Performance evaluation of OFDMA and aggregation downlink stateless service disciplines in Wi-Fi networks

Anh Tuan Giang, Anthony Busson

Performance evaluation of OFDMA and Aggregation downlink stateless service disciplines in Wi-Fi networks*

Anh Tuan Giang, Anthony Busson

May 29, 2024

Abstract

Aggregation and OFDMA (Orthogonal Frequency Division Multiple Access) are two fundamental features allowing access points and stations to benefit from the high physical transmission rates of the recent Wi-Fi standards. They aim to pool frames with a unique overhead (mainly the physical header and the acknowledgment) to mitigate its impact on the throughput. Natural implementations of these features lead to non-FIFO (First In, First Out) service disciplines that may generate unfairness between frames. In this paper, we evaluate three stateless service disciplines that require a low level of resources (computation and memory). We prove that when OFDMA does not introduce additional delay, one of these greedy algorithms minimizes the overhead. Whereas this service discipline maximizes the system capacity, it generates strong unfairness between the stations. Instead, the two other algorithms may offer a good trade-off between capacity and fairness. These algorithms are evaluated through a theoretical framework and simulations that replay actual Wi-Fi traces.

1 Introduction

The new Wi-Fi standards, formally the family of IEEE 802.11 standards, continuously improve Wi-Fi networks from the physical to the MAC layers. Over the years, the physical transmission rate (at which data is transmitted) has increased from 1-2Mbit/s up to several Gbit/s for the last standards. Such transmission rates made the procedure to access the medium the bottleneck that limits the throughput as it can be longer than the data transmission itself. A new feature has then been brought in Wi-Fi 4, frames aggregation, which pools several frames intended for the same destination together with a

*This work was supported by the Vietnam Academy of Science and Technology under the grant VAST01.07/23-24 "Join Scheduling algorithm and OFDMA-resources allocation for IEEE 802.11ax network".

unique access procedure mitigating its effect on the throughput. More recently, OFDMA (Orthogonal Frequency Multiple Access) has been defined in the two recent standards IEEE 802.11ax (Wi-Fi 6) and IEEE 802.11be. Its purpose is also to pool several frames together, even if their destinations differ. Basically, OFDMA allocates a subset of the sub-carries that compose the channel to each frame. The frames are then sent in parallel with a unique medium access procedure. These two features definitely improve the throughput that the Wi-Fi network can reach. However, aggregation and OFDMA may change the transmission order regarding the arrival of the frames in the buffer, i.e., use a non-FIFO discipline to pool a maximum of frames together. This change may generate unfairness between the frames and increase the delay spent in the buffer for certain frames.

We can make the analogy with a queue in a supermarket where, for each customer, the cashier begins with a small chit-chat, scans the products, prints the bill, and waits for the payment. The technology makes the scans of the products very fast, and the associated time becomes negligible with regard to the overhead (chit-chat, bill, and payment). The idea is then to pool several customers with only one chit-chat, bill, and payment. However, all customers cannot be pooled together. It depends on some criteria. The cashier can then ask customers far in the queue to verify the criteria to join the current customers in service to minimize the overhead. Such systems clearly decrease the time to empty the queue, but it may also increase significantly the waiting time for some customers and generate frustration.

The service disciplines that select the customers/frames for the next service must thus offer a trade-off between fairness/delay and throughput. The literature needs to include theoretical studies assessing the performance of queuing discipline for this kind of system. Classical results on queuing theory do not apply here, as the load and the busy-time periods (periods where the system is not idle) depend on the service discipline itself. Metrics that usually measure fairness, such as the variance on the delay, for instance, cannot be used anymore for the same reasons. In this paper, we give some insights about such systems both from a theoretical and practical point of view. The contributions of this paper can be summarized as follows:

- We revisit the classical formulae from the queuing theory for this particular context. We give, for instance, the relationship between the load and the pooling rate (mean number of customers served together).
- We propose three stateless disciplines that combine both aggregation and OFDMA and select the corresponding frames for each transmission. To our knowledge, scheduling algorithms considering aggregation and OFDMA have not been proposed in the literature.
- We give some insights about the optimality of these disciplines.
- We compare and evaluate these disciplines, in particular their capacity to offer a trade-off between fairness and throughput.

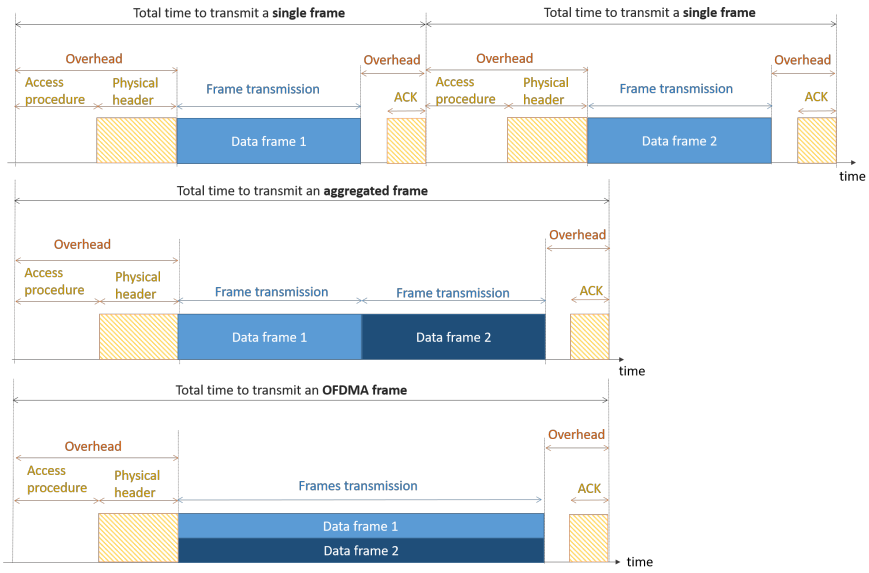


Figure 1: Frame transmission.

- For a practical implementation of these disciplines where OFDMA may generate padding, we propose a metric (as a function of the physical transmission rates and frame sizes), that allows the AP to evaluate the efficiency of a given combination of frames and OFDMA carriers allocation.

The paper is organized as follows. Section 2 presents the aggregation and OFDMA features, and we overview the literature on OFDMA queuing discipline and fairness in queues. Section 3 details the model, the service disciplines, and the theoretical results. Section 4 and 5 are devoted to the numerical results with simulations in Section 5 that faithfully reproduce OFDMA in Wi-Fi 6. We conclude in Section 6.

2 Technological context and state of the art

2.1 Aggregation and OFDMA

This Section overviews the aggregation and OFDMA features implemented in recent Wi-Fi networks. For the sake of clarity, we keep a low level of detail adapted to the analysis carried out in this paper. Nevertheless, the reader can refer to [1, 2, 3] for more technical presentations.

Before presenting frame aggregation and OFDMA, we describe the procedure for the transmission of a single frame. It is illustrated at the top of Figure 1.

The **access procedure** is defined by the CSMA/CA (Carrier Sense Multiple

Access/Collision Avoidance) mechanism. First, a node with a frame to transmit listens to the channel to determine if it is idle. Its time may be random to avoid collisions with the transmissions from the other nodes. If the channel is sensed as idle, the frame transmission can start.

The transmission begins with a **physical header** that gives the parameters of the physical transmission to the receiver. It is followed by the **frame transmission** composed of a frame header, which contains link layer information (e.g., the MAC addresses) and the payload (which is most of the time an IP packet). In case of a proper reception, the receiver senses the channel, and if it is idle, it sends an **acknowledgment**.

In Figure 1, we represent the transmission of two successive frames. We can observe that the overhead (Access procedure, physical header, and acknowledgment) takes an important proportion of the time. Most of the parts in the overhead are constant (from one transmission to another) and incompressible. The time to transmit the frame (in blue in Figure 1) depends on the data length and the physical transmission rate. It varies from 8 to 1201 Mbit/s in Wi-Fi 6 according to the quality and width of the radio channel. The ratio overhead/frame transmission represented in the figure is not unrealistic. For instance, in Wi-Fi 6, a realistic set of parameters [1] leads to a ratio of 47%.

This overhead has become an important limitation of the throughput in Wi-Fi networks, particularly with the recent Wi-Fi generations and their very high physical transmission rates. It has led to the definition of new Wi-Fi features to mitigate the effect of the overhead on the throughput: aggregation and OFDMA.

Aggregation. When several frames are intended for the same destination, the sender can aggregate these frames in a single transmission. There is then only one access procedure and one physical header followed by the frames (MAC header and data). The receiver acknowledges the different frames through a single ACK (a block ACK in practice). An example of the transmission of an aggregated frame (with two frames) is given in the middle of Figure 1. With the same parameters as in the previous paragraph and two aggregated frames, the overhead count is now only 31% of the total transmission time. The Wi-Fi standards limit the number of frames that can be aggregated. It depends on several factors (Wi-Fi generation, maximum transmission time, number of ko, etc.). However, it can reach several tens of frames, drastically reducing the impact of the overhead on the throughput.

Orthogonal Frequency-division Multiple Access (OFDMA). We focus on downlink OFDMA, i.e. transmissions from the access points (AP) to the stations. Since the second Wi-Fi generation (Wi-Fi 2), data is sent over multiple carrier frequencies. For instance, with a 20MHz channel, 64 carriers were used in Wi-Fi 2 and 256 in Wi-Fi 6. The carriers were all used for a single frame in the previous Wi-Fi generations. With OFDMA, the sender (the AP here)

[1] 1000 bytes of payload, Modulation and Coding Scheme 3, one spatial stream, Guard Interval=0.8μsec, channel width=20MHz

may allocate different subsets of carriers to a set of frames that are then sent in parallel. An OFDMA transmission with two frames is illustrated at the bottom of Figure 1. After the access procedure and the physical header, several data frames are transmitted in parallel. The Y-axis represents the carriers allocated to frame 1 and frame 2. The different frames are then acknowledged. In Wi-Fi 6, OFDMA is exclusively used for frames intended for different destinations. As for aggregation, with OFDMA, the different frames share the overhead. It decreases the ratio between the overhead and the total transmission time. Aggregated or OFDMA frames will be called jumbo frames in the rest of this paper.

2.2 State of the art

We first overview literature articles that analyze or propose scheduling algorithms for aggregation or OFDMA in Wi-Fi networks. Fairness in a queue, which poses the fairness problem of the service disciplines, is addressed in a second paragraph.

Aggregation and OFDMA scheduling algorithms. As aggregation was introduced before OFDMA in the Wi-Fi standards, its performance has been studied first independently of OFDMA. In [4], the authors use an analytical model to study aggregation between one AP and one station. In order to ensure aggregation, the algorithm waits to have at least K frames in the transmission buffer before transmitting the jumbo frame. This particular mechanism makes aggregation counterproductive for a great value of K. The optimal value of K is then studied for certain scenarios. The relationship between recent congestion control algorithms and the aggregation feature is investigated in [5]. The authors study the performance of these congestion control algorithms when the Wi-Fi network is the bottleneck. With aggregation, packets arrive in batches. It breaks the network's classical behavior, particularly for the Round Trip Time. For this scenario, the impact of the aggregation size on the throughput and the latency is evaluated. In [6], an algorithm is proposed to evaluate the network load from the current frame aggregation level (the mean number of frames aggregated in a jumbo frame). The algorithm is based on a Markov chain that models a service discipline where each jumbo frame systematically includes the oldest frame of the transmission buffer (the first frame in the FIFO order).

Since the inclusion of OFDMA in the Wi-Fi standard, several papers have studied its impact on performance and its possible implementations. All studies deal with OFDMA algorithms with one AP and N stations and uplink communications. [7] and [8] consider upload OFDMA scheduling formulated as a resource allocation problem. The scheduler aims at maximizing the throughput, fairness (proportional fairness), and the remaining processing time (remaining time to process a flow), which is supposed to know the flow size. The allocation itself is solved through a linear optimization problem. In [9], the authors evaluate the performance of uplink OFDMA with an analytical model. A two-dimensional Markov chain is used to model the backoff of the stations (used to send their frame to the AP). They estimate the network performance, particularly the

throughput and the BSR (Buffer Status Report) delivery rate. The latter expresses the capacity of the stations to send their BSR to the AP (as the stations contend for it). The BSR is crucial to inform the AP about uplink traffic and, therefore, to allocate enough resources to the AP. Uplink transmissions in non-saturated conditions are evaluated in [10]. As the network is not saturated, they consider the delay rather than the throughput as one of the key metrics. They develop an analytical model based on a Markov chain to model both Wi-Fi 6 and legacy stations (stations that implement older Wi-Fi versions). They show that these heterogeneous networks may lead to unfair situations where legacy stations access the channel more often than Wi-Fi 6 stations. It is due to the fact that with OFDMA, uplink transmissions are regulated by the AP, which competes with an equal chance to access the medium with the legacy stations. [11] proposes several scheduling algorithms in the UL for real-time applications. It proposes a resource allocation scheme to ensure a given delay for each frame. Moreover, the algorithm allocates the transmission power to each station transmitting in the same OFDMA jumbo frame to ensure that each frame's reception power (on a subset of carriers) is the same at the AP. It is one of the restrictions of the OFDMA uplink implementation. Proportional Resource Scheduling (PRS) is proposed in [12] for uplink and downlink transmissions. PRS distributes the channel resources proportionally to the stations according to their load and throughput (the algorithm keeps the information about the throughput obtained for each flow). The algorithm utilizes the channel efficiently and ensures fairness between flows. In [13], the authors propose a downlink/uplink scheduler that assigns resources using a linear programming technique considering the load of each station. The authors of [14] proposes an analytical model for a simple OFDMA downlink scheduling scheme. Their algorithm divides the set of carriers homogeneously between the destinations which are served in a round robin way.

Most of these works address uplink transmissions.It is a very different problem with regard to downlink transmissions, as the OFDMA carriers allocation must be managed for a set of transmitters that requires control messages and coordination from the AP. Downlink transmission is simpler from a protocol point of view as OFDMA is implemented locally. Moreover, all these papers deal only with OFDMA, whereas the scheduler has to choose for each of its transmission between aggregation and OFDMA. Surprisingly, algorithms for downlink transmissions considering aggregation and OFDMA have never been addressed in the literature.

Fairness in queue OFDMA or aggregation may change the order in which frames are served compared to a classical FIFO discipline. For instance, all the frames with the same destination can be aggregated in a single jumbo frame, whatever their position in the buffer. It may significantly increase the delay for certain frames. It may even generate starvation. For instance, if the scheduling algorithm aims at maximizing aggregation, a destination with a very high arrival rate could starve a small flow intended for another destination. It may happen

when there are always more frames in the buffer for the first flow than for the second one. It raises the question of fairness between the frames or between the flows. In this paper, a flow is the set of frames intended for a given destination.

This problem of scheduling order in the queuing system is referred to as fairness in a queue in the literature.

We focus on fairness in the G/G/1 system. Such a queue corresponds well to our system as the Wi-Fi channel may be seen as a unique server that serves frames/clients stored in a buffer. With such a system, SRPT (Shortest Remaining Processing Time) [15] is the discipline that minimizes the mean sojourn time. This discipline serves the client with the lowest service time first. However, this discipline is exceptionally unfair as clients with a great service time may suffer from an important sojourn time. Consequently, most of the queuing disciplines used in practice offer a trade-off between fairness and performance (sojourn time, throughput, etc.). Note that, for instance, no existing system implements SRPT, even if it is the discipline that minimizes the mean response time.

The notion of justice or fairness is entirely subjective and also depends on the application (persons queuing in a supermarket, packets waiting for the output interface in a router, running jobs in a CPU, etc.). Therefore, different definitions and approaches may be found in the literature. We review only a few of them, which correspond, in our opinion, to the most representative works.

In [16], a discipline is considered fair if

$$\mathbb{E}\left[\frac{R(x)}{x}\right] \leq \frac{1}{1-\rho} \tag{1}$$

where $R(x)$ is the response time of a client with a service time of length x, and ρ is the system load. This definition states that, on average, a fair queuing discipline should keep the response time proportional to the service time. The constant $\frac{1}{1-\rho}$ is not chosen arbitrarily but corresponds to the lowest constant that verifies equation 1. The paper aims to classify fair and unfair disciplines according to this fairness definition. For instance, processor sharing, where the server processes in parallel all the clients currently in the system with an equal part of its capacity, is shown as fair. Instead, SRPT discipline is unfair.

In [17], the authors defined a generic function based on four properties that reflect the notion of fairness. One of these properties is related to seniority, i.e., related to the arrival order of the clients. For the G/G/1 queue, the fairness function $\overline{F}$ boils down to the difference between the variance of the waiting time for the considered discipline and FIFO:

$$\overline{F} = c\left(Var(W_{discipline}) - Var(W_{FIFO})\right) \tag{2}$$

where c is a positive constant. With such a definition, the FIFO discipline is proved to be the fairest among the service disciplines that do not take into account the client service time in the ordering (e.g., LCFS: Last Come First Served). For the other disciplines, which are service time-dependent, permutations with regard to the order of the arrival may decrease the variance. They

may be consequently fairer than FIFO according to this fairness measure. The "pairwise fair" discipline is given as an example, where a permutation between two clients is performed if it reduces the difference in their waiting times.

W. Sandmann in [18] and [19] defines another fairness measure called Discrimination frequency. It sums two quantities for an arbitrary client C: i) the number of clients that arrived later than C but completed their services before C ii) the number of clients with a greater remaining service time when C arrives and that complete their service before C. The discrimination frequency is derived for certain disciplines (FIFO, LCFS, SJF: Short Job First, a variant of SRPT, etc.) and different service times distribution. They show that, for particular distributions, LCFS is the less fair among these disciplines and SRPT the fairer.

Discussion These different fairness measures are unsuitable for our particular system for several reasons. With OFDMA and aggregation, the receiver delivers the frames to its operating system (OS) once the whole jumbo frame is received. It is obvious for OFDMA as all individual frames end simultaneously, but it is also the current implementation that we observed with aggregation. In Appendix 7, we infer the aggregation implementation on some commercial Wi-Fi cards. It appears that, on all these products, the frames that compose the jumbo frames are sent to the OS at the end of the jumbo frame reception. Consequently, from the point of view of a frame, the transmission time (i.e., the service time) increases with the size of the jumbo frame it belongs to. Equation 1 is thus difficult to apply as x, the service time strongly depends on the service discipline. The variance of the waiting time as expressed in Equation 2 is neither appropriate. In classical G/G/1 systems, the busy time periods and, thus, the system load is invariant to the service disciplines. Clients can then be scheduled in such a way that they wait approximately the same time without impacting the system load. Aggregation and OFDMA reduce the busy time periods. Variance can then decrease due to this gain of load (the waiting time globally decreases), but it is not more able to measure fairness between the clients. Moreover, practical implementation will not reorder frames of the same flow to guarantee that data will be delivered in good order at the transport or application layer. Consequently, fairness issues may arise between flows rather than between frames of the same flow. The variance expressed at the frame level is thus not appropriate.

3 Stateless service disciplines and theoretical results

The system is illustrated in Figure 2. It is composed of an AP and N stations. The buffer of the AP is represented on the top left of the figure, where the frames are represented in their arrival order (FIFO order). Each color represents a different destination (4 destinations in this example). In this paper, we consider

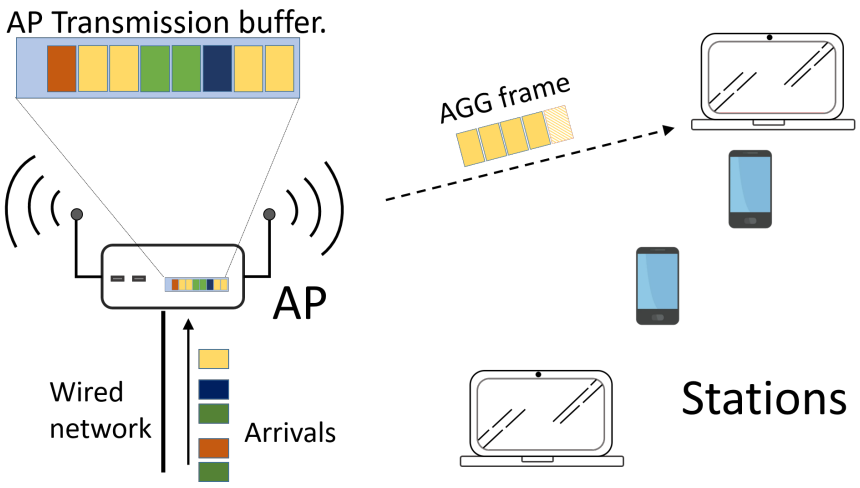


Figure 2: System description: We model the down-link transmission queue of an AP. Traffic arrives from the wired network. N stations (4 in this example) are associated with the AP. The frames have different colors for each station/destination. The AP sends an aggregated frame to the first station in this example.

and evaluate 3 service disciplines. Their roles are to select frames in the buffer that will compose the next transmission (the jumbo frame) and the type of transmission: OFDMA or aggregation. All these disciplines are stateless, i.e., they do not rely on any history of the previous transmissions, as the service discipline must remain very simple in terms of implementation, complexity and memory requirements. The stateless disciplines that are evaluated in this paper are presented below and illustrated in Figure 3.

- FIFO POOLING. With this discipline, the frames are sent in the FIFO order. When several consecutive frames have the same destination, aggregation (denoted AGG in the figure) is used. When several consecutive frames are intended for different destinations, OFDMA is used instead.
- MAX FIFO POOLING takes the first frame in the buffer (in the FIFO order), and it chooses in the rest of the buffer the frames that will maximize the number of frames for the next transmission. In Figure 3, the next jumbo frame begins with the frame 1 as it is the next in the FIFO order. Then, MAX FIFO POOLING compares the size of the maximum OFDMA jumbo frame (it has a size of 3 frames as there are 3 frames with different destinations in the buffer) with the maximum AGG frame (number of frames with the same destination as frame 1). The latter has the maximum size with a jumbo frame composed of 4 frames: 1, 2, 6, and 7. Once this jumbo frame is transmitted, the next transmission begins with frame 3,

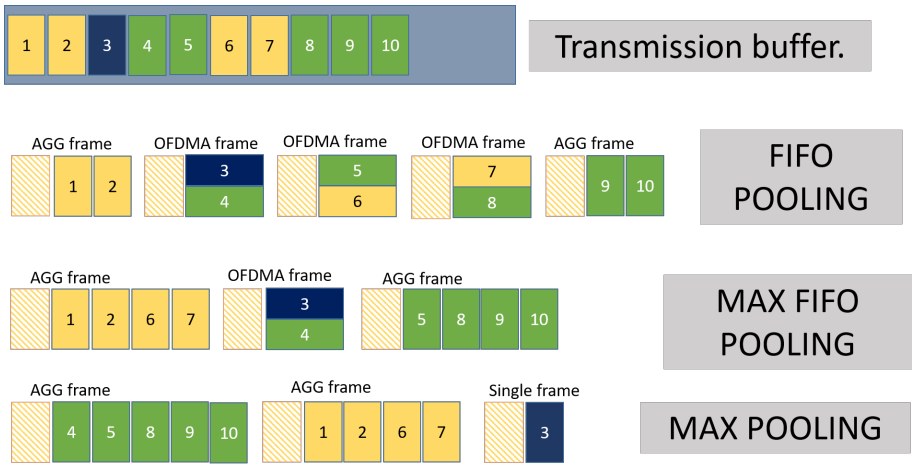


Figure 3: The three scheduling algorithms: FIFO POOLING, MAX FIFO POOLING, and MAX POOLING.

as it is the next frame in the FIFO order. It can be pooled with only one other frame (frame 4 here), and so on.

- MAX POOLING aims to maximize the jumbo frame size without considering the frames' arrival order. It is a greedy algorithm that maximizes the jumbo frame size for each transmission. In Figure 3, the first jumbo frame is composed of 5 frames, an aggregated frame with all the frames intended for the green destination. Any other combination leads to smaller jumbo frames. It is followed by a jumbo frame with the 4 frames intended for the yellow destination, and so on.

For this example, the disciplines MAX FIFO POOLING and MAX POOLING have one overhead less than the FIFO discipline. The time to send all the frames in the buffer is then reduced for these two disciplines.

3.1 MAX POOLING optimality

We show that for a given set of frames in the buffer, the MAX POOLING discipline minimizes the number of transmissions and, consequently, the number of overheads. We consider the buffer at a given time. We assume that it contains frames intended for N different destinations with n_i frames from each destination i ($i = 1, .., N$). A transmission using aggregation for destination i is denoted A_i. An OFDMA transmission is denoted O. A_i and O are called a draw (rather than transmission) in the rest of this proof. We consider only draws with a maximum of frames, i.e., a draw A_i will take all frames with destination i, and a draw O will take one frame of each destination i for all i such that $n_i > 0$.

Property 1 *Let σ be a sequence of draws. If σ leads to an empty buffer, then any reordering of σ leads to an empty buffer.*

Proof Let e be one of the frames present in the buffer. We assume that e is intended for destination i. Since σ leads to an empty buffer, the draw A_i belongs to σ or σ contains at least n_i draws O. This condition is necessary to process all frames with destination i. In both cases, in any reordering of σ, there is a draw that takes the frame e. □

Property 2 *The MAX POOLING discipline minimizes the number of draws.*

Proof Let R be the minimal number of draws to empty the buffer. We get,

$$R \leq min(N, \max_i(n_i))$$

This inequality is an upper bound on the number of draws of an optimal sequence. It considers only draws $A_\cdot$ or draws O to empty the buffer. We show through proof by contradiction that the draw made by the MAX POOLING algorithm at each step is one of the draws of the optimal sequence. As any reordering of the optimal sequence leads to the same optimal (as shown in Property 1), it proves that the MAX POOLING discipline minimizes the number of draws.

At this step of the MAX POOLING algorithm, there are two possibilities. 1) It exists i such that $n_i \geq n_j$ for all $j \neq i$ and $n_i > N$. To serve the n_i frames, n_i draws O are required, or one draws A_i. The choice of n_i draws O is suboptimal as it exceeds the inequality above. Consequently, A_i belongs to the optimal sequence. It is the choice made by the MAX POOLING algorithm. 2) $N \geq n_i$ for all i. If there is no draw O in the optimal sequence, then N draws $A_\cdot$ are required, which exceeds the inequality. Consequently, there is at least one draw O in the optimal sequence. It is the choice made by the MAX POOLING algorithm. □

It proves that the complexity of the service discipline that minimizes the overhead is $O(n)$, where n is the number of frames in the buffer. It is worth noting that this service discipline minimizes the system load for a given buffer state. As the next frame arrivals in the AP buffer are difficult (and even impossible) to infer, it offers a good estimation of the optimal capacity of the system.

3.2 Performance evaluation: the key metrics

We introduce additional notations. The frame arrivals for each station i have intensity λ_i. Consequently, the total arrival rate on the system is $\lambda = \sum_{i=1}^{N} \lambda_i$. The service time is composed of a constant overhead denoted OV that counts all the overhead described in Section 2.1 and the time to transmit the different frames that compose the jumbo frame. Note that OV is in practice different

Model parameters	
Parameter	Meaning
N	Number of destinations
OV	Overhead
D_i	Mean time to transmit a frame to destination i (does take into account the overhead)
t_i^l	Time to transmit a frame l to destination i (does not take into account the overhead)
λ_i	Frames arrival rate for destination i
λ	Total arrival rate
R	RV that describes the sojourn time
S	RV that describes the service time
W	RV that describes the waiting time
ρ	System load ($0 \geq \rho \geq 1$)
μ	Mean number of jumboframes sent per second
τ	Mean number of frames that composes a jumboframe
π	Stationary distribution of the number of frames/clients in the system

Table 1: Model parameters

for OFDMA and aggregation. The difference is considered as negligible in this Section. More details are given in Section 5.

The time to transmit a frame l to destination i is $OV + t_i^l$ where t_i^l is a sample of a random variable with mean D_i. D_i is different from one destination to another to model the different physical transmission rates used by the AP for each station. If a jumbo frame is composed of k frames to destination i (aggregation), the mean service time is then $OV + \sum_{l=1}^{k} t_i^l$. If the jumbo frame is composed of k frames intended for different destinations (OFDMA), the service time is $OV + \sum_{l=1}^{k} t_{dest(l)}^l$ where $dest(l)$ is the destination of the frame l. It corresponds to a perfect OFDMA transmission. Section 5 gives more details on this assumption.

A frame within a jumbo frame is considered as received by the destination at the end of the service time (as explained in Section 2.1). The mean pooling size τ is defined as the mean number of frames that compose the jumbo frames. The jumbo frames rate μ is the mean number of jumbo frames sent per second. The parameters used in our model are listed in Table 1.

We present our methodology to evaluate some key metrics of the system: stationary distribution of the number of frames (in the system), system load, pooling rate, waiting, and sojourn time. In the following, we implicitly consider that the system admits a stationary distribution and is ergodic.

Stationary distribution $(X_t)_{t \in \mathbb{R}^+}$ describes the number of frames in the system at time t. It is a continuous-time process. Its stationary distribution is denoted $(\pi(k))_{k \geq 0}$. It can be computed from the following formula if the system

is simulated:

$$\pi(k) = \lim_{T+\infty} \frac{1}{T} \int_0^T \mathbb{1}_{X(t)=k} dt \tag{3}$$

where $\mathbb{1}_{\cdot}$ is the indicator function. Equation 3 is the proportion of time spent in the state k. All the metrics that are given below can be deduced from this distribution, and more precisely from its average $\mathbb{E}[X] = \sum_{k=1}^{+\infty} k \cdot \pi(k)$.

Remark 1 *If the frames arrival forms a Poisson point process, it is possible to compute this stationary distribution analytically. First, note that according to the PASTA property (Poisson Arrivals See Time Averages) the chain X_n^A that describes the number of frames in the system at each arrival has the same distribution as $\pi(.)$ but is not Markovian (as the service distribution is not exponentially distributed). The solution to compute $\pi(.)$ is then to consider the chain X_n^S at the departures, which is Markovian. Whereas these two chains (at the departures and at the arrivals) have the same distribution for birth and death processes, it is not the case here as several frames may leave the system at the same time. A solution is then to compute the stationary distribution of X_n^S and to use Palm Calculus to deduce the one of X_n^A (which equals to $\pi(.)$). The transition probabilities of the chain X_n^A are quite complex even for the FIFO POOLING discipline and the computation of the stationary distribution requires a numerical method. Moreover, the computation from $\pi_S(.)$ to $\pi_S(.)$ given by Palm calculus also requires a numerical evaluation. We have thus chosen to use a Monte-Carlo method to evaluate $\pi(.)$ for the different service disciplines.*

System load. The load ρ of a system is the proportion of time the system is busy, i.e., contains at least one frame in our case. It can be deduced from the stationary distribution:

$$\rho = 1 - \pi(0) \tag{4}$$

Property 3 *The load can also be expressed as the mean amount of service performed per second:*

$$\rho = \mu \cdot OV + \sum_{i=1}^{N} \lambda_i D_i \tag{5}$$

Properties 3 and 4 are proved in the appendix. Property 4 gives the relationship between τ and μ.

Property 4

$$\tau \cdot \mu = \sum_{i=1}^{N} \lambda_i \tag{6}$$

Sojourn and service times The mean sojourn time $I\!E[R]$ can be deduced from the Little's formula [20]:

$$I\!E[R] = \lambda I\!E[X] \tag{7}$$

with $I\!E[X] = \sum_{k=1}^{+\infty} k \cdot \pi(k)$. $I\!E[R]$ is composed of the service time $I\!E[S]$ (service of the jumbo frame here) and the mean waiting time $I\!E[W]$ (time spent by a frame in the buffer).

Property 5

$$\rho = \mu I\!E[S] \tag{8}$$

The proof of this property is given in the appendix. We can deduce the waiting time $I\!E[W]$ from these two quantities:

$$I\!E[W] = I\!E[R] - \frac{\rho}{\mu} \tag{9}$$

Unfairness measure Eventually, we define a measure of unfairness between the different destinations as

$$Unfairness = \Big(\sum_{i=1}^{N} I\!E[R_i] \Big)^2 - \sum_{i=1}^{N} I\!E[R_i]^2 \tag{10}$$

This unfairness metric gives a measure of the difference between the sojourn times of the flows/destinations. As mentioned in Section 2.2, the variance of the sojourn time does not reflect inequity, as it is calculated over all frames. In fact, this variance may decrease as the delay decreases thanks to the savings made on the number of overheads, whereas for some destinations, with low traffic for example, the sojourn time may increase significantly because their frames are served later (in a non FIFO order). Unfairness must therefore be measured between destinations rather than between frames. Each destination should also have the same weight in this metric, so as not to favor high-traffic destinations. The proposed metric meets these criteria. It measures the variance between the average sojourn times for each destination.

As Little's formula also holds for each class of customers (each destination here), this quantity can be computed as follows.

$$I\!E[R_i] = \lambda_i I\!E[X_i] \tag{11}$$

where $I\!E[X_i]$ is the mean number of clients in the system with destination i ($I\!E[X_i] = \sum_{k=1}^{+\infty} k \cdot \pi_i(k)$). The stationary distribution of the number of frames in the system for each destination ($\pi_i(.)$) is then sufficient to compute all the performance metrics.

4 Perfect OFDMA: Numerical results

To keep the first results as generic as possible and avoid complex considerations about traffic nature and service times, we model the system as a modified M/D/1 queue.

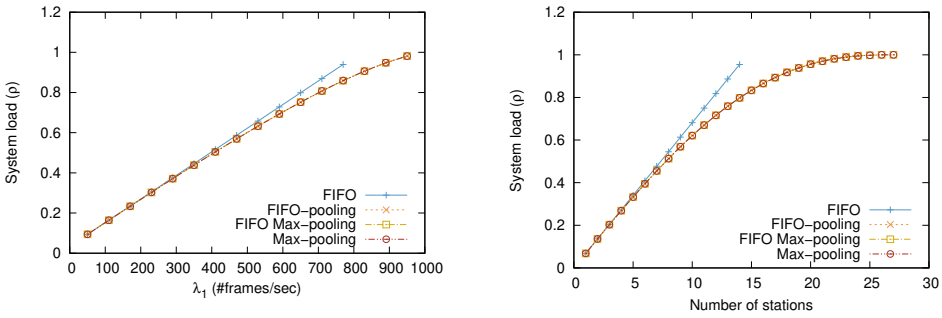
We consider two scenarios to evaluate the impact of the different disciplines on the performance. The first scenario, called the "Two-stations scenario", has two destinations (two stations). The arrival rate for the first destination is constant. It increases for the second destination until it reaches saturation. In the second scenario, "Multi-destinations scenario", the parameters are the same for all destinations, but we increase the number of destinations until reaching saturation. The parameters D_i and OV are set according to the IEEE 802.11ax standard (Wi-Fi 6). They are given in Table 2.

The "Two-stations scenario" corresponds to a case where the buffer will contain mainly frames for the same destination conducive to aggregation, and the "Multi-destinations scenario" with homogeneous flows intended for different destinations should favor the use of OFDMA.

Two-stations scenario		Multi-destinations scenario	
Parameter	Value	Parameter	Value
AIFS	43 μsec	AIFS	43 μsec
Slot time	9 μsec	Slot time	9 μsec
Mean Backoff	67.5 μsec	Mean Backoff	67.5 μsec
Physical header	44 μsec	Physical header	44 μsec
SIFS	16 μsec	SIFS	16 μsec
Block Ack	44 μsec	Block Ack	44 μsec
MCS	0 (8.6 Mbit/s)	MCS	3 (34.4 Mbit/s)
Frame size	1000 bytes	Frame size	1000 bytes
OV	214.5 μsec	OV	214.5 μsec
λ_1	$3e^{-5}$ frames/sec	λ_i	$15e^{-5}$ frames/sec
D_1, D_2	960 μsec	D_i	240 μsec
λ_2	varies	Nb of dest (N)	varies
Number of frames/samples	$9,000,000$		

Table 2: Simulation parameters. OV is computed from the 802.11ax parameters (OV=AIFS + Mean Backoff + Physical header + SIFS + Block Ack). D_i is computed from the frame size plus 32 bytes divided by the physical transmission rate of the destination (its MCS: Modulation and coding scheme). The 32 bytes correspond to a part of the block ack that is transmitted at the same physical transmission rate as the payload. It leads for instance to $\frac{(1000+32)*8}{8.6Mbit/s}$ for D_1.

We implemented a simulator coded in C that simulates the model. It is available here [21]. We generate $9,000,000$ packets for each set of parameters. Then, the simulator computes empirically the stationary distribution according to Equation 3. The different quantities are then deduced from this distribution



(a) Two stations scenario.

(b) Multi-destinations scenario.

Figure 4: System load for the two scenarios.

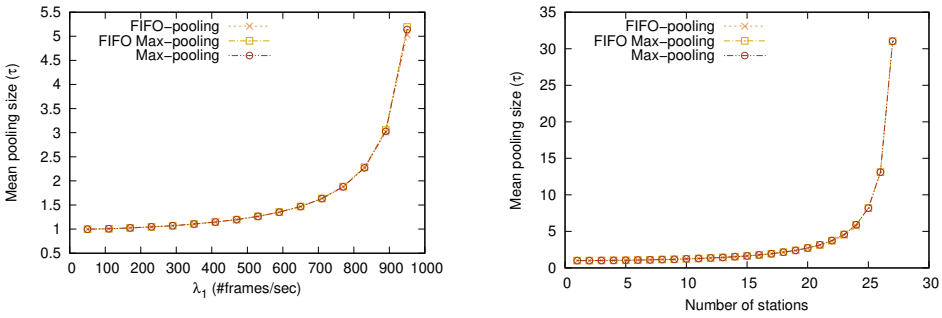
according to the equations given in Section 3.2.

System load As our model has an infinite queue (there is no loss), the system's throughput (number of frames processed per second or bit/s) is equal to the input rate if the load is less than 1. The capacity of the system can thus be seen as the maximal input rates leading to a system load of 1. In all the figures, we do not plot the results when the load is greater than 1 as the system diverges.

In Figures 4a and 4b, we plotted the load for the two scenarios. It is also evaluated for a classical FIFO discipline that does not aggregate frames nor uses OFDMA. It is denoted FIFO in the two figures. As formally shown earlier, the discipline that maximizes the system capacity (in the absence of knowledge about the next arrivals) is the MAX POOLING discipline. The benefit is substantial as the system capacity is approximately 950 frames/sec and 27 destinations for MAX POOLING and 770 frames/sec and 14 destinations obtained with a classical FIFO. There is no visible difference between the three disciplines for the "Two-stations scenario" in terms of load. There is, in practice, a difference of 10^{-2} between FIFO POOLING (with a load of 9.82 for $\lambda_1 = 9.5e^{+02}$) and MAX POOLING (with a load of 9.81 for the same input). The difference is more important for the "Multi-destinations scenario"; the system diverges for 24 destinations with FIFO POOLING and 27 destinations with FIFO MAX POOLING and MAX POOLING. The difference between the two scenarios is mainly due to the frame transmission time, which is 4 times longer in the "Two-station scenario" mitigating the impact of the overhead.

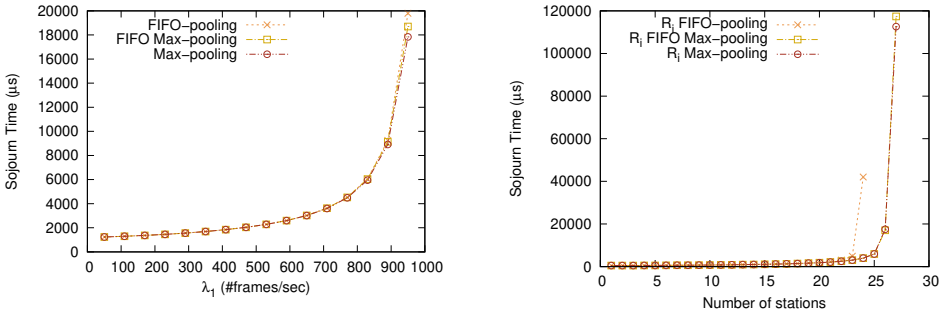
4.1 Mean pooling size (τ)

The mean pooling size (mean number of frames that composed the jumbo frames) is plotted in Figures 5a and 5b. The pooling size is dependent on the number of frames in the buffer. It does not increase linearly with the load. We can then observe an exponential increase when the system reaches saturation.



(a) Two stations scenario. (b) Multi-destinations scenario.

Figure 5: Mean pooling rate for the two scenarios.



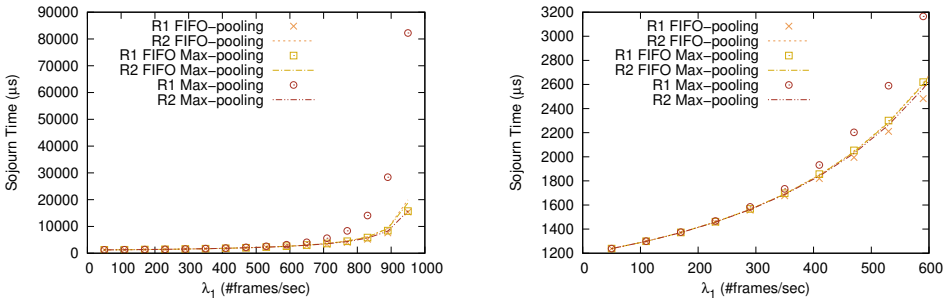
(a) Two stations scenario. (b) Multi-destinations scenario.

Figure 6: Mean sojourn time for the two scenarios.

We observe very different pooling sizes for the two scenarios, until approximately 5 for the "Two-stations scenario" and 30 for the "Multi-destinations scenario". The three disciplines present the same value of τ, except that FIFO POOLING stops at 24 destinations as its capacity is inferior. It is barely visible, but MAX POOLING pools frame a little bit more than FIFO MAX POOLING when the system becomes saturated.

4.2 Mean sojourn time

The mean sojourn time is approximately the same for all disciplines except for FIFO pooling at saturation for the "Multi-destinations scenario". A reasonable sojourn time in a Wi-Fi network is a few milliseconds (it is expected to be less than 5ms for 5G networks for instance). 5ms is thus a good reference to evaluate the users' quality of experience in a Wi-Fi network. In our scenarios, it is reached for $\lambda_1 = 800$ (a load of 0.9) and 25 destinations (a load of 0.98), respectively. The three disciplines are thus able to keep a good operational sojourn time even for high-level loads.



(a) Two stations scenario.

(b) Two-stations scenario. Zoom.

Figure 7: Mean sojourn time for the two-stations scenario.

Nevertheless, this sojourn time is different for all destinations when the traffic is inhomogeneous. In Figures 7a, we plot the sojourn time for each destination of the "Two-stations scenario". The unfairness metric defined in Equation 10 is not necessary here as there are only two destinations. We observe a difference of 65ms between the sojourn time of the two destinations at saturation for the MAX POOLING discipline. This huge difference is due to the fact that as λ_1 increases, the system aggregates frames for destination 1 that are more present in the buffer. Frames for destination 2 may then be transmitted after several jumbo frames. In Figure 7b, we zoom on smaller loads (λ_1 varying between 0 and 600 and the load between 0 and 0.69) corresponding to the operational level of loads, where we still observe a difference of 1ms between the sojourn times of the two destinations.

These results show that it is possible to significantly increase the system's capacity with such features and disciplines. MAX FIFO POOLING offers the best trade-off regarding capacity, where no loss of capacity was observed compared to the optimal (MAX POOLING), but also in terms of fairness as there are no significant differences of sojourn times even when the traffic is very heterogeneous. The selection of the first frame in the buffer mitigates unfairness, but a pure FIFO order, as it is done with FIFO POOLING, is not sufficiently efficient in terms of capacity.

5 Imperfect OFDMA

We present briefly the OFDMA mechanism defined in the IEEE 802.11ax standard and show that it leads to imperfect OFDMA, i.e., to a greater transmission delay compared to a perfect OFDMA transmission.

We describe only the case of a 20MHz channel. The reader can refer to [7] for the other channels width. The number of tones in a 20MHz channel is

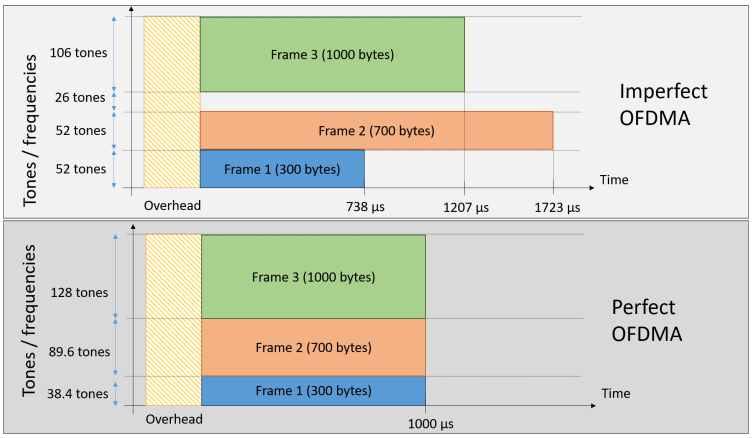


Figure 8: Comparison between perfect and imperfect OFDMA.

256. Tones are grouped into resource units (RU). RU can consist of 26, 52, 106, or 232 tones in a 20Mhz channel. The sum of the RU tones allocated to data transmission is then less than 256. The other carriers are used for direct conversion, Guard, or stay unused. The last constraint is the maximum number of frames in an OFDMA transmission equal to 9 for a 20Mhz channel.

We present a toy example to show how these constraints may increase the transmission time. Let us consider 3 frames with sizes 300, 700, and 1000 bytes intended for 3 destinations. We assume that the AP uses the same physical transmission rate for the three destinations, equal to 16 Mbit/s (transmission rate when the 256 tones are allocated). The RU allocation that minimizes the OFDMA transmission time for this case is 52, 52, and 106 tones for each of the three frames. For the first frame (300 bytes) the transmission time is then $\frac{52}{256}\frac{300\times 8}{16} = 738\mu sec$. We obtain 1723 and $1207\mu sec$ for the two others, respectively. Figure 8 shows the transmission time difference between perfect and imperfect OFDMA transmissions for this example. We can observe that for this particular case, the difference is $723\mu sec$. The benefit of using OFDMA is thus dependent on the frame sizes and the physical transmission rates used for each destination. Depending on these parameters, OFDMA may even be counterproductive compared to the transmission of single frames.

To evaluate the efficiency of a combination of frames, we define the following function that expresses a cost in terms of overhead per frame. Let F be a set of frames pool together for transmission; the overhead cost per frame for an OFDMA transmission is defined as:

$$\frac{1}{|F|}\left(OV + \max_{f\in F}\left(\frac{1}{\alpha_f}t_f\right) - \sum_{f\in F} t_f\right) \tag{12}$$

t_f is the time to send the frame when all the tones are allocated (the time

without OFDMA). α_f is a factor in $[0, 1]$ computed as the ratio between the RU allocated to frame f and the total number of tones (e.g. for a RU=26, we get $\alpha_f = \frac{26}{256}$). This cost takes into account the overhead of the physical header (OV) and the one generated by the imperfect OFDMA (given by the second term: $max_{f \in F}(\frac{1}{\alpha_f})t_f - \sum_{f \in F} t_f$). For the previous example, the overhead cost per frame is $\frac{OV+1723-1000}{3}$. For an aggregated frame, the overhead cost per frame is $\frac{OV}{|F|}$.

We consider two different OV for aggregation and OFDMA. For OFDMA, a HE-SIG-B structure is added to the physical header which describes the RU allocation for each destination. The HE-SIG-B is sent at MCS-0 and contains a common header of less than 32 bits and 20 bits per user. As there are at most 9 RU that can be allocated in a 20MHz channel, the additional time is then at most $25\mu s$. The overhead OV is then $214.5\mu s$ for aggregation (see Table 1 for the computation details) and at most $239.5\mu s$ for OFDMA. The simulation presented in this section has been made with and without this additional field. Note that for the chosen parameters the OFDMA additional field does not impact the numerical results.

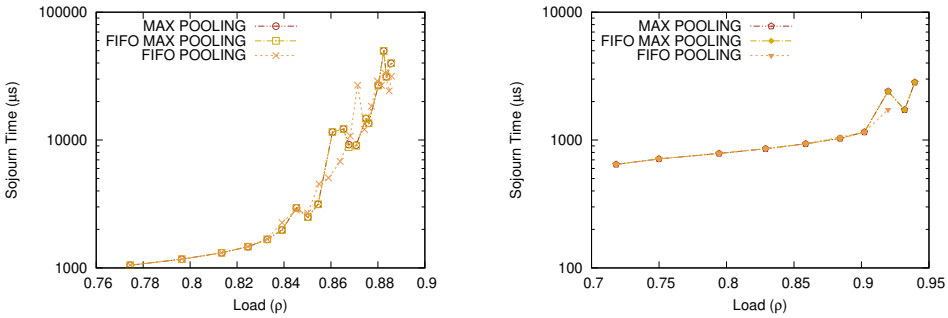
We adapt the three algorithms defined in Section 3 to take into account imperfect OFDMA. In the previous algorithms, a maximum of frames were pooled together. Here, the different possible combinations of frames are selected according to the overhead cost per frame.

- FIFO POOLING. Frames are sent in the FIFO order. If aggregation or OFDMA is possible for the first k frames, the overhead cost per frame is computed. The algorithm starts with $k = 1$ and increments it until OFDMA or aggregation is no longer possible. Let k_{max} be the maximum number of frames that can be pooled. FIFO POOLING selects the i first frames ($1 \geq i \geq k_{max}$) of the buffer that minimizes the overhead cost per frame.

- FIFO MAX POOLING. The first frame in the queue is necessarily sent at each transmission. The algorithm chooses the combination of the remaining frames of the buffer that minimizes the overhead cost per frame to complete the jumbo frame. For OFDMA, only the first frame for each destination is considered for potential transmission (there is no reordering for a given destination).

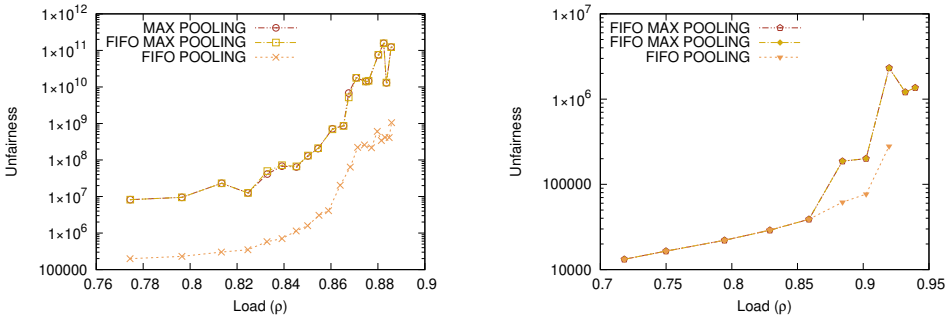
- MAX POOLING. The overhead cost per frame is computed for all possible combinations of frames. The combination with the smallest overhead cost per frame is sent. As for FIFO MAX POOLING, when considering OFDMA, only the first frame of each destination is considered.

The computation of the overhead cost per frame for a given combination is bound by a constant. For OFDMA, the possible combinations are given by the standard and are bound. Let n be the number of frames in the buffer. The complexity of selecting the next transmission frames is $O(n)$ for FIFO



(a) Auditorium trace (b) Set-top Box trace.

Figure 9: Sojourn time for the two traces.



(a) Auditorium trace (b) Set-top Box trace.

Figure 10: Unfairness for the two traces.

POOLING as the buffer is processed in the FIFO order. The complexity of the two other disciplines is $O(n + 2^N)$. Indeed, the possible aggregations are evaluated with a complexity of C x n (where C is a constant). For OFDMA, each destination's first frame is considered a potential candidate. The complexity is then proportional to $\sum_{i=1}^{N} \binom{N}{i} = 2^N - 1$.

5.1 Simulations

In order to consider realistic traffic, we captured Wi-Fi frames in two different environments. A first capture of $53,747$ packets was made in the Auditorium of USTH in Hanoi for 600 seconds with 20 students connected to the school's Wi-Fi network during a lecture. The second traffic capture was made in a private apartment in Hanoi with a typical family connected to the Wi-Fi (their set-top box). The capture shows six devices connected to the network with classical applications (YouTube, Netflix, and video calling) leading to $461,191$ packets. The two captures are available here [21]. We get Wi-Fi 4 (IEEE 802.11n) and Wi-Fi 5 (IEEE 802.11ac) in these two captures. So, to perform the simulations

in Wi-Fi 6, we map the MCS of these previous generations to the one used in Wi-Fi 6. The other properties of the frames have not been changed. These two captures have been replayed to generate the input traffic in our simulator. In order to vary the load in our simulations, we applied a scaling factor to the inter-frame time. Note that the system load is not proportional to this scaling factor as it depends on the service discipline, as explained earlier in this paper.

Sojourn time In Figure 9, we plotted the obtained sojourn time for the three disciplines. The FIFO POOLING discipline cannot reach the same level of load as it is less efficient in terms of overhead. For the Auditorium trace, the sojourn time starts at approximately $1ms$ for a load of 0.76 and can reach up to $60ms$ for the highest load. Also, we can observe that the three disciplines present equivalent behavior and that sometimes FIFO POOLING is even the best despite its simplicity. The sojourn time is less for the Set-Top Box trace due to the limited number of destinations that favor aggregation, which presents a better overhead cost per frame compared to OFDMA.

Unfairness metric In order to evaluate the unfairness between the different destinations brought by each discipline, we plot in Figure 10 the unfairness metric defined in Equation 10. As for the theoretical model in Section 4, the fact to consider a non-FIFO discipline to minimize the overhead cost per frame increases the delay experienced by certain destinations. For the auditorium trace, it is particularly significant and observable, whatever the simulated load level. For the Set-Top Box trace, unfairness appears only for high load.

The qualitative results for these practical scenarios are very different compared to the ones of the perfect OFDMA. Minimizing the overhead cost per frame increases the system capacity; more precisely, the maximum load is reached for a higher input, but to the detriment of the fairness between the destinations. The discipline that offers the best trade-off is FIFO POOLING. It offers an equivalent capacity level with a significant difference in fairness. FIFO MAX POLLING discipline, which offered a good trade-off in the other scenarios, leads to the same unfairness as MAX POOLING. It is explained by the traces that are more bursty than the earlier traffic distribution. Aggregation is then used often and tends to favor certain destinations.

6 Conclusion

OFDMA and aggregation are two important features that benefit from the higher transmission rates of the recent Wi-Fi standards. It involves the selection of frames in the buffer that will be pooled together through a unique transmission. This selection must keep a low complexity and not introduce unfairness that could increase the delay for certain frames. We have shown that the service discipline that aims to maximize the capacity of the system introduces unfairness that can be significant in certain conditions. We have empirically shown that simple service disciplines that select systematically the first frame in the

queue offer a good trade-off between capacity and fairness. We have proposed a practical implementation based on the recent standard 802.11ax of these service disciplines with a metric able to evaluate the efficiency of a combination of frames and carriers allocation. We have also shown that these stateless service disciplines may be implemented with a very low level of resources in terms of complexity and memory.

A natural continuation of this work is to adapt and evaluate these algorithms for the other Wi-Fi standards (IEEE 802.11be, IEEE 802.11ah, for instance) and to propose theoretical bounds on load, delay, or fairness. Another possible extension is to take into account more complex Wi-Fi environments and study their impact on the performance: loss of frames, cohabitation with other legacy Wi-Fi, MCS changes for jumboframes, etc.

References

[1] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi, "A tutorial on ieee 802.11ax high efficiency wlans," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 197–216, 2019.

[2] B. Bellalta, "Ieee 802.11ax: High-efficiency wlans," *IEEE Wireless Communications*, vol. 23, no. 1, pp. 38–46, 2016.

[3] F. Abinader, S. Choudhury, V. de Sousa, and et al., "Distributed wi-fi interference coordination for dense deployments," *Wireless Personnal Communication*, vol. 97, pp. 1033–1058, 2017.

[4] S. Kuppa and G. Dattatreya, "Modeling and analysis of frame aggregation in unsaturated wlans with finite buffer stations," in *2006 IEEE International Conference on Communications*, vol. 3, pp. 967–972, 2006.

[5] C. A. Grazia, "A performance model for wi-fi frame aggregation considering throughput and latency," *IEEE Communications Letters*, vol. 24, no. 7, pp. 1577–1580, 2020.

[6] N. El Houda Bouzouita, A. Busson, and H. Rivano, "Fam: A frame aggregation based method to infer the load level in ieee 802.11 networks," *Computer Communications*, vol. 191, pp. 36–52, 2022.

[7] D. Bankov, A. Didenko, E. Khorov, and A. Lyakhov, "Ofdma uplink scheduling in ieee 802.11ax networks," in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2018.

[8] D. Bankov, A. Didenko, E. Khorov, V. Loginov, and A. Lyakhov, "Ieee 802.11ax uplink scheduler to minimize, delay: A classic problem with new constraints," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–5, 2017.

[9] G. Naik, S. Bhattarai, and J.-M. Park, "Performance analysis of uplink multi-user ofdma in ieee 802.11ax," in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2018.

[10] K.-h. Lee, "Performance analysis of the ieee 802.11ax mac protocol for heterogeneous wi-fi networks in non-saturated conditions," *Sensors (Basel, Switzerland)*, vol. 19, 03 2019.

[11] E. Avdotin, D. Bankov, E. Khorov, and A. Lyakhov, "Ofdma resource allocation for real-time applications in ieee 802.11ax networks," in *2019 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, pp. 1–3, 2019.

[12] G. Z. Islam and M. A. Kashem, "A proportional scheduling protocol for the ofdma-based future wi-fi network," *Journal of Communications*, vol. 17, no. 5, 2022.

[13] M. Kuran, A. Dilmac, Topal, B. Yamansavascilar, S. Avallone, and T. Tugcu, "Throughput-maximizing ofdma scheduler for ieee 802.11ax networks," in *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–7, 2020.

[14] D. Magrin, S. Avallone, S. Roy, and M. Zorzi, "Performance evaluation of 802.11ax ofdma through theoretical analysis and simulations," *IEEE Transactions on Wireless Communications*, vol. 22, no. 8, pp. 5070–5083, 2023.

[15] D. R. Smith, "Technical note - a new proof of the optimality of the shortest remaining processing time discipline," *Oper. Res.*, vol. 26, pp. 197–199, 1978.

[16] A. Wierman and M. Harchol-Balter, "Classifying scheduling policies with respect to unfairness in an m/gi/1," *SIGMETRICS Performance Evaluation Review*, vol. 31, p. 238–249, jun 2003.

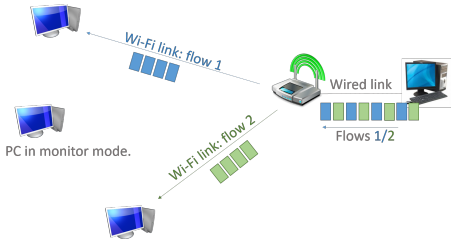
[17] B. Avi-Itzhak and H. Levy, "On measuring fairness in queues," *Advances in Applied Probability*, vol. 36, no. 3, pp. 919–936, 2004.

[18] W. Sandmann, "A discrimination frequency based queueing fairness measure with regard to job seniority and service requirement," in *Next Generation Internet Networks, 2005*, pp. 106–113, 2005.

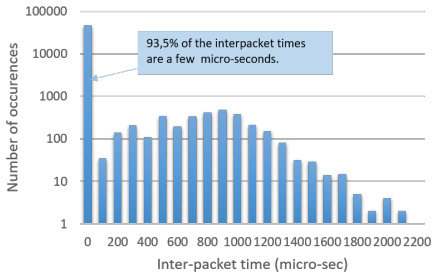
[19] W. Sandmann, "Analysis of a queueing fairness measure," in *13th GI/ITG Conference - Measuring, Modelling and Evaluation of Computer and Communication Systems*, pp. 1–13, 2006.

[20] J. Little, "A proof of the theorem $l = \lambda w$," *Operations Research*, vol. 9, pp. 383–387, 1961.

[21] A. Busson and A. T. Giang, "Pooling code." https://github.com/anthonybusson/poolingCode.git, 2023.



(a) Experiment: a sender sends two nested flows to two Wi-Fi destinations.



(b) Distribution of the inter-packets time measured at the application level (not normalized).

Figure 11: Experiment conducted to infer the aggregation algorithm used by current Wi-Fi products and the frames delivering at the application layer.

7 Appendix: Experiments to infer the aggregation behavior.

AP	#frames	#aggregated	Mean	Maximum
PC	44155	6436	6.86	47
AP Linksys	32169	3748	8.58	11
AP Belkin	35007	669	52.32	64

Table 3: Experiment results. The fields are the total number of frames (#frames), the number of aggregated frames (#aggregated), the mean number of frames that compose an aggregated frame (Mean), and the maximum number of frames observed in an aggregated frame (Maximum).

An experiment has been conducted to confirm two assumptions made in the paper: i) current Wi-Fi products aggregate all the frames to the same destination, whatever their positions in the buffer, ii) the frames that compose an aggregated frame are delivered to the OS/application once the aggregated frame has been fully received.

Scenario. The experiment is illustrated in Figure 11a. A PC on the wired network sends one different flow to each station (the two PC on the Wi-Fi network). The packets of the two flows are perfectly interlaced: the source transmits one packet for station 1, then one for station 2, and so on. The IP Packet length is 1024 bytes. We measure the reception time of each packet at the application layer on each station. Besides, a PC in monitor mode captures Wi-Fi traffic. We tested three different AP: Linksys LAPAC 1750 (IEEE 802.11n and ac), Belkin AX 32000 RT3200 (IEEE 802.11n/ac/ax), and one PC configured with hostapd. All the PC are the same (the two Wi-Fi stations, the one in

monitor mode, and one of the AP) with the following hardware: motherboard APU 4C4, with Qualcomm Atheros QCA986x/988x 802.11ac Wi-Fi card and Atheros driver (ath10k). The system on the PCs is a Debian 10 (kernel 4.19.0-18-amd64). The AP and the stations used the IEEE 802.11ac amendment (Wi-Fi 5) with channels in the 5 GHz band and configured to use a maximal channel width of 80 MHz.

Results. We first analyze aggregation performed by the APs. It has been obtained from the capture on the PC in monitor mode. Even if the packets are interlaced in the AP buffer, the results show that the three tested AP pool the frames intended for the same destination in aggregated frames. The discipline is thus not FIFO. Table 3 shows the results for each AP. The maximum size of the aggregated frame differs significantly from one AP to another (from 11 to 64), which impacts the mean aggregated frame size. The capacity of certain AP to support high load is consequently not the same for all APs. It also appears that the APs never change the initial packet order for the same destination.

Besides, we analyze the reception time at the application level on the two Wi-Fi stations. Figure 11b shows the Distribution of the inter-packets reception time for one of the two stations when the AP is the PC. We observe that when two packets are in the same aggregated frame, the inter-packet time is only a few microseconds (mostly between 4 and 8), and the time between two consecutive frames that belong to two different aggregated frames is between 200 and 1000 microseconds. This observation also holds for the second station and whatever the AP. It empirically proves that the delivery of the frames by the card is done once the full aggregated frame is received. It significantly increases the time to deliver a packet through the Wi-Fi network. For instance, for these experiments, the delivery delay can reach up to 10 milliseconds for the AP that aggregated the most (AP Belkin).

8 Appendix: Proofs.

8.1 Proof of Properties 3 and 4

We prove that $\rho = \mu OV + \sum_{i=1}^{N} \lambda_i D_i$. The proof is quite trivial and relies on the ergodic property of the system.

By definition, ρ is the busy time, i.e., the proportion of time where the system is not empty or, equivalently, the proportion of time where the server is idle.

$$\rho = \lim_{T \to +\infty} \frac{1}{T} \int_0^T \mathbb{1}_{X(t)>0} dt \tag{13}$$

It can also be expressed as the proportion of time the server is idle. We get:

$$\frac{1}{T}\int_0^T \mathbb{1}_{X(t)>0}dt = \frac{1}{T}\sum_{i=1}^{J(T)}\Big(OV + \sum_{j=1}^{k_i} t_{j,i}\Big) + \frac{\epsilon}{T} \tag{14}$$

where $J(T)$ is the number of jumboframes sent between 0 and T, k_i is the number of frames in the i^{th} jumboframe, and $t_{j,i}$ is the time to send the j^{th} frame that composes the i^{th} jumboframe.

ϵ is a variable that describes the transmission time of a potential frame that is being transmitted at time T and for which the transmission is not finished. It will become negligible when $T \to +\infty$.

The sum $\sum_{j=1}^{k_i} t_{j,i}$ can be rewritten considering the different destinations.

$$\frac{1}{T}\sum_{i=1}^{J(T)}\Big(OV \quad + \quad \sum_{j=1}^{k_i} t_{j,i}\Big) + \frac{\epsilon}{T} = \frac{J(T)OV}{T} + \frac{1}{T}\Big(\sum_{l=1}^{N}\sum_{j=1}^{n_l(T)} t_l^j\Big) + \frac{\epsilon}{T} \tag{15}$$

$$= \quad \frac{J(T)OV}{T} + \sum_{l=1}^{N}\Big[\frac{n_l(T)}{T}\frac{1}{n_l(T)}\sum_{j=1}^{n_l(T)} t_l^j\Big] + \frac{\epsilon}{T} \tag{16}$$

where t_l^j is the j^{th} frame to destination l sent during the period $[0, T]$, and $n_l(T)$ is the total number of frames sent to destination l during the period $[0, T]$.

As the system is ergodic, when T tends to infinity, $\frac{J(T)OV}{T}$ tends to μ, $\frac{n_l(T)}{T}$ tends to λ_l (as the process is stationary the mean number of arrivals per second asymptotically equals to the mean number of departures), and $\frac{1}{n_l(T)}\sum_{j=1}^{n_l(T)} t_l^j$ tends to D_l. It proves the property.

To prove Property 4, we compute the asymptotic number of frames that leave the system per second. We denote τ_k, the number of frames that composes the k^{th} jumbo frame. We get,

$$\frac{1}{T}\sum_{k=1}^{J(T)}\tau_k = \frac{J(T)}{T}\frac{1}{J(T)}\sum_{k=1}^{J(T)}\tau_k \tag{17}$$

$\frac{J(T)}{T}$ tends to μ as T tends to infinity and $\frac{1}{J(T)}\sum_{k=1}^{J(T)}\tau_k$ tends to τ. Consequently, the asymptotic departure rate converges to $\tau \cdot \mu$. As the system is stationary, this quantity is also equal to the mean number of frames that enter the system per second $\sum_{i=1}^{N}\lambda_i$.

8.2 Proof of Property 5

The mean service time is deduced from Little's formula applied to the server. The server can be idle or busy and contains either 0 frame/jumbo frame or 1 if it is busy. We get,

$$
\begin{aligned}
I\!E[X_{serveur}] &= 0 \cdot I\!P(X_{serveur} = 0) + 1 \cdot I\!P(X_{serveur} = 1) && (18)\\
&= \rho && (19)
\end{aligned}
$$

The mean sojourn time in the server is the service time of the jumbo frame $I\!E[S]$, and the input rate is μ. Little's formula leads to the results.

$$\rho = \mu I\!E[S] \tag{20}$$