Modeling of IEEE 802.11 Multi-hop Wireless Chains with Hidden Nodes

Thiago Abreu
Université Lyon 1 - LIP
France
thiago.wanderley@ens-lyon.fr

Bruno Baynat
Université Pierre et Marie Curie - LIP6
France
bruno.baynat@lip6.fr

Thomas Begin
Université Lyon 1 - LIP
France
thomas.begin@ens-lyon.fr

Isabelle Guérin-Lassous
Université Lyon 1 - LIP
France
isabelle.guerin-lassous@ens-lyon.fr

Nghi Nguyen
Université Lyon 1 - LIP
France
huu.nguyen@ens-lyon.fr

ABSTRACT

Despite the abundant literature devoted to the performance analysis of the IEEE 802.11 protocol, only a handful of works have tackled the case of multi-hop flows along a chain with relay nodes. In this paper, we follow up an existing modeling framework to analytically evaluate the performance of multi-hop flows along a wireless chain of four nodes. Unlike existing approaches, the proposed model accounts for a non-perfect physical layer, handles the hidden node problem, and is applicable under workload conditions ranging from flow(s) with low intensity to flow(s) causing the network to saturate. Its solution is easily and quickly obtained and delivers estimates for the expected throughput and for the datagram loss probability of the chain. The accuracy of the model is generally good and, importantly, it captures the performance collapse that takes place in certain wireless multi-hop networks.

1. INTRODUCTION

Most WLANs are based on the IEEE 802.11 standard, which implements a probabilistic media access control (MAC) layer. The IEEE 802.11 standard is generally appreciated for its ease of implementation and simple configuration. However, the performance evaluation of these WLANs are generally not straightforward because of their non-deterministic nature (due to the use of a probabilistic contention algorithm and to the dynamic behavior of the radio medium).

In the case of infrastructure mode, where nodes communicate through an access point, a large body of analytical models have been proposed in the literature. These models afford a quick means for researchers and practitioners to forecast many aspects of a WLAN behavior before its deployment, or to better set it up.

Multi-hop wireless networks are another type of WLANs where each node participates in routing by forwarding packets for other nodes. Typically, their decentralized nature makes them suitable for cases where there are no central nodes or for emergency situations like natural disasters. However, multi-hop wireless networks raise new issues with regards to the routing protocols and to the discovery and refinement of their performance.

This paper addresses the performance evaluation of a multi-hop wireless network based on IEEE 802.11, where packets need to hop several relay nodes before reaching their final destination. We refer to these networks as chains. In a previous paper [1], we analyzed the behavior of the simplest chain which has only one relay node. Though a necessary milestone, its limited size allowed us to overlook the well-known but complex issue of hidden node problem.

The contributions of this paper are twofold. First, we extend our modeling framework to evaluate the performance for a flow conveyed through larger chains in which the hidden node problem takes place. The solution to the model is based on a simple iterative scheme that is solved typically within less than a second. In general, the proposed model delivers good forecasts for the expected throughput and for the datagram losses as a function of the actual positions of the relay nodes and for various values of the flow rate. Second, our model affords a convenient means to quickly investigate the performance behavior of a chain under various conditions. Given the fast solution of our model, we explore many possible configurations for the relay nodes and for the levels of the flow rate in order to get a better understanding of multi-hop wireless networks and to highlight properties inherent to those networks.

The remainder of the paper is organized as follows. In Section 2 we present in detail the considered scenario. Section 3 reviews the relevant literature to this work. The modeling framework and its solution for the hidden node problem are described in Section 4. In Section 5 we report numerical results in order to assess the accuracy of our model as well as to point out performance properties existing in wireless multi-hop networks. Section 6 concludes the paper.

2. SCENARIO DESCRIPTION

The scenario under consideration is depicted in Figure 1. It consists of a wireless multi-hop chain with 4 nodes, each equipped with a single IEEE 802.11 communication interface. Every node can communicate only with its 1-hop neighbors, but its carrier sensing range covers its 2-hop neighbor nodes. Note that there are no restrictions to the nodes alignment and position, as long as they meet the aforementioned assumptions for the communication and for the carrier sensing ranges.

The physical layer used for the frame transmission is unreliable (non-perfect), and therefore, frames may be lost because of bits error or alteration. This is taken into account by the Bit Error Rate (BER), which gives the probability that a bit is misinterpreted at a receiver node due to the transmission process (which includes noise, distortion, attenuation, etc). In our study, the BER is affected by the propagation process and noise. In addition to BER, frames may also be lost when nearby nodes are transmitting simultaneously, which causes frame collisions. In a four-nodes chain, collisions are frequent since nodes 1 and 4 are exposed to the hidden node problem.

The four-nodes chain conveys packets (datagrams) from node 1 up to node 4 (see Figure 1). All datagrams are of same length and the datagrams generation at node 1 follows a Poisson process with a rate λ_1. This flow of datagrams constitutes the workload for the chain.

The nodes communicate using the DCF mode of IEEE 802.11. For the sake of saving space, we do not detail the IEEE 802.11 DCF principles. The interested reader may refer to [10] for a detailed description, or to [1] for a summary of the DCF mode. In our scenario, we disable EIFS by only using DIFS. Actually, several studies (e.g., [5]) have suggested that the EIFS is not triggered in many IEEE 802.11 cards. Besides, some chipsets (e.g., Atheros AR9331) have also the possibility to ignore EIFS. We also disable the RTS/CTS mechanism since RTS/CTS has been shown to be inefficient in the case of a chain [17]. Otherwise, all the other mechanisms of the DCF mode (DIFS, backoff, backoff freezing, SIFS, acknowledgment, Binary Exponential Backoff and retransmission limit) are considered in our scenario.

Though it may not be clear at first glance, the performance evaluation of a four-nodes chain is far from obvious. Several issues hinder its analysis. First, as discussed above, a four-node chain is exposed to the hidden node problem. Nodes 1 and 4 are out of each other's carrier sensing range and thus, frames from node 1 often collide with the acknowledgments returned by node 4 (to node 3). Depending on the specific locations of the relay nodes and on the rate of the workload, the collision probability for frames issued by node 1 can peak to 30%. In this case, the performance of the chain are markedly diminished since several frame transmissions are likely to be needed to hop a datagram over that link. Second, every wireless link experiences a different BER since in general the distance between nodes differs, which in turn leads to various channel qualities. As a result, the number of frames required before getting a successful transmission differs for each node. Third, the nodes of a chain exhibit a strong interdependence. The periods of backoff freezing

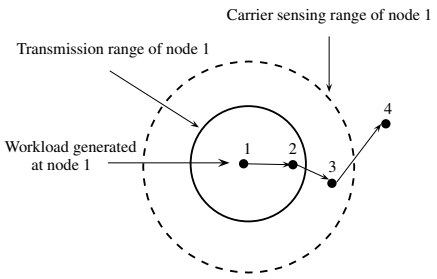

Figure 1: Multi-hop chain with 4 nodes

of a node correspond precisely to the transmission periods of its neighbors. Also, the workload is unevenly scattered among the nodes. For instance, for some nodes configurations, node 1 can be exposed to a high level of workload while the following nodes may experience a relatively low or even a quasi-null (starvation) degree of workload. These inner dependencies are generally complex to emulate in the analytical models. To handle them, we resort on an iterative scheme that discovers the proper values for each of the node parameters.

3. RELATED WORK

There is a large body of literature devoted to the performance modeling of the IEEE 802.11 protocol. Many of the proposed models address different problems, though. The differentiating factors include the specific topology of interest (cells or multi-hop networks), the assumptions on the generated traffic (while some studies restrict their analysis to the sole case of saturated traffic sources at nodes, others cover the more general case of nodes in non-saturated regime), the length of flows (single-hop flows versus multi-hop flows), the properties of the physical layer (e.g., perfect or non-perfect with BER), the length (if any) of the buffer at nodes (to enqueue extra datagrams before an overflow occurs), as well as some precise DCF mechanisms (e.g., retransmission limit).

In the case of a cell (i.e., a single-hop network in which each node senses the transmission of each other), Bianchi [4] developed a seminal model based on a Discrete-Time Markov Chain (DTMC) to evaluate the attainable throughput of the cell. In this work, the author assumes that every node runs in a saturated regime, and that frame transmissions are not exposed to a BER, neither to a retransmission limit. Several follow-up attempts have been carried out to better match some IEEE 802.11 mechanisms, e.g., [14], or to fit to more realistic physical conditions [15]. However, these latter studies are devoted to a single cell scenario, thereby bypassing the hidden node problem (since each node senses all other transmissions).

For networks with multi-hop scenarios, where nodes are not necessarily in each other carrier sensing range, existing works typically fall into two categories depending on the length of flows, i.e., single-hop versus multi-hop flows. Note that in both cases, nodes can be exposed to the hidden node problem, which clearly hinders the study of their performance. In the case of single-hop flows, in which the source and the

destination are only one hop away, Wang and Kar [16] developed an analytical Markovian model to evaluate the average throughput of flows. To help their analysis, the authors assume no binary exponential backoff, a perfect physical layer (no BER) and a RTS/CTS mechanism, which always performs successfully for the channel reservation. Moreover, they consider all nodes to be identical in terms of workload and of neigborhood activity. In another set of works, the authors of [7, 8, 12] have proposed alternate analytical models to derive the throughput of single-hop flows. They come up with a closed-form expression for the throughput in [12], but the used approach does not seem to be applicable for other performance parameters of interests, e.g., the packet loss probability. In a prior work [7, 8], the authors use an iterative approach to estimate parameters like flow throughput and packet loss probability. In these two latter works, the proposed analysis is eased by the fact that all source nodes are statistically identical and thus exposed to the same hidden node condition. This assumption is obviously not applicable for our purpose.

When it comes to the case of multi-hop flows, i.e., flows where source and destination can not directly communicate, and are possibly exposed to the hidden node problem, there is only a handful of relevant works. Among them, we discuss a couple of works, which have tackled through an analytical standpoint the performance evaluation issue for a flow conveyed through a chain of nodes. In [3], the authors highlighted, with theoretical and numerical results, some of the complex phenomena undergone by a wireless chain (e.g., mutual exclusion of links, unfairness between the nodes). In particular, they studied the length evolution of the queues at the buffer nodes, and they emphasized their correlation (some queue may be empty, while others are not). The authors also proposed an analytical model in [3], which applies only when every node (including relay nodes of the chain) is in a saturated regime (i.e., it always has a datagram waiting to be transmitted). In a separate work [9], the authors have proposed to estimate the maximum attainable throughput of the chain by considering that its value approximately corresponds to the actual capacity of the bottleneck link (i.e., the link that takes the longest time to transmit a datagram). However, this assumption overlooks the fact that the backoff periods of neighbor nodes are often overlapped, which in turn clearly affects the actual value of the throughput. Finally, in a previous work [1], we proposed an analytical model that takes into account the buffer length at each node. The model handles the fact that for a flow along a chain, intermediate nodes are often in a starvation state (i.e., empty buffers) while the flow may saturate the first node of the chain. However, this latter model was restricted to the case of a three-nodes scenario, in which the hidden node problem does not exist, thereby preventing virtually all frame collisions.

To summarize, there is only a handful of works specifically devoted to the analytical performance evaluation of IEEE 802.11 in the case of multi-hop networks with multi-hop flows. Besides, it seems that none of them handles at the same time realistic assumptions regarding the behavior of the MAC protocol, the inter-dependencies in the distribution of the workload among the nodes (some nodes may be in saturation while others may be in starvation) and the hidden node problem, which are fundamental properties of a wireless chain with several nodes. Hence, we propose a new analytical model that deals with all these issues.

4. MODELING FRAMEWORK

In [1] we introduce the first parts of a general framework for performance evaluation of a multi-hop wireless chain. The analysis was restricted to a simple scenario with only three nodes. Here, we extend markedly our proposed framework by introducing one important missing feature in the previous analysis, i.e., the hidden node problem which takes place in larger chains. Unlike our previous study of [1] in which virtually no frames collisions can occur, larger chains are prone to frequent frames collisions. In this paper, we present a practical means to handle these collisions within our modeling framework. By doing so, we also demonstrate that our preliminary work is extendable to more general scenarios, with the ultimate objective being to derive a whole framework for the performance evaluation of any multi-hop wireless chain.

Following the framework developed in [1], our model is composed of two levels: a global queueing network model, and several local Markov chain models. The queueing network matches the chain topology (one queue being associated with each transmitting node) and mostly aims to capture the workload level and the buffer overflows at intermediate nodes. The Markov chains precisely describe the transmission delay of a datagram according to IEEE 802.11 DCF protocol. If the input parameters of the global model are supposed to be known, its output performance parameters allow to parameterize the local Markov chain models. Conversely, once the local Markov chain models are parameterized, they provide an estimation of the missing input parameters for the global model. As a result, our overall model can be solved using a fixed-point iterative procedure.

The remainder of this section is as follows. First, we present our hierarchical modeling framework when applied to a chain with 4 nodes following the principles of [1]. Then, we turn to the difficulties introduced by larger chains, and in particular, to the derivation of the frame collision probability (see Section 4.3). Finally, we end up this section by presenting the iterative algorithm used to solve our model.

4.1 Global model

Since only three nodes are effectively transmitting frames (node 4 only returns acknowledgments), the global queueing model associated with the chain is composed of three queues with finite buffer as illustrated in Figure 2. The customers of this queueing model are the datagrams of the chain and the buffer size of queue i is denoted by K_i. The service rate μ_i of queue i is, by definition, the inverse of the mean service time S_i of queue i, which corresponds to the average time node i needs to transmit a datagram that is ready to be sent over the radio channel. As developed in [1], S_i includes all successive frame (re)transmissions (corresponding to the considered datagram), as well as all IEEE 802.11 DCF protocol delays (DIFS, backoff, SIFS, timeout) and all freezing times due to another node transmission during the backoff of node i. This parameter will be estimated thanks to the local Markovian model associated with node i (in Section 4.2).

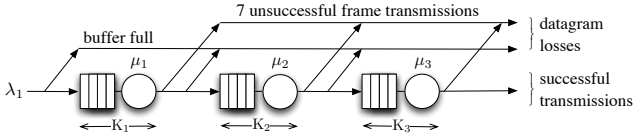

Figure 2: Global queueing model

Like in our previous model [1], a datagram can be lost either because of a buffer overflow or because of excessive retransmissions of the associated frames. But there is a fundamental difference between the global model presented here and the one previously developed in [1]: a frame loss can either be due to low quality of the channel or to collision over the shared medium. The frame loss probability p_{f_i} of node i must account for these two possibilities, whereas in [1] it was only related to the BER. If we denote by p_{BER_i} the probability that a given frame sent by node i is lost because of the BER, and by p_{coll_i} the probability that the frame sent by node i is lost because of a collision with another frame, and if we assume that these two events are independent but not disjoint (a frame can be both in error and in collision), the frame loss probability p_{f_i} of node i can be obtained as:

$$p_{f_i} = p_{coll_i} \cup p_{BER_i} = p_{coll_i} + p_{BER_i} - p_{coll_i}\, p_{BER_i} \quad (1)$$

As for the BER probability, which was solely used in [1] for estimating the frame error probability, its derivation remains identical. However, the collision probability, which was virtually null in the case handled by [1], can not be neglected and, as will be seen later, is a very sensitive parameter that must be carefully estimated. This will be detailed in Section 4.3.

Provided that all the parameters of the global model are estimated (i.e., the services rates μ_i and the frame loss probability p_{f_i}), the queueing model can be solved using the approach described in [1]. It consists in decomposing the queueing network into independent single *M/M/1/K* queues, as illustrated in Figure 3. The underlying approximations have already been discussed and validated in our previous work. Basically, it shows that considering exponential service times and poissonian arrivals at each queue results in a negligible additional error, but it drastically simplifies the solution to our model and helps its extension to more general scenarios.

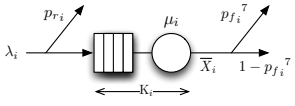

Figure 3: Corresponding decoupled queueing model

We derive all the performance parameters of our interest based on the well-known closed-form formulas of the *M/M/1/K* queue. The output throughput of queue i is:

$$\overline{X}_i = \mu_i(1 - \pi_i(0)) \quad (2)$$

where $\pi_i(n)$ is the stationary probability of having n customers in the i-th *M/M/1/K* isolated queue. From relation (2), we can derive the level of workload at node $i+1$ (i.e., average number of datagrams that are correctly transmitted by unit of time from node i to node $i+1$) as:

$$\lambda_{i+1} = \overline{X}_i(1 - {p_{f_i}}^7) \quad (3)$$

The utilization of node i is:

$$\overline{U}_i = 1 - \pi_i(0) \quad (4)$$

The buffer overflow probability for node i, which corresponds to the probability that a datagram finds the buffer of node i full upon its arrival, is simply obtained using the PASTA theorem:

$$p_{r_i} = \pi_i(K_i) \quad (5)$$

As for the overall performance of the chain which will be used to validate our model in Section 5, we first consider the chain output throughput, $\overline{X}_{\text{chain}}$, corresponding to the average number of datagrams by units of time that reach the final destination:

$$\overline{X}_{\text{chain}} = \overline{X}_3(1 - {p_{f_3}}^7) \quad (6)$$

We also consider the overall loss probability, $p_{r_{\text{chain}}}$, defined as the probability that a datagram is lost due either to buffer overflow or to seven unsuccessful consecutive frame transmissions at any node of the chain (see Figure 2), and that can be derived as follows:

$$p_{r_{\text{chain}}} = 1 - \prod_{i=1}^{3}(1 - p_{r_i})(1 - {p_{f_i}}^7) \quad (7)$$

Note that $p_{r_{\text{chain}}}$ can also be considered as the proportion of lost datagrams and is thus related to $\overline{X}_{\text{chain}}$ by:

$$p_{r_{\text{chain}}} = \frac{\lambda_1 - \overline{X}_{\text{chain}}}{\lambda_1} \quad (8)$$

4.2 Local models

The missing parameters of the global queueing model presented in the previous section are the frame collision probability p_{coll_i} and the service rate μ_i. This section focus on the derivation of the later one. Following the methodology presented in [1], we associate with each queue i of the global model a Continuous-Time Markov Chain (CTMC) that precisely describes the transmission process of node i according to IEEE 802.11 DCF protocol. The objective of this CTMC is to provide an estimation of the mean service time S_i, inverse of the service rate μ_i of queue i. This CTMC is depicted in Figure 4 and its structure is similar to the one described in [1]. However, some of its parameters have to be adjusted to account for frame collisions.

Let us quickly remind that this CTMC is globally made of seven "blocks", each one corresponding to a given stage k of the backoff and modeling the backoff time preceding the k-th transmission of a given datagram. Any transmission attempt starts with a DIFS time and then enters a backoff procedure, that can be interrupted anytime when the node senses a busy canal, and ends with a transmission time, that can be either successful leading to the "end of service" state, or in error bringing the process to the next stage of the backoff. Rates α_i and γ_i that appear in this CTMC are simply the inverse of a time-slot duration for the first one and the inverse of the freezing duration for the second one. Both can be derived from IEEE 802.11 specifications. The inverse

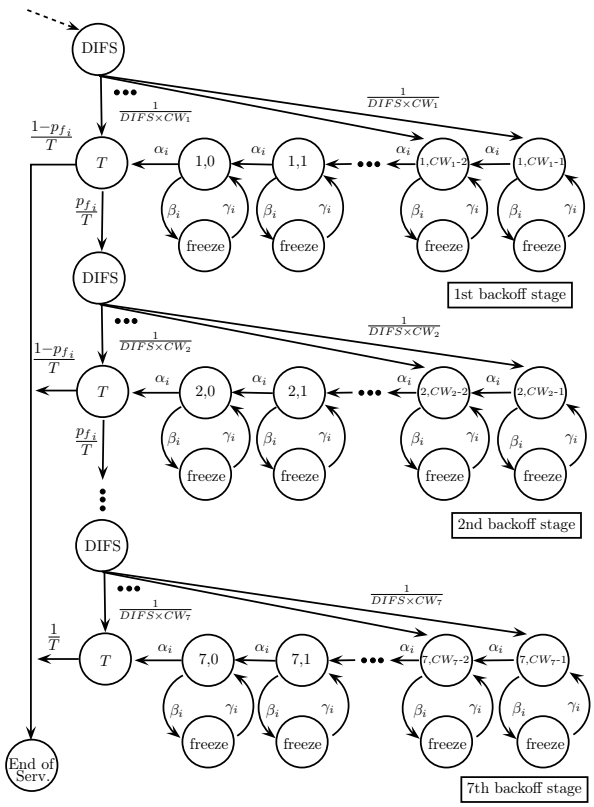

Figure 4: Local Markov chain model

of β_i corresponds to the average time separating two backoff freezing, provided that the node is in backoff. More details concerning the description of this CTMC can be found in [1].

Assuming the values for every parameter of the CTMC are known, it is then straightforward to compute the mean transmission time for a frame S_i, i.e., the time needed to go from the first "DIFS" state up to the "end of service" state, and therefore getting the service rate μ_i of each $M/M/1/K$ queue i. We remind the computation of S_i:

$$S_i = \frac{1}{\mu_i} = t_{1,i} + p_{f_i} \times (t_{2,i} + p_{f_i} \times (t_{3,i} + ... + p_{f_i} \times t_{7,i})) \quad (9)$$

where $t_{k,i}$ corresponds to the average time spent by the process in "line" k of the CTMC:

$$t_{k,i} = \text{DIFS} + \frac{CW_k}{2} \times r_i + T \quad (10)$$

and r_i is the average time spent in any pair of loop states $(\{k,j\},\{freeze\})$ at node i:

$$r_i = \frac{1}{\alpha_i} \times \left(1 + \frac{\beta_i}{\gamma_i}\right) \quad (11)$$

Basically, every parameter of this CTMC, except β_i and p_{f_i}, can easily be parameterized (following [1]). We now detail the derivation of parameter β_i while the computation of the frame loss probability p_{f_i} is described in Section 4.3. As stated before, the inverse of β_i corresponds to the mean time between two successive backoff freezing (provided node i is in backoff). This quantity is directly related to the mean backoff duration for a frame of node i, denoted as $\overline{B_i}$, and the average number of freezing of the backoff of node i, denoted as $\overline{npi}$, by the following relation (see [1] for derivation):

$$\frac{1}{\beta_i} = \frac{\overline{B_i}}{\overline{npi}} \quad (12)$$

As shown in [1], the mean backoff duration for a frame of node i can be expressed as:

$$\overline{B_i} = \frac{\frac{CW_1}{2} f_{1,i} + \frac{(CW_1+CW_2)}{2} f_{2,i} + \cdots + \frac{(CW_1+CW_2+\cdots+CW_7)}{2} f_{7,i}}{\overline{f_i}} T_S \quad (13)$$

where T_S is the duration of one time-slot, CW_k is the contention window size in the k-th backoff stage:

$$CW_k = \min(2^{4+k} - 1, 1023) \quad (14)$$

and $f_{k,i}$ is the probability that the transmission of a datagram at node i requires exactly k frames. This probability can be derived from the frame error probability as:

$$f_{k,i} = {p_{f_i}}^{k-1}(1 - p_{f_i}) \text{ for } k \leq 6 \text{ and } f_{7,i} = {p_{f_i}}^6 \quad (15)$$

Finally, $\overline{f_i}$ is the average number of frame (re)transmissions per datagram at node i:

$$\overline{f_i} = \sum_{k=1}^{7} k.\ f_{k,i} \quad (16)$$

Let us now turn our attention to the estimation of the average number of freezing of the backoff of node i, $\overline{npi}$, also referred to as the average number of pauses of the backoff. Remind that the backoff of a given node i is paused whenever a node j in the carrier sensing range of node i makes a transmission. For sake of simplicity we begin with the study of the backoff of node 1 assuming node 1 has always datagrams to transmit. As illustrated by Figure 5 for node 1, two consecutive frame transmissions are always spaced by a backoff which may be paused by potential transmissions of nodes 2 and 3. The average number of pauses for each backoff of node 1 is thus equal to the average number of frame transmitted by either node 2, denoted as $\overline{F_2}$, or by node 3, denoted as $\overline{F_3}$ (as transmissions of nodes 2 and 3 are exclusive) during the time between two consecutive frame transmissions of node 1. This formula can be extended to any node i of the chain, and it gives the following relation:

$$\overline{np_i} = \frac{\sum_{j \neq i} \overline{F_j}}{\overline{F_i}} \quad (17)$$

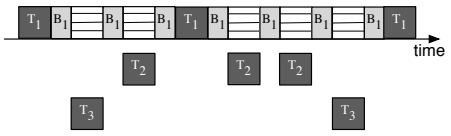

Figure 5: Backoff freezings of node 1 versus transmissions of its neighbors for the saturated case.

The frame throughputs involved in relation (17) can directly be obtained given the datagram throughputs of the global queueing model (relation (2)) and the average number of frame transmissions per datagram (relation (16)) as:

$$\overline{F_i} = \overline{X_i}.\ \overline{f_i} \quad (18)$$

In the more general case where node i is not saturated, the

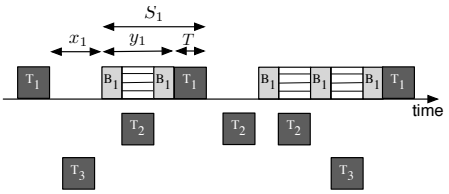

Figure 6: Backoff freezings of node 1 versus transmissions of its neighbors for the non-saturated case.

expression of $\overline{np}_i$ must be adjusted with a corrective factor δ_i as expressed in relation (19). This case is illustrated in Figure 6 for node 1.

$$\overline{np}_i = \frac{\sum_{j \neq i} \overline{F_j}}{\overline{F_i}} \delta_i \tag{19}$$

Indeed, when node i is not saturated, between two of its consecutive transmissions, there may be an idle time (x_i in the figure) before the beginning of its backoff. During this idle time, transmissions of any other node will not cause a freezing of its backoff, whereas during the rest of the time up to the transmission (y_i in the figure), transmissions of other nodes (in the sensing range of node i) will freeze its backoff. The corrective factor is thus simply given by the following ratio:

$$\delta_i = \frac{y_i}{x_i + y_i} \tag{20}$$

In order to estimate the corrective factor, it is worthwhile noting that the average service time of queue i, S_i, is equal to the time y_i plus a time T (that corresponds itself to the frame transmission time plus the transmission time of the corresponding ACK). Now, the missing value x_i is clearly related to node i utilization $\overline{U}_i$ and can be expressed with notations of Figure 6:

$$\overline{U}_i = \frac{S_i}{S_i + x_i} \tag{21}$$

As a result, the expression of the corrective factor becomes:

$$\delta_i = \frac{S_i - T}{S_i \frac{1-\overline{U}_i}{\overline{U}_i} + S_i - T} \tag{22}$$

Finally, from relations (12)-(16) and (18)-(22), we obtain the following expression for the missing parameters β_i:

$$\beta_i = \frac{\sum_{j \neq i} \overline{X}_j \ \overline{f}_j}{\overline{X}_i \ \overline{f}_i} \frac{\delta_i}{\overline{B}_i} \tag{23}$$

4.3 Frame collision probability

This section presents the derivation of the frame collision probability, which is a very important and sensitive parameter that is part of the frame loss probability (relation (1)) and that is both needed in the global queueing model and in local Markov chain models. A frame collision may be the result of two different factors. First, a collision can result from the well known hidden problem of two nodes that are not in the carrier sensing range of each other. Second, because of the sensing mechanism of 802.11, a collision can also occur when two neighboring nodes finish their backoff countdown

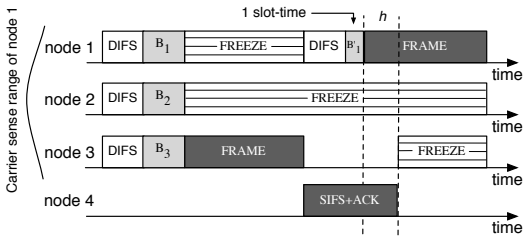

Figure 7: Collision between ACK from node 4 and frame from node 1

simultaneously. By assuming that these two possibilities result in disjoint events (which turns out to be exact in our scenario), we can decompose the frame collision probability of node i as the sum of the probability of both events:

$$p_{coll_i} = p_{hid_i} \cup p_{st_i} = p_{hid_i} + p_{st_i} \tag{24}$$

Let us first consider the hidden problem case and see how we can estimate the collision probability at node i due to frame collision with nodes that are hidden from node i, denoted as p_{hid_i}. In our scenario, since we assume a 2-hop carrier sensing range, the hidden problem can only take place between node 1 and node 4, and more precisely, between a data frame sent by node 1 to node 2 and an acknowledgement ("ACK") sent back by node 4 to node 3. As an illustration, in Figure 7, node 3 senses the medium idle for the duration of its backoff and then starts the transmission of a frame, freezing the backoff countdowns of nodes 1 and 2. The associated acknowledgement ("ACK") sent by node 4 does not prevent node 1 from resuming its backoff countdown (event "B_1'"), since nodes 1 and 4 are hidden. If the remaining backoff of node 1 is short enough (it corresponds to 1 time-slot in the example), node 1 will transmit its frame and a collision will occur for both frame and ACK. Although the ACK from node 4 collides, we consider that a collision happens at node 3, since the retransmission mechanism will be performed by this node.

As can be seen on the figure, the duration of the collision is bounded by the maximum overlap h between the frame transmission of node 1 and the ACK transmission of node 4:

$$h = \text{SIFS} + \text{ACK} - \text{DIFS} - 1 \text{ time slot} \tag{25}$$

We subtract 1 time slot to it, since after a backoff freezing period, the remaining backoff has at least 1 time slot to decrement.

By considering that the nodes have always a frame to transmit, the collision probability p_{hid_i} of node i due to a hidden node j can be estimated in a first approximation as the ratio between the duration of a possible collision (h) and the time during which a collision may take place ($h + \overline{B}_j$) in between two transmissions of node i:

$$p_{hid_i} = \frac{h}{h + \overline{B_j}} \tag{26}$$

Relation (26) has however two limitations. First, by only considering the average duration of the backoff ($\overline{B_j}$), we do not take into account the high variability of the binary exponential backoff used in IEEE 802.11. For instance, if node

1 is in the first stage of its backoff, an ACK from node 4 will very likely collide with a frame of node 1. Conversely, when node 1 is in one of the last stages of its backoff, an ACK from node 4 will have a high chance to be transmitted successfully. Let us define $tb_j(k)$, the proportion of time during which hidden node j remains in backoff stage k. $tb_j(k)$ is the ratio between the average time effectively spent in the k-th backoff stage of node j ($p_{f_j}^{k-1}\ t_{k,j}$) and the average service time of node j (S_j):

$$tb_j(k) = \frac{p_{f_j}^{k-1}\ t_{k,j}}{S_j} \tag{27}$$

The collision probability p_{hid_i} can thus be rewritten as:

$$p_{hid_i} = \sum_{k=1}^{7} tb_j(k)\ \frac{h}{h + \overline{B}(k)} \tag{28}$$

where $\overline{B}(k) = \frac{CW_k}{2}\ T_S$ is the average backoff duration at stage k.

Second, relation (26) (or equivalently relation (28)) implicitly assumes that node j has a datagram to transit (otherwise no collision can occur with node i) and should actually be denoted as the conditional probability $p_{hid_i|\text{node } j \text{ is not idle}}$. From the law of Total Probabilities, we can obtain the unconditioned collision probability, by noting that the probability $p_{hid_i|\text{node } j \text{ is idle}}$ is null and by reminding that the probability that node j is not idle is nothing but node j utilization:

$$p_{hid_i} = p_{hid_i|\text{node } j \text{ is not idle}}\ \overline{U}_j \tag{29}$$

By combining previous relations, the collision probability due to hidden nodes can finally be expressed as:

$$p_{hid_i} = \sum_{k=1}^{7} tb_j(k)\ \frac{h}{h + \overline{B}(k)}\ \overline{U}_j \tag{30}$$

Let us now consider the possible simultaneous transmissions of two neighboring nodes. As explained above, two nodes in the carrier sensing range of each other are very likely to synchronize themselves (mainly when the load is high). And there is a non negligible probability that the backoff countdowns of these two nodes expire simultaneously and that the two nodes start their transmission exactly at the same time, resulting in frame collisions. Let p_{st_i} denote the probability that a frame of node i collides with a frame of any node that is in its carrier sensing range and that starts a transmission at the same time as node i. This probability can be estimated as follows:

$$p_{st_i} = 1 - \prod_{j \neq i} (1 - \tau_j \overline{U}_j) \tag{31}$$

where τ_j is the probability that a given node j in the carrier sensing range of node i starts its transmission at the same time as node i, provided node j has something to transmit, and $\overline{U}_j$ is the utilization of node j. This is an approximation, assuming that three (or more) nodes have a very small chance to start their transmission all together. Now we simply estimate the missing conditional probability τ_j as the inverse of the average backoff duration of node j expressed in number of time slots:

$$\tau_j = \frac{1}{\overline{B}_j} \tag{32}$$

4.4 Fixed-point solution

As stated in the introduction of this section, the global queueing model needs as inputs what turns out to be output performance parameters of the local Markov chains models (μ_i and p_{f_i}). Conversely, the local Markov chain models require inputs that can be derived from the performance parameters of the global queueing model ($\overline{X}_i$, $\overline{U}_i$ and p_{r_i}). As a result, our overall model can be solved using the following fixed-point iterative procedure. The main loop of this algorithm is repeated until a convergence criterion is reached. We have used as a classical criterion, the maximum relative error obtained for the values of parameters μ_i between two successive iterations. When the algorithm has converged, it returns the chain performance parameters of interest ($\overline{X}_{\text{chain}}$ and $p_{r\,\text{chain}}$).

Algorithm 1 Fixed-point solution for the overall model

```
initialize μ_i and p_{f_i} for all i with non-absurd values
repeat
    solve the global queueing model and obtain for all i
      X̄_i, Ū_i and p_{r_i} (rel. (2)-(5))
    for each node i solve the local CTMC model and
      compute f̄_i, B̄_i and δ_i (rel. (16), (13) and (20))
      compute β_i (rel. (23))
      compute p_{hid_i} and p_{st_i} (rel. (30) and (31))
      update p_{f_i} (rel. (1))
      update μ_i (relation (9))
until convergence of the algorithm
return X̄_chain and p_{r chain} (rel. (6) and (7))
```

5. NUMERICAL RESULTS

In this section we address the accuracy concern of our proposed model, and we show how its exploitation can bring new insights in the behavior of wireless multi-hop chains at a very low cost of computation. Throughout this section, we use the standard parameter values of IEEE 802.11b as reported in Table 1. To ease the readiness of this section, the links have a constant capacity (physical rate) of 11 Mb/s (i.e., no Rate Adaptation) corresponding to the modulation CCK 11. The communication and carrier sensing ranges cover 399 and 700 meters, respectively. The received signal power at each node is computed using a transmission power of 31.6 mW, an antenna gain of 1 dBi and the two-ray ground reflection model. We derive the BER, which accounts for the non-perfect physical layer, based on a relation between the received signal power and the used modulation. In a nutshell, the BER rises from $4e^{-9}$ to $8e^{-5}$ as the distance between two communicating nodes widens from 150 to 399 meters. For more details, the reader can refer to the Appendix or to [6].

DIFS	50μs
SIFS	10μs
Time slot	20μs
Contention window size (min,max)	31, 1023
Frame retransmission limit	7

Table 1: IEEE 802.11b parameters

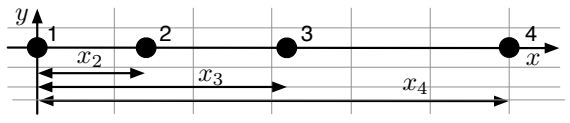

Figure 8: Topology used for the numerical results

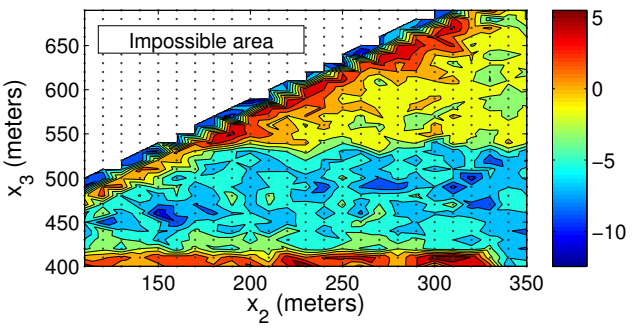

Figure 9: Percentage relative errors for the expected throughput for various positions of relay nodes (x_2 and x_3) with K_i=20 and λ_1=2 Mb/s

Average	$< \pm 5\%$	$\pm$5-10%	$\pm$10-15%	$> \pm 15\%$
3.87%	66.78%	32.50%	0.71%	0.0%

Table 2: Overall accuracy of the model for the throughput

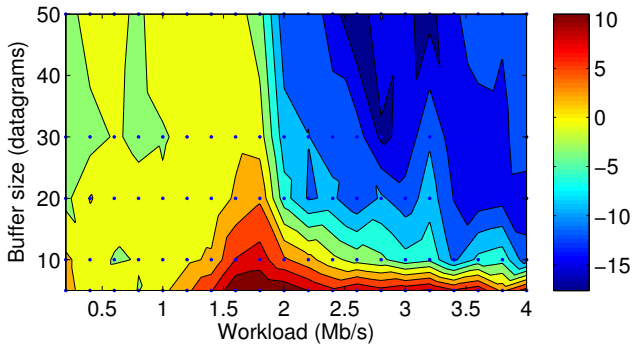

Figure 10: Percentage relative errors for the total loss probability for various buffer sizes K_i and workload levels λ_1 with x_2=100 and x_3=400 meters

Average	$< \pm 5\%$	$\pm$5-10%	$\pm$10-15%	$> \pm 15\%$
6.37%	47.0%	23.0%	27.0%	3.0%

Table 3: Overall accuracy of the model for the datagram loss probability

5.1 Model accuracy

To evaluate the accuracy of our model, we compare its performance parameters obtained with an implementation in MATLAB with those delivered by a discrete-event simulator (Network Simulator version 2.35 - *ns-2.35* [13]) for a large set of possible chains with four nodes. Note that all simulation results have been performed by generating 100,000 packets at the source node. Throughout this section, we define the percentage relative error of our model versus the actual values (delivered by *ns-2.35*) as the ratio 100× *(approximate - actual) / actual*. To simplify the presentation of the results, the four nodes of the chain are scattered in a straight line as illustrated by Figure 8. Nodes 1 and 4 are steady while the positions of nodes 2 and 3 vary. We denote by x_i the distance between node i and node 1. Values of x_2 are within the interval [110, 350] meters, while x_3 belongs to the interval [400, 690] meters. x_4 is constantly set to 750 meters. Note that the positions of nodes 2 and 3 must obey certain rules so that 1-hop neighbors can communicate. This is the reason behind the white "impossible area" band in Figure 9. Aside from x_i, two additional parameters can be tuned in our scenario: the size of buffers K_i, and the workload rate λ_1. Datagrams have a size of 1500 bytes.

First, we focus on the ability of our model to provide fair predictions for the expected throughput of the 4-nodes chain using relation (6). Figure 9 represents the relative error value on the chain throughput for our proposed model as a function of the distances of both relay nodes 2 and 3 to node 1. In this example, the buffers are of length $K_i = 20$, and the workload is set to a high, but not excessive level, which corresponds to a datagrams arrivals rate at the source node (node 1) of $\lambda_1 = 2.0$ Mb/s. We choose this value of λ_1 because it leads to analytical difficulties as the buffer at node 1 is neither completely full nor empty. Note that this figure (as well as the forthcoming Figure 10) corresponds to hundreds of data points explored (both by the simulator and by the model), and the surfaces shown are obtained using an interpolation from sets of scattered data points. The relative error tends to be low as it stands below 10% for virtually all of the nearly 550 configurations we have performed to generate this figure. We observe that the relative error tends to attain its maximum value (near 10%) (in the top points of the figure) when the link between nodes 2 and 3 is at is maximum distance, causing very high values of BER and a frame loss probability exceeding 60%. Table 2 shows the overall distribution of relative errors in the throughput. We observe that the mean error is around 4%, in close to 90% of cases the error remains below 10% and it never exceeds 15% in all considered cases.

In our second example, we study the relative error in the datagram loss probability for the 4-nodes chain as a function of the size of buffers K_i, and of the workload rate λ_1. Let us remind that this latter probability reflects the proportion of datagrams that are lost somewhere during their delivery to the destination node (mostly due to buffer overflow) and can be computed using relation (7). We explore the accuracy of our model for hundreds of settings of the parameters K_i, ranging from a low value of 5 to 50, and λ_1, spanning from 0.2 (low traffic) to 4 Mb/s (saturation regime). We set $x_2 = 100$ and $x_3 = 400$ meters, and as shown by Figure 9, these latter values do not lead to an advantageous case. The accuracy results are reported in Figure 10 and Table 3. We note that, overall, the model predictions deviate from the actual values by typically less than 10%. Worst cases may lead to deviations reaching up to 15%, and, as expected, they tend to occur for large values of K_i (larger than 30) and moderately high values of λ_1, namely when the dynamic of the queues length at buffers is at its peak. Table 3 shows the overall absolute errors for the datagram loss probability. We observe that in almost 70% of the cases considered the relative error is below 10% and exceeds 15% in about 3% of cases. The mean absolute error is close to 6%.

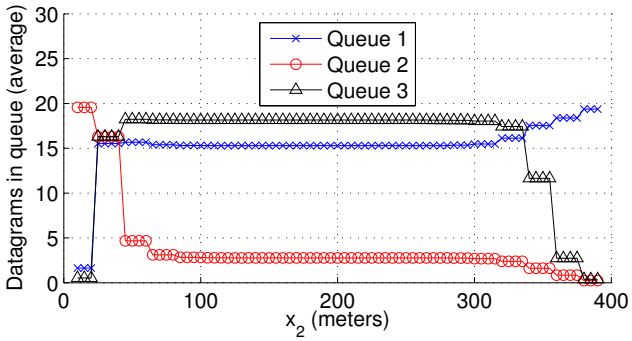

Figure 11: The mean buffer occupation as a function of the distance between nodes 1 and 2 (i.e., x_2) with K_i = 20 and λ_1 = 2.5 Mb/s

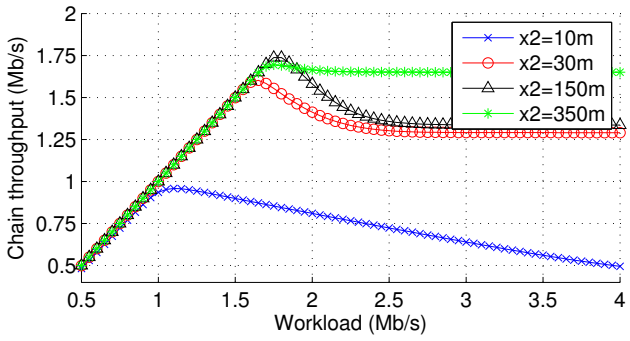

Figure 12: The expected throughput as a function of the workload level λ_1 for different positions of node 2 with K_i=20 and x_3=400 meters

These two exemples present only a small fraction of our numerical results. We have assessed the ability of our model to estimate the chain throughput and the loss probability for many other settings of the chain in terms of the workload rate λ_1, the size of buffers, K_i, the positions of relay nodes x_2 and x_3, and the size of datagrams. It is our conclusion that, in general, the accuracy of the model is good and that the results presented above reflect its typical behavior.

We now discuss the convergence behavior of the iterative scheme involved within the solution to our model. Although we do not have a theoretical proof of convergence, in the thousands of examples we have explored, the iteration never failed to converge within a small to moderate number of iterations (typically less than several tens). Therefore, and because the computational cost at each iteration is very small, the solution to hundreds of chain configurations (as it was the case to generate Figures 9 and 10) requires less than a minute for the MATLAB implementation of our model, while it takes about 12 hours to *ns-2.35* to deliver its corresponding results.

5.2 Model exploitation

Following the validation of the model accuracy, we provide two practical examples that illustrate how the proposed model can help in the deployment of a multi-hop wireless network. Of course, we cross-validate the correctness of the following results using our simulator (although they are not presented for the sake of readiness).

First, we rely on our model to study the behavior of the queue length in the buffer of nodes. Let us remind that datagrams are enqueued in buffers, waiting for transmission over the next link. We represent in Figure 11 the evolution of the mean queue length for nodes 1, 2 and 3 as a function of the distance between nodes 1 and 2, x_2. We set the rate of the workload to λ_1 = 2.5 Mb/s and we limit the buffer spaces to K_i = 20 datagrams. The nodes are scattered as follows: $x_1 = 0$, $x_3 = 400$ and $x_4 = 750$ meters, while we let x_2 vary in [5m, 390m] meters. Of course, because of the BER, the shorter the distance x_2, the poorer the quality of the link between nodes 2 and 3, and, the better the link between nodes 1 and 2. As a result, by gradually shifting node 2 away from node 1, the bottleneck link of the chain will also change. With small values of x_2, i.e. less than 50 meters, the BER for the link between nodes 2 and 3 is such that more than 60% of frames transmissions fail, which in turn dramatically extends the time node 2 takes for handling a datagram. Hence, the bottleneck occurs on the link between nodes 2 and 3 and a queue is built up at node 2 (clearly shown by Figure 11 where the mean buffer occupation is close to 20). For larger values of x_2, say between 60 and 350 meters, the BER is relatively small for every links such that one could have expected the bottleneck to be spread uniformly over the links. However, unlike the links between nodes 2 and 3, the two other links are exposed to the hidden node problem and thus have a slowdown pace of transmission. This is the reason why the queue at nodes 1 and 3 are close to the saturation while node 2 is not. Finally, for x_2 larger than 350 meters, the link between nodes 1 and 2 is greatly impaired (see Appendix). It becomes the bottleneck link, and so a large queue is built up at node 1 (see Figure 11). This first case study clearly points out the high sensitivity of a chain to the BER of its links, hence making its behavior hardly predictable by most analytical models.

In our second example, we focus on the presence of a throughput optimum (for a given level of workload λ_1) and on the associated performance collapse which occurs when the workload is set to excessive values. Multi-hop wireless chains are known to reach their best throughput level when the workload is capped under a certain threshold [11, 2]. In Figure 12 we represent the expected throughput of several 4-nodes chains for values of λ_1 spanning from low levels to high levels of workload with a buffer length K_i of 20. Each curve corresponds to the expected throughput for a specific spot of node 2 (expressed in terms of its distance to node 1, i.e., x_2) while the other relay node (i.e., node 3) is always set to 400 meters away from node 1 ($x_3 = 400$). As shown by this figure, our model captures the existence of this performance optima. We observe that the magnitude of these tipping points (turnaround) widely varies depending on the precise location of node 2. For instance the gap between the maximum value and the asymptotic value attained when the system is totally saturated goes up to 48% for $x_2 = 10$ meters, while it stands at only 4% in the case of $x_2 = 350$ meters. Moreover, Figure 12 clearly pinpoints that the maximum throughput of the chains peaks at different values of λ_1

depending on the position of the relay nodes. Our proposed model provides an accurate, simple, and fast way (unlike virtually endless simulations) to locate this optimal value of λ_1. The existence of this performance collapse and its accurate reproduction by our model enable the implementation of controlling mechanisms at the border of the chain (such as admission control and traffic shaping policies) to refrain the network from such congested and "counter-productive" states.

6. CONCLUSIONS

This paper presents a modeling framework to analytically evaluate the performance of multi-hop flow along a wireless chain of four nodes. Unlike existing approaches, the proposed model accounts for a non-perfect physical layer, handles the hidden node problem, and is applicable under workload conditions ranging from flow(s) with low intensity to flow(s) causing the network to saturate. Its solution is easily and quickly obtained and delivers estimates for the expected throughput and for the datagram loss probability of the chain.

The accuracy of the model is generally good and, importantly, it captures the performance collapse that takes place in certain wireless multi-hop networks. We assess its ability to deliver fair predictions for a large set of experiments (in which we change the chain topology, the wireless link quality, the workload conditions, the size of buffers, etc.) and we compare the obtained results with those coming from a discrete-event simulator. We believe the model is a first milestone in the development of analytical tools that could help the deployment, the management and the configuration of multi-hop wireless networks.

Acknowledgments

This work was funded by the French National Research Agency (ANR) under the projects ANR VERSO RESCUE (ANR-10-VERS-003) and ANR-JST PETAFLOW (ANR-09-BLAN-0376).

7. REFERENCES

[1] T. Abreu, B. Baynat, T. Begin, and I. Guérin-Lassous. Hierarchical Modeling of IEEE 802.11 Multi-hop Wireless Networks. In *Proceedings of ACM MSWiM*, pages 143–150. ACM, 2013.

[2] T. Abreu, N. Nguyen, T. Begin, I. Guérin-Lassous, and B. Baynat. Substitution Networks: Performance Collapse Due to Overhead in Communication Times. In *Proceedings of ADHOCNETS*, volume 1, pages 1–16, October 2012.

[3] A. Aziz, M. Durvy, O. Dousse, and P. Thiran. Models of 802.11 multi-hop networks: Theoretical insights and experimental validation. In *IEEE COMSNETS*, 2011.

[4] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE JSAC*, 18(3), 2000.

[5] C. Chaudet, D. Dhoutaut, and I. G. Lassous. Experiments of some Performance Issues with IEEE 802.11b in Ad Hoc Networks. In *Proceedings of WONS*, pages 158–163, 2005.

[6] M. Fiore. http://perso.citi.insa-lyon.fr/mfiore/research.html.

[7] M. Garetto, T. Salonidis, and E. W. Knightly. Modeling per-flow throughput and capturing starvation in CSMA multi-hop wireless networks. In *Proceedings of IEEE INFOCOM*, 2006.

[8] M. Garetto, J. Shi, and E. W. Knightly. Modeling media access in embedded two-flow topologies of multi-hop wireless networks. In *Proceedings of ACM MobiCom*, 2005.

[9] M. M. Hira, F. A. Tobagi, and K. Medepalli. Throughput analysis of a path in an IEEE 802.11 multihop wireless network. In *Proceedings of IEEE WCNC*, 2007.

[10] IEEE 802.11 Working Group and others. IEEE 802.11-2012 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. *IEEE 802.11 Wireless LAN Standards*, 2012.

[11] J. Li, C. Blake, D. S. J. D. Couto, H. I. Lee, and R. Morris. Capacity of Ad Hoc Wireless Networks. In *Proceedings of MOBICOM*, pages 61–69, 2001.

[12] B. Nardelli and E. W. Knightly. Closed-form throughput expressions for CSMA networks with collisions and hidden terminals. In *Proceedings of IEEE INFOCOM*, 2012.

[13] NS2. http://www.isi.edu/nsnam/ns/.

[14] I. Tinnirello, G. Bianchi, and Y. Xiao. Refinements on IEEE 802.11 distributed coordination function modeling approaches. *IEEE Transactions on Vehicular Technology*, 59(3), 2010.

[15] V. Vishnevsky and A. Lyakhov. 802.11 lans: Saturation throughput in the presence of noise. In *Proceedings of IFIP Networking*, pages 1008–1019, 2002.

[16] X. Wang and K. Kar. Throughput modelling and fairness issues in CSMA/CA based ad-hoc networks. In *Proceedings of IEEE INFOCOM*, pages 23–34. IEEE, 2005.

[17] K. Xu, M. Gerla, and S. Bae. Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks. *Ad Hoc Networks*, 1, 2003.

APPENDIX

Figure 13 shows the evolution of BER (Bit Error Rate) and of the corresponding FER (Frame Error Rate) for datagram of 1500 bytes as a function of the distance between 2 communicating nodes (from 150 to 399 meters). For more details on the derivation of these values for the BER, the reader can refer to [6]. It is worth noting that the FER can reach values up to nearly 60% when two neighbor nodes are spaced by more than 350 meters.

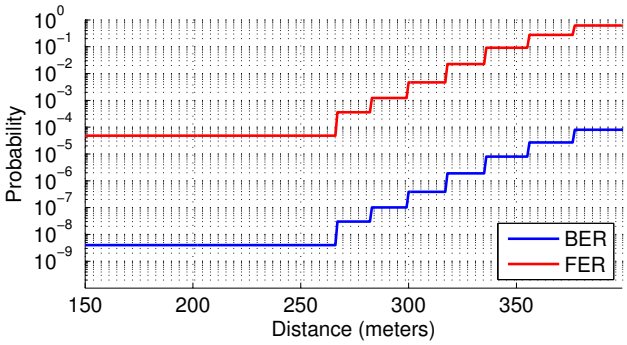

Figure 13: Evolution of BER and FER in our scenario as a function of the distance for a datagram size of 1500 bytes