with significantly different data rates, the method FAM is still able to deliver reasonable predictions for the BTF.

Table 3: BTF and cross traffic nature estimations for ns3 simulations where the cross traffic is aggregated - Different MCS

Ground Truth BTF	Estimated BTF and cross traffic nature					
	Cross traffic aggregates				Cross traffic does not aggregate	
	$\leq 0.25$	0.375	0.5	0.625	$\leq 0.25$	$> 0.25$
0	S/A				S/A	
0.125	S	A			S	
0.25	S/A				S/A	
0.375		S/A				
0.5			S/A			
0.625				S/A		

555 **Scenario #3 - Exponential On/Off cross traffic.** Instead of taking into account deterministic cross traffic, this scenario considers cross traffic generated using exponentially distributed On/Off periods. Table 4 shows the results returned by FAM for the corresponding scenario. It can be seen that FAM predicts reasonably well the BTF values and the cross traffic nature for the 0, 0.125, 0.25, 0.375 and 0.625 cases and slightly overestimates the BTF for the 0.5 case. Overall, despite a slight discrepancy in estimating the precise level of  
560 BTF in some cases, the results returned by the method are in good agreement with those provided by the simulations.

Table 4: BTF and cross traffic nature estimations, NS-3 - Exp On/Off agg cross traffic

Ground Truth BTF	Estimated BTF and cross traffic nature					
	Cross traffic aggregates				Cross traffic does not aggregate	
	$\leq 0.25$	0.375	0.5	0.625	$\leq 0.25$	$> 0.25$
0	S/A				S/A	
0.125	S/A				S/A	
0.25	S/A				S/A	
0.375		S/A				
0.5			S	A		
0.625				S/A		

565 **Scenario #4 - 802.11ax traffic.** In order to further assess the validity of the approach, in this scenario, we study FAM's accuracy when relying on 802.11ax aggregated probe and cross traffics in the 2.4 GHz band. All the nodes (AP and stations) operate at HE-MCS11 corresponding to 286.8Mbps. The network topology is the same as the scenario depicted in Figure 3. We run our FAM method while considering an empirical threshold  $T$  equal to 100. The results of FAM are shown in table 5. It can be seen that FAM predicts reasonably well the BTF values for all the cases except the fifth case: BTF=0.5 with 10%  
570 of error. In addition, it predicts well the nature of the traffic for all the cases. We, therefore, can conclude that FAM provides results that match well with the



simulations under 802.11ax when relying on theoretical estimates provided by the models.

Table 5: BTF and cross traffic nature estimations for ns3 simulations - 802.11ax traffic

Ground Truth BTF	Estimated BTF and cross traffic nature					
	Cross traffic aggregates				Cross traffic does not aggregate	
	$\leq 0.25$	0.375	0.5	0.625	$\leq 0.25$	$> 0.25$
0	S/A				S/A	
0.125	S/A				S/A	
0.25	S/A				S/A	
0.375		S/A				
0.5		A	S			
0.625				S/A		

### 5.2. FAM validation for non-aggregated cross traffic scenarios

We use this section to validate the accuracy of our method FAM when we consider the models' outcomes based on aggregated and non-aggregated cross traffic. We here consider non-aggregated cross traffic scenarios. Analogously to the previous section, we validate the method by comparing its estimation to the results delivered by the network simulator.

We here evaluate FAM's accuracy when we used the scenario described in Figure 5, where an 802.11n WLAN and an 802.11g WLAN were deployed. Table 6 shows that FAM estimates well the BTF and the type of the traffic for the cases 0, 0.25, 0.375, 0.5 and 0.625 and overestimates the BTF for the case BTF=0.125. Unfortunately, the Wi-Fi link may experience several transient effects in practice. Since the analytical model neglects some protocol aspects such as beacons and MAC losses, it underestimates the aggregation level for some probe packet gaps. As a result, FAM overestimates the BTF which is typically a better outcome for most applications than an aggressive underestimate.

We also evaluated the case where there are several cross traffic STAs. The results show the same tendency. For instance, with 6 STAs (4 STAs receive cross traffic) the method finds the good BTF level except for a BTF=0.125 where the method gives 0.375.

Table 6: BTF and cross traffic nature estimations for ns3 simulations, non-aggregated cross traffic - Three STAs

Ground Truth BTF	Estimated BTF and cross traffic nature					
	Cross traffic aggregates				Cross traffic does not aggregate	
	$\leq 0.25$	0.375	0.5	0.625	$\leq 0.25$	$> 0.25$
0	S/A				S/A	
0.125	S	A			S	
0.25	S/A				S/A	
0.375						S/A
0.5						S/A
0.625						S/A

## 6. Related work

Performance evaluation of Wi-Fi networks has been intensely studied in the literature to offer relevant information and methods to help its configuration and improvement. In this paper, we focus on three categories of work: Available bandwidth estimation, crowd-sensing of wireless networks, and Analytical models.

**Available bandwidth estimation.** Conversely to achievable throughput estimation tools, there exists another pivotal metric for estimating the overall network performance which is the available bandwidth. Active Available bandwidth measurements that involve the usage of probing packets can be generally divided into two categories: Packet Rate Model (PRM) and Probe Gap Model (PGM). The works based on PRM, such as TOPP [7], pathload [8], pathChirp [9] and DietTOPP [10], focus on comparing the probe packet and received packet rates. They inject increasing probe traffic in the network path and try to find the point where link congestion starts occurring to discern the available bandwidth. These methods differ according to the probing rate settlement.

Conversely to PRM, PGM techniques (e.g., Spruce [11], IGI/PTR (Initial Gap Increase/Packet Transmission Rate) [12], Delphi [13]...) leverage the dispersion in time between two received packets to infer the cross traffic. All these approaches were primitively focused on wired networks and unfortunately failed with the complexity of CSMA/CA of Wi-Fi networks. To cope with this latter, more complex schemes were developed, such as Exact [14], IdleGap [15] and WBest [16]. These techniques intensely depend on the fact that the received packet rate or packet gap should be an effective indicator to reveal the cross traffic. However, the dissimilar packet scheduling system under frame aggregation, proposed by modern Wi-Fi versions, prevents the use of these techniques as inter-packet arrival becomes null when they were embedded in the same aggregated frame. For monitoring the network conditions of these new versions of Wi-Fi, WBest+ [3] and AIWC [4] use the characteristics of frame aggregation to discern the available bandwidth by relying on a threshold-based method: if an inter-arrival time is less than a given threshold, the two packets are considered as aggregated. In [17], the mean aggregation level is compared to the one of an analytical model to estimate the network load. The network scenario used in these papers consists of a server with a wired Internet connection that sends [3, 4] or receives [17] probing traffic along the network path to/from a client with a last-hop wireless connection that assumed to be the bottleneck. The scenario considered here is different as both the probe traffic generator and the receiver are associated with the same Wi-Fi AP. The behavior of the mean aggregation is then much more complex as it may be different on the uplink and downlink links due to their different parameters (e.g., MCS). Moreover, the probe traffic accesses twice the channel that decreases the level of probe traffic for which the network becomes saturated and the size of the aggregated frames. While evaluating the probe aggregation levels of the UL probe traffic, the model presented in [17] assumes that the connection between the AP and the server

is ideal. It is therefore modeled as if the server were implemented on the AP. However, the model presented here appraises the aggregation levels of the DL probe traffic by considering that the server is embedded on a second wireless device which is more realistic and practical.

**Crowd-sensing.** Alternatively, some techniques focus on crowd-sensing applications where mobile devices become active sensors to actively estimate the end-to-end network performance. The authors of [18] proposed MCNet, a tool based on active smartphone measurements to estimate the user-perceived performance in enterprise wireless networks. Another work based on channel scans (spectrum sensing) was also proposed in [19]. It is based on passive measurements while our work investigate the effectiveness of active ones. The authors in [20] proposed a measurement crowd-sensing study in the city of Edinburgh to characterize urban Wi-Fi. It revealed several problems in Wi-Fi deployments in public spaces. However, none of these works involves the network load while crowd-sensing the network performance.

**Analytical models.** Wi-Fi Frame aggregation has attracted a lot of attention due to the significant efficiency enhancements brought by this feature. Consequently, several frame aggregation analytical models have been studied in recent years. [21, 22] predict throughput while incorporating frame aggregation under saturated conditions. A Discrete-Time Markov Chain (DTMC) that considers the frame aggregation has been proposed to show its performance under various aspects under unsaturated conditions [23]. Unfortunately, these models are not applicable or extensible to our work since they consider dissimilar scenarios and traffic patterns.

## 7. Conclusion

In this paper, we have proposed analytical models that predict the DownLink aggregation levels of a targeted probe traffic concurrent to cross traffic that can aggregate or not its frames. Based on that, we proposed a method that allows a vanilla device, in particular a smartphone, to estimate the expected network load via the Busy Time Fraction and infer if the current traffic aggregates its frames or not. Our models and our method were validated through ns-3 simulations across a variety of scenarios. Overall, from a general view point and for almost all the tested scenarios, we find that the models and the method estimate with a satisfactory degree of precision the aggregation levels of the probe traffic and the Busy Time Fraction respectively with at most 10% of errors. Based on six levels of network loads from 0 to 0.625 with a resolution of 0.125, FAM seeks to classify the APs into three classes according to the network load: low, medium, and high loaded APs. It therefore compares APs with each others to detect the APs that are less loaded and provide guidance to hierarchize different WLANs. However, we should note that if an application needs better resolution, FAM can be adapted according to this new resolution.

Our future efforts include addressing some of the open challenges in network load estimation by an unmodified mobile device and the extension of the proposed method to be used in a crowd-sensing context.

### Acknowledgments

The authors wish to thank the anonymous referees for their thorough and constructive review of an earlier version of this paper.

### References

- [1] Cisco, Annual internet report (2018–2023) white paper (2020).
- [2] T. Adame, M. Carrascosa, B. Bellalta, I. Pretel, I. Etxebarria, Channel load aware ap / extender selection in home wifi networks using ieee 802.11k/v, *IEEE Access* 9 (2021) 30095–30112. doi:10.1109/ACCESS.2021.3059473.
- [3] A. Farshad, M. Lee, M. K. Marina, F. Garcia, On the impact of 802.11n frame aggregation on end-to-end available bandwidth estimation, in: *IEEE International Conference on Sensing, Communication, and Networking (SECON)*, 2014, pp. 108–116.
- [4] L. Song, A. Striegel, Leveraging frame aggregation for estimating wifi available bandwidth, in: *IEEE International Conference on Sensing, Communication, and Networking (SECON)*, 2017, pp. 1–9.
- [5] IEEE, Standard for information technology–telecommunications and information exchange between systems - local and metropolitan area networks–specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications, *IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016)* (2020) 1–4379.
- [6] N. E. H. Bouzouita, A. Busson, H. Rivano, Inference of wi-fi busy time fraction based on markov chains, [Research Report] Inria Lyon. 2022. hal-03641948.
- [7] B. Melander, M. Bjorkman, P. Gunningberg, A new end-to-end probing and analysis method for estimating bandwidth bottlenecks, in: *IEEE Global Telecommunications Conference (Globecom)*, Vol. 1, 2000, pp. 415–420.
- [8] M. Jain, C. Dovrolis, Pathload: A Measurement Tool for End-to-End Available Bandwidth, in: *Passive and Active Measurement Workshop (PAM)*, 2002, pp. 14–25.
- [9] V. Ribeiro, R. Riedi, J. Navrátil, L. Cottrell, PathChirp: Efficient Available Bandwidth Estimation for Network Paths, in: *Passive and Active Measurement Workshop (PAM)*, 2003, pp. 1–11.
- [10] A. Johnsson, M. Bjorkman, B. Melander, An Analysis of Active End-to-end Bandwidth Measurements in Wireless Networks, in: *IEEE/IFIP Workshop on End-to-End Monitoring Techniques and Services*, 2006, pp. 74–81.

- [11] J. Strauss, D. Katabi, F. Kaashoek, A measurement study of available bandwidth estimation tools, in: ACM SIGCOMM Conference on Internet Measurement (IMC), 2003, p. 39–44.
- [12] N. Hu, P. Steenkiste, Evaluation and characterization of available bandwidth probing techniques, *IEEE Journal on Selected Areas in Communications* 21 (6) (2003) 879–894.
- [13] V. Ribeiro, M. Coates, R. Riedi, S. Sarvotham, B. Hendricks, R. Baraniuk, Multifractal cross-traffic estimation, in: ITC Specialist Seminar on IP Traffic Measurement, Modeling, and Management, 2000, pp. 15–1.
- [14] S. Shah, K. Chen, K. Nahrstedt, Available bandwidth estimation in IEEE 802.11-based wireless networks, First ISMA/CAIDA Workshop on Bandwidth Estimation (BES), 2003, pp. 1–1.
- [15] H. K. Lee, V. Hall, K. H. Yum, K. I. Kim, E. J. Kim, Bandwidth estimation in wireless lans for multimedia streaming services, in: IEEE International Conference on Multimedia and Expo (ICME), 2006, pp. 1181–1184.
- [16] M. Li, M. Claypool, R. Kinicki, Wbest: A bandwidth estimation tool for IEEE 802.11 wireless networks, in: 2008 33rd IEEE Conference on Local Computer Networks (LCN), 2008, pp. 374–381.
- [17] N. E. H. Bouzouita, A. Busson, H. Rivano, Analytical study of frame aggregation level to infer IEEE 802.11 network load, in: International Wireless Communications and Mobile Computing (IWCMC), 2020, pp. 952–957.
- [18] S. Rosen, S. Lee, J. Lee, P. Congdon, Z. M. Mao, K. Burden, Mcnet: Crowdsourcing wireless performance measurements through the eyes of mobile devices, *IEEE Communications Magazine* 52 (10) (2014) 86–91.
- [19] J. Shi, L. Meng, A. Striegel, C. Qiao, D. Koutsonikolas, G. Challen, A walk on the client side: Monitoring enterprise wifi networks using smartphone channel scans, in: IEEE International Conference on Computer Communications (IEEE INFOCOM), 2016, pp. 1–9.
- [20] A. Farshad, M. K. Marina, F. Garcia, Urban wifi characterization via mobile crowdsensing, in: IEEE Network Operations and Management Symposium (NOMS), 2014, pp. 1–9.
- [21] N. Mohammad, S. Muhammad, Modeling and analyzing mac frame aggregation techniques in 802.11n using bi-dimensional markovian model, Vol. 293, 2012, pp. 408–419.
- [22] N. Hajlaoui, I. Jabri, M. Ben Jemaa, An accurate two dimensional Markov chain model for IEEE 802.11n DCF, *Wireless Networks* 24 (2018) 1019–1031.
- [23] B. S. Kim, H. Y. Hwang, D. K. Sung, Effect of frame aggregation on the throughput performance of IEEE 802.11n, in: IEEE Wireless Communications and Networking Conference (WCNC), 2008, pp. 1740–1744.

## Author Biography

# Nour El Houda Bouzouita

33 (0)604783150 | [nourelhouda.bouzouita@gmail.com](mailto:nourelhouda.bouzouita@gmail.com) | [publons.com](https://publons.com) | [Linkedin](#)

A Computer communications and telecommunications engineer — PhD in computer science

- **Surname:** Bouzouita
- **Name:** Nour El Houda
- **Nationality:** Tunisian

## EDUCATION

### Ph.D. in computer science

*ENS of Lyon*

Lyon, France

Nov. 2018 – Jan. 2022

- **Thesis title:** "Load inference in Wi-Fi networks: models and experiments"
- **Laboratories:** LIP (Laboratoire de l'Informatique et du Parallélisme, UMR 5668) and CITI (Centre d'Innovation en Télécommunications et Intégration de services, EA 3720)
- **Director:** Anthony Busson, Professor at University Lyon 1
- **Co-director:** Herve Rivano, Professor at INSA Lyon
- **Examiners:** Khaled Boussetta, Associate professor at University Paris 13  
Véronique Vèque, Professor at University Paris-Saclay
- **Reporters:** Thomas Noël, Professor at University of Strasbourg  
Lila Boukhatem, Associate professor at University Paris-Saclay
- **Work done:** Mathematical modeling of the Wi-Fi network and implementation of the models using Matlab  
Realization of simulation scenarios on the ns-3 simulator  
Implementation of experiments: Android, C  
Creation of tests and automation scripts  
Data analysis and visualization using Python, Jupyter notebook, Matplotlib

### Network computing and telecommunications Engineer

*National Institute of Applied Sciences and Technology (INSAT)*

Tunis, Tunisia

Sep. 2013 – June 2018

- passed with highest honors

### High school diploma in "mathematics"

*Lycée EL ALia, Bizerte*

Bizerte, Tunisia

2009 – 2013

- passed with highest honors

## EXPERIENCE

### Final year engineering trainee

*LIP laboratory, ENS of LYON*

March–July 2018

Lyon, France

- **Project Area:** Wireless Networks
- **Objective:** Optimization of Wi-Fi network through Crowd-Sensing applications.
- **key words:** Wi-Fi, Crowd-sensing, Android, APISENSE.

### Engineering trainee

*Sofrecom Tunisie*

July–August 2017

Tunis, Tunisia

- **Project Area:** Development
- **Objective:** Design and development of a JSON data interactive diagram generator with AngularJS and D3JS.

### Engineering trainee

*Tunisian bank*

July 2016

Tunis, Tunisia

- **Project area:** Network and security
- **Objective:** Configuration of network equipment and security of the Tunisian bank.

### Trainee

*Tunisie Telecom*

Aug. 2015

Bizerte, Tunisia

- **Project Area:** Networking
- **Objective:**
  - \* Realization of interventions after complaints
  - \* Maintenance of Tunisie Telecom WAN networks

## PUBLICATIONS

- 
- Peer-reviewed International Conference** | *Alicante, Spain* November 2021
- N. E. H. Bouzouita, A. Busson, H. Rivano, Exploiting Frame Aggregation to Enhance Access Point Selection, in: PE-WASUN 2021, Alicante, Spain [\[hal\]](#)
- Peer-reviewed International Conference** | *Limassol, Cyprus* June 2020
- N. E. H. Bouzouita, A. Busson, H. Rivano, Analytical study of frame aggregation level to infer IEEE 802.11 network load, in: 2020 International Wireless Communications and Mobile Computing (IWCMC), 2020, pp. 952–957, [\[hal\]](#).
- Peer-reviewed National Conference** | *Lyon, France* Sep. 2020
- N. H. Bouzouita, A. Busson, H. Rivano. "Etude du niveau d'agrégation des trames dans les réseaux IEEE 802.11 pour l'évaluation du niveau de charge". CORES 2020, Sep 2020, Lyon, France, [\[hal\]](#).
- Peer-reviewed International Journal** | *Computer Communications* Feb. 2021
- [Under major revision] N. E. H. Bouzouita, A. Busson, H. Rivano, FAM: A Frame Aggregation based Method to infer the load level in IEEE 802.11 networks, submitted to Computer Communications Journal.
- Peer-reviewed International Journal** | *Ad Hoc Networks* April 2022
- [To be submitted] N. E. H. Bouzouita, A. Busson, H. Rivano, Inference of Wi-Fi Busy Time Fraction based on Markov Chains, to be submitted to Ad Hoc Networks.

## TECHNICAL SKILLS

---

**Programming languages:** Python, Bash, Matlab, C/C++, L<sup>A</sup>T<sub>E</sub>X  
**WebD:** HTML/CSS, JavaScript  
**Developer Tools:** Git/Gitlab, Visual Studio, PyCharm, IntelliJ, Jupyter notebook, Android Studio  
**Data science:** Numpy, Pandas, Matplotlib, scikit-learn  
**Network & Simulations:** Ns-3, Wireshark, Scapy, Iperf

## RESEARCH SKILLS

- 
- Scientific writing
  - experimental test-beds
  - Peer reviewing
  - Mathematical modeling
  - Adaptable to new programming languages and developer tools.
  - Discrete-event simulation (ns-3, Matlab)

## PROFESSIONAL ACTIVITIES

**Paper reviews**

I have reviewed papers for the following journals: Transactions on Emerging Telecommunications Technologies ( [Certificate Link](#) )

**Presentations**

- **December 2021, Seminar of the INRIA Agora team (CITI Lab Lyon):** Presentation of my thesis work.
- **November 2021, Virtual Conference: "Exploiting Frame Aggregation to Enhance Access Point Selection"** Presentation associated with the publication for the PE-WASUN 2021 conference.
- **May 2021: Working group on Performance Evaluation for Networks in Lyon:** Presentation of my thesis work.
- **September 2020, Virtual Conference: "Etude du niveau d'agrégation des trames dans les réseaux IEEE 802.11 pour l'évaluation du niveau de charge"** Presentation associated with the publication for the CORES 2020 conference.

- **June 2020, Virtual Conference:** “Analytical study of frame aggregation level to infer IEEE 802.11 network load” Presentation associated with the publication for the IWCMC 2020 conference.

#### Events organization

- Co-organizer of PhD student days 2020 at the LIP Laboratory (Lyon)

#### CERTIFICATION

---

**Certified Linux Administrator (LPIC-1)** [Link](#)

**CCNA 1,2,3,4** [Link](#)

**Reviewer CERTIFICATE** [Link](#)

#### COMMUNICATION

---

**Arabic:** Native or bilingual proficiency

**English:** Full professional proficiency

**French:** Full professional proficiency

**Italian:** Notions

#### RELEVANT COURSES

---

**Classroom:** Writing scientific articles, Introduction to data mining using R, Introduction to Machine Learning, Programming with python

**Online:** Prepare a memorable conference presentation

#### REFERENCES

---

**Anthony Busson:** Professor at University Lyon 1, France, [anthony.busson@ens-lyon.fr](mailto:anthony.busson@ens-lyon.fr), +33 6 68 10 83 33

**Herve Rivano:** Professor at INSA Lyon, France, [herve.rivano@inria.fr](mailto:herve.rivano@inria.fr), +33 6 37 64 76 92



Author Photo

[Click here to access/download;Author Photo;Author\\_photography.jpg](#)



**Declaration of interests**

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: