A Flow-Level Wi-Fi Model for Large Scale Network Simulation

Clément Courageux-Sudan, Loic Guegan, Anne-Cécile Orgerie, Martin Quinson

A Flow-Level Wi-Fi Model for Large Scale Network Simulation

Clément Courageux-Sudan
Univ. Rennes, Inria, CNRS, IRISA
Rennes, France
clement.courageux-sudan@irisa.fr

Loïc Guegan
UiT The Arctic University of Norway
Tromso, Norway
loic.guegan@uit.no

Anne-Cécile Orgerie
Univ. Rennes, Inria, CNRS, IRISA
Rennes, France
anne-cecile.orgerie@irisa.fr

Martin Quinson
Univ. Rennes, Inria, CNRS, IRISA
Rennes, France
martin.quinson@irisa.fr

ABSTRACT

Wi-Fi networks are extensively used to provide Internet access to end-users and to deploy applications at the edge. By playing a major role in modern networking, Wi-Fi networks are getting bigger and denser. However, studying their performance at large-scale and in a reproducible manner remains a challenging task. Current solutions include real experiments and simulations. While the size of experiments is limited by their financial cost and potential disturbance of commercial networks, the simulations also lack scalability due to their models' granularity and computational runtime. In this paper, we introduce a new Wi-Fi model for large-scale simulations. This model, based on flow-level simulation, requires fewer computations than state-of-the-art models to estimate bandwidth sharing over a wireless medium, leading to better scalability. Comparing our model to the already existing Wi-Fi implementation of ns-3, we show that our approach yields to close performance evaluations while improving the runtime of simulations by several orders of magnitude. Using this kind of model could allow researchers to obtain reproducible results for networks composed of thousands of nodes much faster than previously.

CCS CONCEPTS

• **Networks** → **Network simulations**; **Wireless local area networks**.

KEYWORDS

Performance evaluation; modeling and simulation; Wi-Fi networks

ACM Reference Format:
Clément Courageux-Sudan, Loïc Guegan, Anne-Cécile Orgerie, and Martin Quinson. 2022. A Flow-Level Wi-Fi Model for Large Scale Network Simulation. In *Proceedings of Proceedings of the International Conference on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWiM '22).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3551659.3559022

1 INTRODUCTION

Wireless standards such as IEEE 802.11n are extensively used in modern networks, whether by end-users, or application services deployed at the edge of networks. This leads to an increased share of wireless communications, estimated to be 71% of the IP traffic by 2022 [3], and to denser wireless networks. While Wi-Fi networks have been extensively studied to improve their throughput, security, or reliability, obtaining reproducible and realistic performance measurements of Wi-Fi networks remains a challenging task, especially when studying large-scale infrastructures.

There are currently two different approaches to conduct Wi-Fi performance evaluations. The former consists in running experiments. However, the hardware costs for large-scale experiments rapidly increase, as well as the difficulty of controlling all experimental conditions to ensure the credibility of the results [5, 17]. The latter is to use simulation. Existing simulation models provide estimations of metrics such as data throughput, packet error rate (PER), or signal-to-noise ratios (SNR), but their scalability remains an issue [15, 16]. While recent works improved the accuracy of the simulation outputs [19], the amount of computations to obtain these results also increased. Available Wi-Fi models from the literature cannot scale over a few dozens of nodes without a simulation runtime exceeding several hours [10].

In this paper, we model the performance of Wi-Fi communications with a focus on scalability, while keeping coherent estimations. We propose a new simulation model validated against IEEE 802.11n performance metrics obtained using ns-3. Based on flow-level simulation, this model differs from existing works by adopting a more abstract representation of communications. Flow-based models have been used successfully in the past to simulate large-scale wired networks [25] with a significant simulation runtime improvement compared to more fine-grained approaches. To the best of our knowledge, this approach has not been applied to Wi-Fi networks prior to this work. This model enables the observation of the Wi-Fi throughput of network infrastructures and the bandwidth associated with single flows. We show that our model produces accurate estimations under various configurations: Modulation and Coding Schemes (MCS), Request / Clear To Send mechanism (RTS/CTS), and datarates. It could be leveraged to study the performance of Wi-Fi networks at scale, such an evaluation was hard to perform before given the cost of large-scale experiments and simulations.

This paper is organized as follows. Section 2 introduces already existing Wi-Fi models, their advantages along with their limitations. In Section 3, we describe the goals and hypotheses underlying this work and the framework on which we build our model. Section 4 describes our Wi-Fi bandwidth sharing model. In Section 5 we explain the methodology we follow to calibrate this model to fit experimental conditions. After validation in Section 6, Section 7 concludes this work.

MSWiM '22, October 24–28, 2022, Montreal, QC, Canada
2022. ACM ISBN 978-1-4503-9479-6/22/10…$15.00
https://doi.org/10.1145/3551659.3559022

2 RELATED WORK

The Wi-Fi standards and their limitations have been extensively studied in the literature [27]. Work has been done to study signal propagation under various conditions [23] or to design realistic error models [19]. Other works analyze the energy consumption of wireless communications [21].

To this end, Wi-Fi performance analysis is possible, either using purely analytical models or with the help of simulators. Analytical approaches such as [4] give accurate throughput estimations for Wi-Fi communications. However, these models need to be instantiated manually and cannot scale to large networks. Other works aim at implementing simulation models such as the IEEE 802.11 models of ns-3 [11], OMNet++ [24] and OPNET [8]. Ns-3 is a discrete-event simulator with packet-level models. In this paradigm, every network packet sent over the network, routing and protocol stack mechanisms are simulated in details. Ns-3 802.11 models have been validated in a wide range of scenarios and extended to fit most of the standards defined up to date [19],[12].

Existing packet-level and analytical models have shown to lead predictions matching the results obtained with IEEE TGn and other low-level Wi-Fi models [19]. But the granularity of these approaches has a drawback: the performance of packet-level models is impacted by the size of the simulated network and the number of network messages. In [10], simulations based on ns-3 have a runtime exceeding 11 hours to simulate 250 communication flows, and evolve linearly with the number of packets to simulate. As we show in Section 6, ns-3 models can hardly simulate more than a hundred Wi-Fi stations. Existing packet-based simulators are thus limited to studying size-limited infrastructures because of their design.

Packet-based simulation is not the only solution to simulate network communications. Link simulation consists in studying network transmissions at signal-level [20]. Link simulation is used to study the physical layer of Wi-Fi by modeling all signal properties. But the granularity of link models limit their use to single link communication.

Another approach is flow-level simulation. While a packet-level model represents each network packet, in a flow-level model communication between two nodes are represented by a single flow entity. The throughput of each flow is then computed by solving an inequation system. More details about flow-based models are presented in Section 3.2. Existing simulators such as SimGrid use flow-level models [7]. Despite the simplicity of this approach, it has been shown that communications within large wired networks can be simulated with a reasonable accuracy [25]. Yet, to the best of our knowledge, no such model has been proposed for IEEE 802.11 networks.

In the rest of this paper, we introduce a flow-based model to greatly improve the scalability compared to packet-based Wi-Fi simulation without trading for accuracy. This model allows for performant and accurate simulations even with networks composed of thousands of nodes.

3 CONTEXT

Section 3.1 explains the goals and hypotheses underlying this work, while the framework used to build our flow-based model is introduced in Section 3.2.

3.1 Wi-Fi channel access

The IEEE 802.11 standards define the rules that must be followed by Wi-Fi nodes (stations and access points) to communicate in a network. They define the frequencies available, channel sizes, and coding schemes, among other properties impacting the communication throughput. The Distributed Coordination Function (DCF) defines the access sharing of the wireless channel between the stations (STA) that need to send data and are connected to an Access Point (AP). DCF determines the time and the amount of data each station will send over the channel. The DCF relies on different mechanisms to avoid collisions between several STAs that would otherwise start communicating at the same time. Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) is a MAC layer protocol stating that every STA must sense an IDLE channel for a duration called Distributed Inter Frame Space (DIFS) before transmission. After sensing the channel as IDLE, an exponential backoff requires STAs to wait for an additional amount of time before sending their data. This duration is randomly chosen and multiplied by two after each channel access failure. Using these mechanisms, the channel should be accessible for all stations fairly while ensuring both a low collision probability and a high channel usage.

Modeling this DCF is mandatory to simulate the sharing of the Wi-Fi channel between STAs. Our model relies on the following hypotheses. Complex features of modern 802.11 standards, such as energy management from 802.11ax [26] are not covered by the current design. This paper also does not cover complex phenomenons such as multipath fading and inter-cell interferences as these phenomenons depend on information that is not available using flow-based approaches. We leave mobility for future work.

Despite these hypotheses, the model we introduce can still simulate various scenarios: different MCSs, RTS/CTS activated or not, distance to the AP, and heterogeneous throughputs. Obtaining accurate estimations then depends on a thorough process as we will see in Section 5.

3.2 Flow model framework

As introduced in Section 2, obtaining a scalable flow-based Wi-Fi model requires switching to a more abstract representation of communications than packet-level models. Packet-level simulators consider the details of every packet transmitted between nodes, each of them requiring computations to estimate the probability of its successful reception and its transmission time. Using flow-level simulation, the communication between two nodes of a network is represented as a single flow. The bandwidth allocated to a flow depends on the properties of the link, and on the other flows in the network. This is similar to a hydraulic system where water flows depending on the size of the pipes, and the amount of water sent in each pipe. Flows have a constant throughput between network events such as the arrival or deletion of a flow. Upon network events, the bandwidth of all flows is updated to match the new state of the network. This approach is much less compute-intensive than packet-level simulation since the number of packets is typically much higher than the number of events.

In this context, designing a flow-level model consists in defining the properties of each flow, along with the rules to update the bandwidth of flows upon network events. This is done through the use of a solver that takes the state of the network as an input, modeled as an inequation system, and outputs the bandwidth allocated to each active flow.

The inequation system is defined using variables and constraints similar to Equation 1.

Computing the bandwidth of network flows can be done as follows. Each link of the network is represented by one inequation r associated with a constraint C_r. The inequation is used to compute the throughput of all the flows passing through this link. The variable ρ_i represents the bandwidth allocated to the flow i. The constraint C_r ensures that the sum of the bandwidths of all the flows passing through link r will not exceed the link's capacity. C_r thus corresponds to the maximum throughput of the link. Coefficients $a_{r,i}$ can be used for each variable to model behaviors such as backward acknowledgments as their usage happens to reduce the link's available capacity. Weights w_i can be used to prioritize some flows and model phenomenons such as RTT-Fairness (when the bandwidth of a node decreases inversely proportionally to its round trip time).

$$\begin{cases} \sum\limits_{\substack{i \text{ using} \\ \text{link } 1}} a_{1,i} * \rho_i \leq C_1 \\ \dots \\ \sum\limits_{\substack{i \text{ using} \\ \text{link } r}} a_{r,i} * \rho_i \leq C_r \\ \dots \\ \sum\limits_{\substack{i \text{ using} \\ \text{link } m}} a_{m,i} * \rho_i \leq C_m \end{cases} \tag{1}$$

Solving the system consists in determining the values of variables ρ_i for all flows, i.e. the bandwidth associated to flow i. This can be done iteratively, starting from the most constrained link until all throughputs are determined. The load of the link associated to inequation r can be noted $\epsilon_r = C_r / (\sum_{i \text{ using } r} \frac{a_{r,i}}{w_i})$. Consequently, the most constrained inequation of the system is the one that minimizes ϵ_r. Solving that inequation is done by computing $\rho_i = \epsilon_r / w_i$ for each flow i traversing the link corresponding to this inequation. Once an inequation is fixed, constraints C_r through which passes flow i are updated such that $C'_r = C_r - a_{r,i} * \rho_i$. Then, the same operation is iterated over the modified system until all flows are fixed.

This kind of solver is used in the SimGrid simulation framework [7], and in other works [9]. More details about the implementation of this solver in SimGrid called the Linear Max-Min solver (LMM), and how to apply it to wired communication modeling can be found in [14].

4 A FLOW-LEVEL WI-FI MODEL

This section describes the design of our Wi-Fi model. Section 4.1 designs the inequation constraints to model the bandwidth sharing of a Wi-Fi link. Sections 4.2 and 4.3 extend the base model to take more phenomenons into account: flow concurrency and position of stations.

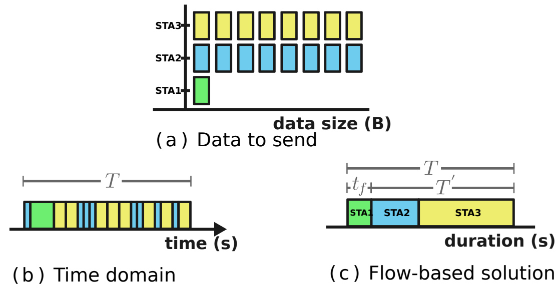


Figure 1: Example of Wi-Fi communication over a period T with 3 STAs where $d_1 = 4d_2 = 2d_3$

4.1 Wi-Fi bandwidth sharing

Our model needs to reproduce the behavior of the DCF, i.e., managing the access of STAs to the wireless channel. In the following, the channel is supposed to be in ideal conditions with no interferences, no signal loss, and no hidden node. The two factors that play a major role in the way the channel is shared among stations are: *a)* r_i, the datarate associated with station STA_i which depends on AP and STA configurations, *b)* d_i the amount of data to be sent by STA_i. For conciseness, in the rest of this paper, we use the term channel to describe both the nodes (STA, AP) and the channel of a Wi-Fi cell.

The model's design is illustrated using the example of Figure 1 where three STAs want to send data concurrently on the same channel over some time T. The data each of the three stations need to send is shown in Figure 1a i.e. STA_1 and STA_2 send 8 packets, while STA_3 sends 1 packet. Figure 1b shows one of the possible share of the channel in this use case. This is the results that could be obtained for instance using packet-level models.

Our goal is to compute the time each STA spends sending data over T, as shown in Figure 1c. The use of the Wi-Fi link for period T is expressed as $C = \frac{\sum_{i=1}^{n} d_i}{T}$. Because of the exponential backoff described in Section 3.1, all nodes have the same probability of accessing the channel and transmitting frames. This leads to a fair share of the channel regarding the amount of data sent by each station. Theoretically, all stations will transmit the same amount of data, $\forall i, j, d_i = d_j = d$, where d_k is the amount of data sent by STA_k during T. As a consequence, the throughput of all nodes during T is equal to the same value, $\forall i, j, \rho_i = \rho_j = \rho$.

Therefore, we define the usage of link C in our inequation system as a fair share among all active flows:

$$C = \sum_{i=0}^{n} \rho_i = \frac{n * d}{T} = \frac{n * d}{\sum_{i=0}^{n} \frac{d}{r_i}} = \frac{1}{\frac{1}{n} \sum_{i=0}^{n} \frac{1}{r_i}} \tag{2}$$

However, some of the flows might be fixed by other constraints of the system. Suppose the throughput of the flow starting from STA_1 (noted ρ_1) is fixed such that $\rho_1 \leq \frac{C}{n}$ because the receiver of the flow is slower than STA_1 can transmit. Because STA_1 does not take all its fair share of airtime, the other flows will be shared with the remaining link capacity. We define C', the remaining capacity of the channel once the fixed flow has been removed. Let d_f be the amount of data sent by the fixed flow such that $d_f \neq d$. Since

$t_f = \frac{d_f}{r_f}$ an $\rho_f = \frac{d_f}{T}$, we have:

$$T = T' + t_f = T' + \frac{d_f}{r_f} = T' + T * \frac{\rho_f}{r_f}, \text{hence } T = \frac{T'}{1 - \frac{\rho_f}{r_f}} \quad (3)$$

Let d' be the amount of data to be sent by the remaining stations. Since we have

$$C = \sum_{i=0}^{n} \rho_i = \rho_f + C' = \frac{d_f + (n-1)d'}{T},$$

the capacity of the remaining flows can be computed as:

$$C' = \sum_{i=0, i \neq f}^{n} \rho_i = \frac{(n-1)d'}{T} = \frac{(n-1)d'}{T'} * \left[1 - \frac{\rho_f}{r_f}\right]$$

$$= \frac{1}{\frac{1}{n-1} \sum_{i=0, i \neq f}^{n} \frac{1}{r_i}} * \left[1 - \frac{\rho_f}{r_f}\right] \quad (4)$$

Let I be the set of all the flows not fixed in the system, and F the set of fixed flows, we can generalize Equation 4 to an arbitrary number of fixed flows:

$$C' = \frac{1}{\frac{1}{|I|} \sum_{i \in I} \frac{1}{r_i}} * \left[1 - \sum_{f \in F} \frac{\rho_f}{r_f}\right] \quad (5)$$

Equation 5 defines how to update the inequation system once one or several flows have external constraints limiting their sending capacity. Solving this system is much less computationally intensive than a packet-level approach where each network packet has to go through a whole set of complex models. This model requires few parameters compared to packet-level models: the number and sizes of flows along with the datarate of each station.

4.2 Capacity reduction of concurrent flows

The equations from Section 4.1 enable sharing the bandwidth of a Wi-Fi channel between a set of flows in ideal conditions. This section refines the base model by taking some potential causes of performance degradation on the Wi-Fi channel into account.

One example is the case of collisions between nodes. The more flows are executed concurrently on a single channel, the higher the probability to have two stations sending data at the same time, leading to collisions and retransmissions. This reduces the channel time for successful communications. The exponential backoff and mechanisms such as RTS/CTS try to minimize the probability of such events, but collisions still happen in real networks.

In packet-level simulators, successful transmissions and receptions are computed using protocol-specific propagation and probabilistic loss models. We propose a more minimalistic approach to capture this issue, adapted to a flow model. The main drawback of this approach is that it requires careful calibration depending on the simulated scenario to produce accurate results. More details can be found in Section 6.

We expand the bandwidth sharing model to adjust the constraint associated to the bandwidth of a Wi-Fi link depending on the number of flows active on the link. To compute a performance loss given the number of concurrent flows, we can thus define a function f, such that:

$$C_r = f(\sum_{i \text{ uses } r} 1) \quad (6)$$

This function f can be used to capture collision effects in heavily-loaded Wi-Fi networks. Indeed, we observed an overall stationary throughput in 802.11n cells up to a certain threshold, after which the throughput linearly decreases. Once the threshold and the coefficients of the linear curve are obtained experimentally (either through real or simulated experiments using more fine-grained models), we can define f such that:

$$f(x) = \begin{cases} bw_0 & x < thresh \\ a * x + bw_0 & x >= thresh \end{cases} \quad (7)$$

Where bw_0 is the maximum throughput, and a is the coefficient of the curve. While this calibration process requires some work, once the function has been extracted, it can be reused for several experiments and scale up to thousands of nodes within large network infrastructures. In addition, other definitions of f could be used in the future to model other effects even if we only use it as presented to model packet loss on the channel.

4.3 Propagation model with SNR Levels

This section further extends our base model to allow STAs to have different positions when starting an experiment. STA mobility during the experiments remains out of the scope of this paper. The position of a STA can have an impact on its datarate. Packet-level simulators compute a Signal to Noise Ratio (SNR) for each data frame based on a loss model and use it to deduce a Packet Error Rate (PER) [19].

Again, our approach is more minimalistic than packet-level models: we do not have access to the same amount of information in a flow-level simulator, preventing us to estimate SNR and PER values accurately. Instead, we allow the experimenter to define the list of datarates that we name *SNR levels* for each station in a cell. At runtime, it is possible to define which datarate to use for each STA separately depending on its position. As described in the equations of Section 4.1, the datarate limits the maximum bandwidth of a station if its SNR level is lower than the theoretical maximum throughput of the station.

5 MODEL IMPLEMENTATION AND CALIBRATION

5.1 Implementation

We have implemented the flow-based model described in Section 4 for the bandwidth sharing of a channel between the nodes of a Wi-Fi cell within the SimGrid simulation framework [7]. SimGrid is a flow-level simulation framework for distributed systems proposing various models for Ethernet communication, CPU, and disk usage, among others. However, no Wi-Fi model was available within SimGrid prior to this work. We chose SimGrid for the efficiency of the integrated inequation solver, and the overall software framework that allows us to focus on the Wi-Fi modeling. Transposing the equations of Section 4 in this solver is straightforward. SimGrid being open source software, this model is integrated within SimGrid's source tree and will be available in all future release archives [22].

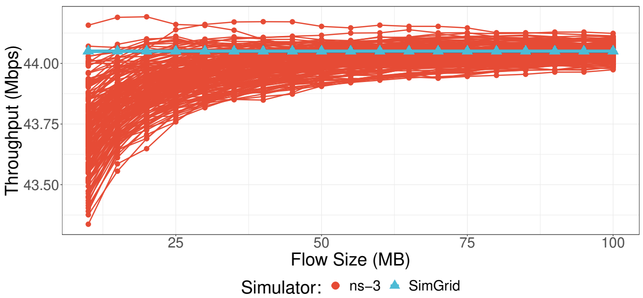


Figure 2: Single station throughput given flow size with 100 different seeds for ns-3 in red.

Adding this model to SimGrid consists of the creation of a new type of communication link specific to Wi-Fi. Following SimGrid's terminology, a Wi-Fi link corresponds to the Wi-Fi channel and the nodes attached to this channel. In the solver, specific equations are used for such links. These inequations are updated using the equations of Section 4 instead of the historical wired model of SimGrid. Wired and wireless links are compatible and can be used jointly in a simulation.

5.2 Calibration

As noted earlier, the flow model must be carefully calibrated to obtain accurate performance estimations. In particular, the bandwidth sharing between several channels is highly dependent on the experimental scenario under study. The configuration of STAs, the density of nodes in the network, and external channel degradation have all an impact on the capacity of a Wi-Fi cell. Our model cannot evaluate the impact of those effects, thus we rely on a calibration of parameters before executing simulations. Calibration values can be obtained through real measurements or microbenchmarks on more fine-grained simulators such as ns-3. The rest of this section describes these different calibration steps.

5.2.1 Single station throughput: An IEEE 802.11 channel consists of more than applications' data frames. Control frames, beacons, RTS and CTS messages make regular use of the Wi-Fi channel. This time spent managing the cell impacts the throughput available for application data compared to the theoretical maximum available throughput. The flow model does not account for the time spent sending such frames, and thus needs to be calibrated to fit that maximum throughput. This value can be obtained by running microbenchmarks to obtain the maximum data throughput of a station communicating to its access point and calibrating SimGrid with this value.

This is done by running an ns-3 simulation consisting of one Wi-Fi cell made of an AP and one STA using an MCS value of 3. A single flow is created from the STA towards the AP in this cell. The simulation is repeated several times with 100 different random seeds. Figure 2 shows the results of this experiment where red points are the throughput of ns-3 for different flow sizes. With this configuration, the maximum flow throughput converges on average around 44.1Mbps.

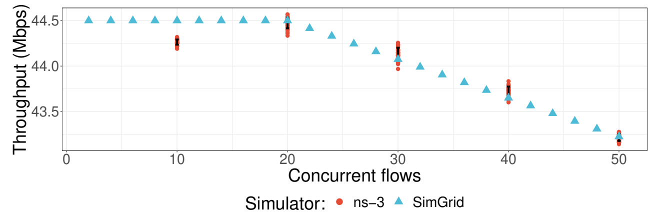


Figure 3: Throughput degradation under concurrent flows.

By calibrating SimGrid's model with this value (i.e. approximately 44.1Mbps), we obtain the blue line of Figure 2. SimGrid estimations do not depend on random number generation, thus all experiments will give the same outputs in contrast to ns-3. The throughput observed for small-sized flows is slightly decreased by approximately 1%. Our goal is to simulate large traffic and many nodes experiments, thus we ignore this phenomenon in the rest of this paper.

5.2.2 Bandwidth reduction function: This step focuses on the throughput available for stations depending on the number of concurrent flows in the network.

Various amounts of concurrent flows are simulated in a cell, where a different number of stations try to communicate simultaneously. In this case, flows towards the AP (ascending) and from the AP to the STAs (descending) are mixed. Pairs of STAs are created and communicate from one node to the other, passing by the AP. We run similar experiments with exclusively ascending or descending flows, leading to similar results. The results obtained with ns-3 are shown in Figure 3.

This experiment highlights a new phenomenon that was not observable in the previous calibration step. The throughput available to communicate decreases with the number of concurrent flows. This happens regardless of the MCS configuration used for the experiments as shown in Section 6. This decrease can be explained by several factors: more concurrent flows require more management frames and RTS/CTS requests if activated (in this case, RTS/CTS is used for packets above 100 bytes), and it also increases the probability of having collisions when STAs start sending data at the same time. Even without collisions, already communicating nodes slow down the others, which need to increase their backoffs. This issue is accounted through the link capacity degradation mechanism described in Section 4.2.

Using the observations of ns-3, the function f is designed as a piecewise function, using a constant maximum throughput below 20 concurrent flows before linearly decreasing the maximum throughput once this threshold is exceeded. A linear regression has been done to estimate the throughput degradation related to the amount of concurrency. Once these values are obtained, we inject them into SimGrid to obtain the same effect in our model.

5.2.3 Distance to the access point: The last benchmarks we used to calibrate our model consist in observing the maximum distance between an AP and the STAs that communicate with it. It can be done by running an experiment consisting of a Wi-Fi cell with one AP and a single STA. Several simulations are done, where the throughput obtained between the AP and the STA is observed depending on the distance between the two nodes. Once we know the maximum communication distance, we can set the datarate of STAs according to the values obtained experimentally, i.e. any STA located above the maximal distance will be assigned a datarate of 0bps. Rate adaptation algorithms [13] are out of the scope of this paper, along with mobility during the experiment. In our validation, STAs have a fixed position, defined at the beginning of the experiment.

Once the calibration has been done, the flow model can simulate scenarios with several Wi-Fi cells mimicking the realistic measurements, and scale up to thousands of nodes, which cannot be done using packet-level approaches.

6 PERFORMANCE EVALUATION

This section evaluates the flow-based model, as implemented within SimGrid. As a baseline to compare the metrics provided by our model, we compare the outputs of SimGrid to the outputs of ns-3. This choice to compare our flow model to another simulator is motivated by two factors: *a)* real experiments using wireless devices are very hard to reproduce at scale, *b)* ns-3 is extensively used for Wi-Fi simulation. Much work has been performed to obtain realistic predictions using the many available Wi-Fi models.

Different metrics are compared to validate our approach, such as the overall Wi-Fi cell throughput and single flow throughputs or performance metrics for a wide selection of use cases. In the following, we first assess the validity of our model on simulations of limited scale, before assessing the scalability of our approach on a realistic infrastructure.

Experimental Setup. Experiments are using SimGrid v3.29 modified to implement our model as described in Section 4, whereas we use an unmodified ns-3 v3.33 with 802.11n models. Random number generation within ns-3 has shown to lead to very different results depending on the random seed used to initialize the simulations. This is due to the probabilistic nature of some phenomenons such as drawing random backups or signal propagation, among others. Experiments are executed on the Grid'5000 [2] testbed to sample executions using different seeds. Regarding STAs positions, we uniformly put the stations within a circle of radius of 15 meters around the AP when not explicitly written. This mitigates issues such as the hidden node problem [18].

Source code and reproducibility. Experiment artifacts, including code, scripts, and visualizations used for this paper, and additional results are available at https://github.com/klementc/wifi-reproducibility

6.1 Small-scale validation through microbenchmarks

In the following, we study the capacity of our model to give accurate performance estimations on limited-size platforms.

6.1.1 Using the throughput reduction mechanism: We observe the impact of the number of concurrent flows using different Wi-Fi modulation and coding schemes (MCS). Figure 4 show the overall throughput of a cell using four MCS configurations (MCS 2 to 5). First, we run microbenchmarks to calibrate our model with the maximum available throughput and the coefficient of the linear decrease of function f. Then, we simulate the Wi-Fi cell with different numbers of concurrent flows and compare the outputs of the flow model to the outputs of ns-3.

Each plot in Figure 4 shows the throughput estimations obtained after calibration. Different MCS configurations lead to different maximum theoretical throughput values, visible in our plots where Figure 4a (MCS2) is estimated to reach approximately 33.3Mbps against 86.5Mbps in Figure 4d (MCS5). The maximum relative error of the overall throughput is 1.5% under MCS2 in the case of a small number of concurrent flows. This error could be further reduced by using a more sophisticated f function.

This experiment shows the ability of our model to simulate the share of a Wi-Fi channel between an arbitrary number of concurrent flows under various MCS configurations. After calibrating the model, it is possible to obtain closely related outputs compared to ns-3.

6.1.2 SNR levels microbenchmark: This experiment simulates communications when STAs have different locations, increasing the probability of collisions and errors. Two STAs send data to the AP, where node 1 is fixed, and the distance between node 2 and the AP gradually increases. The two stations are always in reach to avoid the hidden node issue. The flow-based model is calibrated following the methodology of Section 5.2.3. The MCS of the stations is set to 3, channel bonding and RTS/CTS are deactivated.

Figure 5 shows the throughput of nodes 1 and 2, where the distance between the AP and node 2 ranges from 50m to 65m. Ns-3 results are sampled with 30 different seeds since its signal propagation and packet-loss models depend on probabilistic computations. Below 51.5 meters, the channel is fairly shared between the two stations in both ns-3 and the flow model. Above this limit, node 2 is too far from the AP to attach and communicate properly, and cannot send any more data. The bandwidth that was previously dedicated to node 2 is given to node 1, which can use the full capacity of the link.

These results show that even without complex PER computations, defining the datarates of STAs as described in Section 5.2.3 allows to simulate STAs with different locations. Both the flow model and ns-3 lead to the same bandwidth modification when node 2 is too far from the AP. We conducted similar experiments with RTS/CTS activated, but it is not included here as it leads to the same conclusions.

6.2 Use case: large scale infrastructure

This section evaluates the performance and accuracy of Wi-Fi communications in a more realistic use case. The infrastructure simulated in this section is a metropolitan public Wi-Fi network. Such networks are available to provide Internet access to users in vast areas, such as in the case of Google Wi-Fi [1], or networks within commercial centers [6]. They can be composed of hundred APs, used by clients to access the Internet. We simulate several Wi-Fi cells, connected to the outside world by a gateway through a wired core network, and a set of clients in the Wi-Fi cells. Network flows are created between client STAs and the gateway node of the core network.

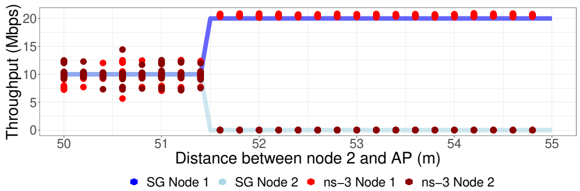


Figure 5: Impact of STA distance on the throughput share between stations.

We evaluate the accuracy of the flow-based model against ns-3, modifying the number of STAs per cell, the number of APs, the size of network flows, the number of messages sent by each STA, and the random seed of ns-3.

6.2.1 Overall throughput: In this scenario, each Wi-Fi cell is composed of 20 nodes, sending a message whose size is between 10MB and 30 MB to the gateway. All flows start at the same simulated timestamp of 10 seconds. This first ensures that all nodes have enough time to connect to their respective AP before the flows start. It also constitutes a more challenging situation for the model, as non-concurrent flows are easier to predict.

Figure 6 shows the duration between the beginning of communications and the time of the last received byte in both SimGrid and ns-3 with 3 Wi-Fi cells. We use 35 different seeds for ns-3, where we remove the runs where the flows are not finished by the end of the experiment (150 seconds), considered as outliers. This experiment shows that SimGrid's estimation of the time to send data from concurrent stations in the network is within the error range of ns-3 outputs that differ by the random seed.

When considering the overall network usage, the outputs of SimGrid are coherent with the ones of ns-3 despite the more abstract model. Repeating this experiment with different flow sizes (between 10 and 30MB per flow) and number of Wi-Fi cells (1 to 10 cells) gives similar outputs that are omitted for concision.

6.2.2 Single flows throughput: This experiment compares the duration of each communication flow between the SimGrid implementation of the model and ns-3. As in the previous section, flows are ascending, from the STAs towards the AP. The flow-level model being less fine-grained than ns-3's, it doesn't compute probabilistic phenomenons as ns-3 does. In Figure 7, flows do not overlap, a

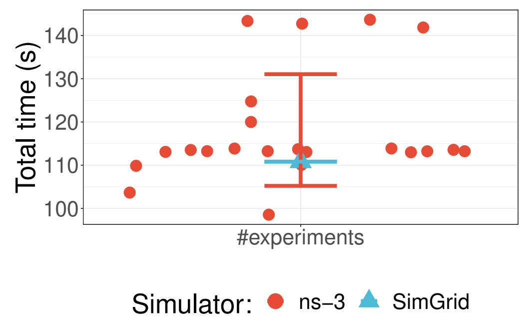
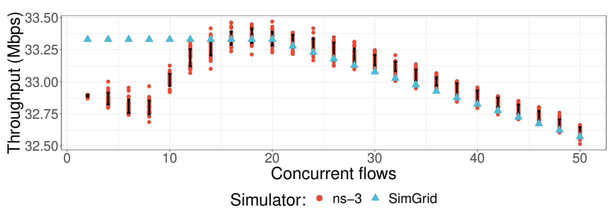
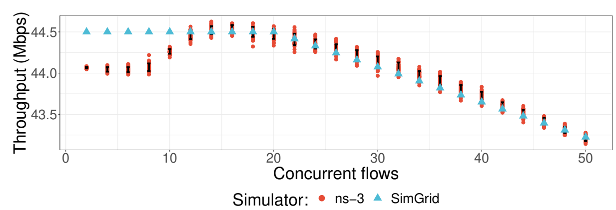


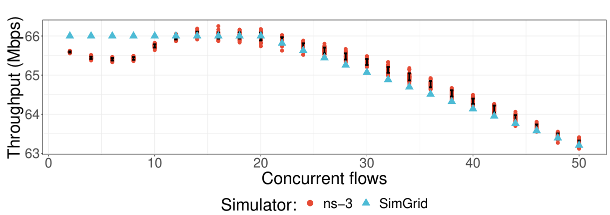
Figure 6: Time to communicate the same amount of data between SimGrid in blue and ns-3 in red on the same network topology. Times above 145s with ns-3 are considered as outliers and have been removed. Ns-3 points have different x-axis positions to observe the values for different seeds more easily.



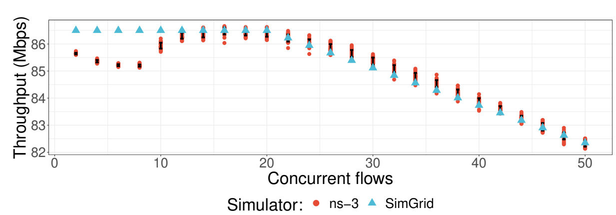
(a) MCS2 with $x_0 = 4239574$, $a = -3210$



(b) MCS3 with $x_0 = 5678270$, $a = -5424$



(c) MCS4 with $x_0 = 8517520$, $a = -12023$



(d) MCS5 with $x_0 = 1.120e + 07$, $a = -1.792e + 04$

Figure 4: Comparison of overall throughputs between ns-3 (in red) and SimGrid (in blue) simulations, depending on the number of concurrent flows in a single Wi-Fi cell using 35 ns-3 seeds. Black bars show the standard deviation around the mean for ns-3 simulations.

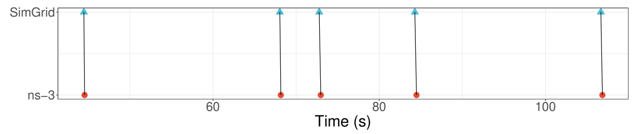
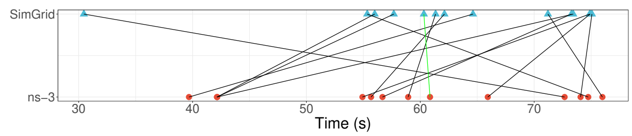
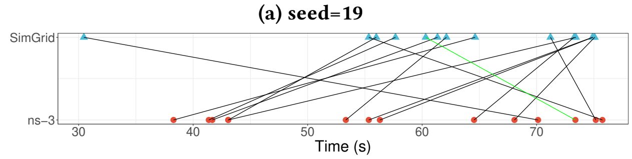


Figure 7: Flows end timestamps without concurrency (no overlap between flows start and end) in a cell of 5 STAs



(a) seed=19



(b) seed=20

Figure 8: Flows end timestamps with concurrency (all flows start at T=10s) in a cell of 14 STAs

situation that would happen in a quiet Wi-Fi cell with low traffic. In the case of Figure 8, all the flows start at the same time, T=10s.

Figures 7 and 8 present chronological plots indicating the flow termination timestamps in both SimGrid and ns-3. The top points indicate the end times for each flow in SimGrid, while the bottom points represent the timestamps of the same flows in ns-3. Each vertical line connects SimGrid and ns-3 for a given flow. When the outputs of SimGrid and ns-3 perfectly match for a given flow (as we intend to), the corresponding line is perfectly vertical. In the case of non-overlapping flows, the logarithmic error, as defined in [25], is on average equal to 0.051, meaning a 5% relative error between SimGrid and ns-3 flow durations. In the case of concurrent flows, however, the end timestamps do not match between ns-3 and SimGrid, leading to very important relative errors. In this case, the throughput sharing of ns-3 is very dependent on the random seed used for the simulation, as highlighted for seeds 19 and 20. For instance, the flow colored in green finishes just after 60 seconds in SimGrid. While ns-3 outputs a similar timestamp in Figure 8a, there is more than 10 seconds of delay in the case of Figure 8b (around 73 seconds). The error between the flow model and ns-3 is thus very different depending on the random seed of ns-3. Additionally, we can also observe that the end timestamps of the last flows are similar in SimGrid and ns-3.

What is obtained using SimGrid is one of the possible share of the channel between the concurrent flows (a theoretical fair share). Despite having very different results between ns-3 and SimGrid for single flow durations depending on the random seed, leading to an important logarithmic error, the flow-based model outputs a coherent time-share of frames in the network. As outlined previously, we estimate accurately the overall duration for all flows (shown in Figure 6) and give one possible share of the channel when observing flows separately.

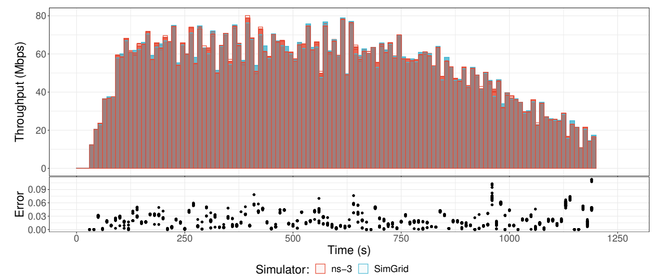


Figure 9: On top, data received by the gateway node by 10s intervals with 22 cells and 126 stations. In the bottom, relative error between ns-3 and SimGrid

6.2.3 Network throughput over time: This experiment explores the evolution of the throughput in the same infrastructure under a variable load. In this case, 22 Wi-Fi cells and 126 active STAs are simulated. Each station sends a number of messages centered around 40 with a deviation of 3, of variable size 1.5MB with a deviation of 1MB over 1250 seconds of simulated time. Wi-Fi cells do not overlap to avoid inter-cell interference. Ns-3 experiments are replicated using 35 different seeds.

Figure 9 shows the amount of data originating from the STAs and received by the gateway node by intervals of 10 seconds. We note th_{sg} (respectively th_{ns}) the throughput of SimGrid (respectively ns-3) during each interval. The amount of data going through the network changes over the simulated time because of the different packet sizes and time intervals between consecutive packets for each STA. The relative error regarding the data throughput (computed as $\frac{|th_{sg}-th_{ns}|}{th_{ns}}$ for each interval) remains below 12.5% at most, and below 10% on average.

This experiment shows that our model can be used to study large-scale platforms under dynamic network loads.

6.2.4 Performance comparison: This experiment explores the performance improvement of our flow-based model in relation to ns-3. To that extent, we simulate between 1 and 22 Wi-Fi cells. The number of STAs in each cell is centered around 7, with a deviation of 3, uniformly distributed around the AP. Flows are the same as in Section 6.2.3.

Figure 10 shows the time spent simulating our scenarios for infrastructures of different sizes. While the simulations using ns-3 require several hours for each seed, SimGrid based simulations take at most a few seconds for the longest simulations. This difference is explained by the tradeoff between the model granularity of Wi-Fi communications depending on flow and packet-based models. While the packet-based approach can be of linear complexity with regard to the number of simulated packets, the complexity of the flow-based approach only depends on the number of flows. We executed other simulations in SimGrid, made of up to 650 STA across 100 cells. The longest simulations took approximately 25 seconds to complete. Similar experiments using ns-3 would require more than 44 hours according to the linear regression visible in Figure 10. For large simulations, the peak memory usage was higher with SimGrid compared to ns-3 (approximately 300 MB against 150 for ns-3 to simulate 100 STAs) due to the large amount of

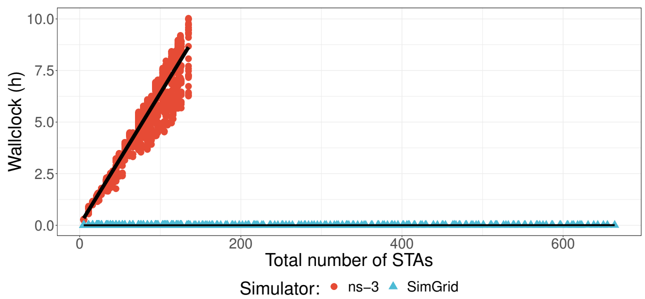


Figure 10: Wallclock comparison of simulation durations between ns-3 and SimGrid. The black line is the linear regression of time values.

simultaneous flows to handle in SimGrid. But this memory usage takes place over a much shorter amount of time in SimGrid than ns-3.

Overall, our experiments show that using a flow-based model instead of the classical packet-level one leads to a performance improvement of several orders of magnitude for simulating large Wi-Fi networks while providing estimations of similar accuracy.

7 CONCLUSION AND PERSPECTIVES

This paper presented a new Wi-Fi model based on flow-level simulation. This model consists of inequations describing the share of a Wi-Fi channel between STAs that need to send data, to which is added a throughput degradation function to account for the performance loss of concurrent communications. The main drawback of our approach is that it requires to be calibrated for the MCS configuration, the distance of STAs to the AP, and the number of concurrent flows. Yet, this calibration can be done through small-scale simulation experiments or using real measurements. Once properly calibrated, it can be used to fit different Wi-Fi configurations.

The presented experiments show that the predictions of our model match the ones of ns-3, one of the most widely used simulators in the network community. By exploring the use case of a metropolitan area network using Wi-Fi, we managed to obtain similar values compared to ns-3 (average relative error below 10%), while significantly decreasing the simulation runtime, from more than eight hours to a few seconds. This allows us to simulate large Wi-Fi infrastructures, which was hardly possible with previously developed models to our knowledge.

In the future, this model could be improved to simulate mobility and to obtain more detailed signal propagation predictions. This model could also be used to perform network simulations of large-scale infrastructures composed of heterogeneous communication technologies (i.e. mixing wired and wireless nodes) in order to better understand the advantages and drawbacks of modern distributed infrastructures, such as edge computing.

ACKNOWLEDGMENT

Experiments presented in this paper were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations (see https://www.grid5000.fr).

The authors want to thank Arnaud Legrand for his valuable help on designing the models presented in this paper.

REFERENCES

[1] M. Afanasyev, T. Chen, G. Voelker, and A. Snoeren. 2010. Usage patterns in an urban WiFi network. *IEEE/ACM Trans. on Netw.* 18, 5 (2010), 1359–1372.
[2] D. Balouek et al. 2013. Adding Virtualization Capabilities to the Grid'5000 Testbed. In *Cloud Computing and Services Science*. Vol. 367. 3–20.
[3] T. Barnett, S. Jain, U. Andra, and T. Khurana. 2018. Cisco visual networking index (VNI) complete forecast update, 2017–2022. *Cisco Knowledge Netw. Pres.* (2018).
[4] G. Bianchi. 1998. IEEE 802.11-saturation throughput analysis. *IEEE Communications Letters* 2, 12 (1998), 318–320.
[5] P. Bosch, J. Wyffels, B. Braem, and S. Latré. 2017. How is your event Wi-Fi doing? Performance measurements of large-scale and dense IEEE 802.11n/ac networks. In *IFIP/IEEE Symp. Integrated Network and Service Management*. 701–707.
[6] V. Brik, S. Rayanchu, S. Saha, S. Sen, V. Shrivastava, and S. Banerjee. 2008. A measurement study of a commercial-grade urban wifi mesh. In *ACM SIGCOMM Conf. on Internet Measurement*. 111–124.
[7] H. Casanova, A. Giersch, A. Legrand, M. Quinson, and F. Suter. 2014. Versatile, Scalable, and Accurate Simulation of Distributed Applications and Platforms. *J. Parallel and Distrib. Comput.* 74, 10 (2014), 2899–2917.
[8] X. Chang. 1999. Network simulations with OPNET. In *Winter Simul. Conf.* 307–314.
[9] T. Giuli and M. Baker. 2002. Narses: A scalable flow-based network simulator. *CoRR* cs.PF/0211024 (2002).
[10] L. Guegan, B. Amersho, A.-C. Orgerie, and M. Quinson. 2019. A Large-Scale Wired Network Energy Model for Flow-Level Simulations. In *Int. Conf. on Advanced Information Networking and Applications*. 1047–1058.
[11] T. Henderson, M. Lacage, G. Riley, C. Dowell, and J. Kopena. 2008. Network simulations with the ns-3 simulator. *SIGCOMM demonstration* 14 (2008).
[12] M. Lacage and T. Henderson. 2006. Yet Another Network Simulator. In *Workshop on Ns-2: The IP Network Simulator*.
[13] M. Lacage, M. Manshaei, and T. Turletti. 2004. IEEE 802.11 rate adaptation: a practical approach. In *MASCOTS*. 126–134.
[14] A. Legrand. 2015. *Scheduling for large scale distributed computing systems: approaches and performance evaluation issues.* HDR dissertation. Univ. Grenoble.
[15] Y. Liu, S. Crisp, and D. Blough. 2021. Performance Study of Statistical and Deterministic Channel Models for MmWave Wi-Fi Networks in Ns-3. In *Workshop on Ns-3*. 33–40.
[16] Sohaib Manzoor, Yachao Yin, Yayu Gao, Xiaojun Hei, and Wenqing Cheng. 2020. A Systematic Study of IEEE 802.11 DCF Network Optimization From Theory to Testbed. *IEEE Access* 8 (2020), 154114–154132.
[17] C. Miao, J. Wang, H. Wang, J. Zhang, W. Zhou, and S. Liu. 2018. A Multi-dimension Measurement Study of a Large Scale Campus WiFi Network. In *IEEE Conf. on Local Computer Networks*. 351–359.
[18] P. C. Ng, S. C. Liew, K. C. Sha, and W. T. To. 2005. Experimental study of hidden node problem in IEEE 802.11 wireless networks. In *SIGCOMM Poster*.
[19] R. Patidar, S. Roy, T. Henderson, and A. Chandramohan. 2017. Link-to-System Mapping for Ns-3 Wi-Fi OFDM Error Models. In *Workshop on Ns-3*. 31–38.
[20] H.-A. Safavi-Naeini, F. Nadeem, and S. Roy. 2016. Investigation and improvements to the OFDM Wi-Fi physical layer abstraction in ns-3. In *Workshop on ns-3*. 65–70.
[21] P. Serrano, A. Garcia-Saavedra, G. Bianchi, A. Banchs, and A. Azcorra. 2014. Per-frame energy consumption in 802.11 devices and its implication on modeling and design. *IEEE/ACM Transactions on networking* 23, 4 (2014), 1243–1256.
[22] SimGrid 2022. SimGrid. https://simgrid.org/. last accessed June 2022.
[23] M. Stoffers and G. Riley. 2012. Comparing the ns-3 propagation models. In *MASCOTS*. 61–67.
[24] A. Varga. 2010. OMNeT++. In *Modeling and tools for network simulation*. 35–59.
[25] P. Velho, L. Schnorr, H. Casanova, and A. Legrand. 2013. On the validity of flow-level TCP network models for grid and cloud simulations. *ACM Transactions on Modeling and Computer Simulation* 23, 4 (2013), 1–26.
[26] H. Yang, D.-J. Deng, and K.-C. Chen. 2018. On Energy Saving in IEEE 802.11ax. *IEEE Access* 6 (2018), 47546–47556.
[27] Zhenzhe Zhong et al. 2015. Issues and challenges in dense WiFi networks. In *Int. Wireless Communications and Mobile Computing Conf.* 947–951.