Asymptotic delay analysis and timeout-based admission control for ad hoc wireless networks with asymmetric users

Rachid El-Azouzi, Essaid Sabir, Sujit K. Samanta, Ralph El-Khoury

Abstract

In this paper, we present an analytical model for an approximate calculation of the end-to-end delay performance in multi-hop wireless ad hoc networks. In contrast to literature that largely focuses on average delay, our paper focuses on the distribution of end-to-end delay. In this paper, we assume that each source injects packets in the network, which traverse intermediate nodes until they reach the destination. Firstly, we employ discrete-time queueing theory to derive the expressions for the queue length and the delay in terms of probability generating functions. Secondly, in order to improve the control routing and transmission scheduling, we adopt a cross-layer design that allows information sharing across different layers for efficient utilization of network resources, and meeting the end-to-end performance requirements of demanding applications. Thirdly, we propose a packet admission control scheme based on delay timeout mechanism. This guarantees quality of service for multimedia applications such as voice and video streaming. Finally, we conduct extensive simulations in order to verify our analytical results.

1. Introduction

An ad hoc wireless network is a collection of wireless nodes that communicate with each other without any established infrastructure or centralized control. Due to the limited transmission range of wireless network interfaces, multiple network “hops” may be needed for a given node to exchange data with another across the network. In such a network, the packets may have to be forwarded by several intermediate nodes before they reach their destinations, and therefore each node operates not only as a host but also as a router. Thus each node may be a source, destination and relay (intermediate). Many factors interact with each other to make the communication possible like routing protocol and channel access method. With the emergence of real-time applications in wireless networks, delay guarantees are increasingly required. In order to provide support for delay-sensitive traffic in such network, an accurate evaluation of the distribution of delay is a necessary first step. Knowing the nature of the multi-hop ad hoc networks, many factors are crucial for the study of the end-to-end (e2e) delay. We cannot study separately the delay generated by a given layer without considering the others. Hence we adopt a cross-layer architecture with its potential synergy of information exchange between different layers, instead of the standard OSI non-communicating layers. Many studies of packet delay and loss in various network environments have been reported in the literature. A large number of studies on multi-hop wireless networks have been devoted to system stability and throughput. The delay performance has been studied under particular topologies (as linear networks or grid networks) or under uniform traffic distribution. It should be pointed out that due to the lack of analytic solutions, many studies of packet delay and loss behavior have been conducted with simulation and experimental approaches. In this paper, we provide a framework for cross-layer of delay distribution in the context of wireless ad hoc networks. The analysis takes into account the queueing delays at source and intermediate nodes. The delay of a path (we will also refer to it as connection) of this network depends on the number of nodes, the source traffic characteristics, the number of re-transmissions at nodes, the forwarding cooperation level and the behavior of the MAC protocol. We assume that time is slotted into fixed length time frames. At any time slot, a node having a packet to be transmitted to one of its neighboring nodes decides with some fixed probability in favor of a transmission attempt. If there is no other transmissions by other nodes those may interfere with the node under consideration, the transmission is successful. As examples of this mechanism, we find Aloha and Carrier Sense Multiple Access (CSMA) type protocols. We consider a parameter that mea-
sures the aptitude of a node to forward packets coming from its neighbors. At any instant of time, a node may have two kinds of packets to be transmitted: (1) packets generated by the node itself; data or control packets, and (2) packets from other neighboring nodes those need to be forwarded. To carry these two types of packets, we consider two separate queues handled with a weighted fair queueing (WFQ) discipline. We focus on the asymptotic properties of the delay due to buffering of packets at network layer and random access protocol on MAC layer. The analysis is done using the probability generating function approach which allows us to estimate the distribution of delay at intermediate nodes for all ongoing connections.

Supporting real-time flows with delay and throughput constraints is potentially important for wireless multi-hop networks. From the network designer’s point of view, one major concern is how long it will take a packet to reach the destination. In this paper we investigate an important issue for real-time multimedia applications in which a large delay and jitter will be unacceptable. Such application requires receiver playback buffers to smooth network delay variation and reconstruct the periodic nature of the transmitted packets. Packets arriving after their scheduled deadline are considered late and are not played out. This requires that the network should be able to offer quality of service (QoS) appropriate for the delay bounds of the real-time application constraints. Our analysis results allow us to study the impact of bounded delay on throughput. We compute the rate of packets arriving before their scheduled playout time (delay constraint). A major focus of the paper is understanding the impact of multi-hops, the source traffic characteristics, the number of retransmissions at nodes, the forwarding cooperation level and the behavior of the MAC protocol. Based on the analysis, we provide a way to obtain a trade-off between end-to-end delay and throughput. Furthermore, we propose a cross layer admission control including the network and the MAC layer to support real-time traffic. This scheme is useful to reduce the loss rate by decreasing the packets arrived after their scheduled deadline.

1.1. Main contributions

The main contributions addressed in this work are:

- The distributed cross-layer scheme proposed here, besides of its novelty and efficiency, is characterized by its high simplicity. It does not need any external information, but a local decision can be taken with the help of routing information from the network layer as well as the MAC layer.
- We derive a mathematical framework based on probability generating function (PGF) approach to estimate the distribution of delay.
- Unlike [1] and [17], we relax the symmetry assumption. Indeed, our analysis takes into consideration the users and the topology asymmetry. Therefore, each user may have different Network/MAC layer intrinsic parameters (such as attempt rate and cooperation level) and may experience different extrinsic factors (such as the collision probability) due to asymmetric topology and nodes local density.
- In contrast to [4] where average end-to-end goodput is calculated based on approximation, we derive here a closed form of the e2e goodput and then conclude the exact value of packet admission rate (the probability that the end-to-end delay of a connection does not exceed the timeout delay).
- Getting the distribution of delay for each source/destination connection, we investigate an important issue for real-time and interactive data services (e.g., conversational and streaming flows) over multi-hop ad hoc networks. A fundamental feature of the streaming service is that the content is played back at the receiver during the delivery. Instead of satisfying a low delay bound as conversational services, streaming services need to maintain a continuous steady flow for smooth playback. In other words, conversational flow has a hard constraint on delay, whereas streaming flow has to solve the jitter problem in addition to the delay relatively soft constraint. In order to play the receiver stream, an application buffers the packets and plays them out after a certain deadline to get again a periodic stream at the application level. Packets arriving after their corresponding delay timeout are lost and then not played out. We further define an admission control to bound the e2e delay and then an acceptable service quality may be guaranteed.

1.2. Prior work

In [1], Bianchi model dealt with the behavior of the binary back-off counter at one tagged node as a discrete Markov chain with two-dimensional state. At steady state and based on the remark that each transmission “sees” the system in the same state, computation of the transmission and the collision probabilities becomes possible. Then, he analyzes the saturation throughput under the assumption that in each transmission attempt, regardless of the number of retransmissions, each packet collides with constant and independent probability. Kumar et al. [10] present a fixed point analysis of Bianchi’s model, and give closed form for the collision probability, the aggregate attempt rate, and the aggregate throughput in the asymptotic regime of a large number of nodes. All these studies focus on single-cell WLANs. Yang et al. [17] extend and characterize the channel activities in IEEE 802.11 DCF-operated multi-hop wireless networks from the perspective of an individual sender and under saturation condition. This is natural since a consistent view for the entire network cannot be symmetric in a multi-hop topology: A node may detect the channel to be busy while another node senses the channel to be idle. Later, they study the impact of the transmit power and the carrier sense threshold on the channel efficiency.

Many papers in the literature have studied the problem of cooperation in ad hoc networks, see [12,15]. In [4] and [8], authors worked with the above mentioned system model, and studied the impact of routing, channel access rates and weights of the weighted fair queueing on throughput, stability and fairness properties of the network. Important insights were revealed into various tradeoffs that can be achieved by fine-tuning certain network parameters. The throughput maximization of the multi-hop wireless networks has been extensively studied in [6] and [9]. However, it is shown that the high throughput in an ad hoc network is achieved at the cost of a high amount of delay. In [5], the authors characterized the delay-throughput tradeoffs in wireless networks with stationary and mobile nodes. These problems have drawn our attention to the relation between the delay characteristic and the throughput. However, most of the related studies do not consider the problem of forwarding. The authors in [11] contributed to quantifying the impact of hidden nodes on the performance of linear wireless networks based on the IEEE 802.11 protocol and taking into consideration the effects of queueing and retransmissions at each node. In [13], authors provided closed form expressions for the queue length in the presence of arbitrary arrival patterns, packet size distributions and finite network load. In [16], using the decomposition approach authors analyzed the e2e delay of wireless multi-hop networks for two MAC schemes, m-phase TDMA and slotted aloha, and related references therein. They considered the arrival processes to every node are only delayed versions of the original traffic flow generated at the source node.

The rest of the paper is organized as follows. In the next section, we introduce the general model framework. We study the distribution of delay in multi-hop ad hoc networks in Section 3. Based on
the conducted survey, we compute the rate of packets arrived before their scheduled playout time in Section 4. In Section 5, we validate the analytical results using a discrete time simulator and carry out extensive numerical examples.

2. Wireless model and cross-layer architecture

We consider a collection of autonomous nodes able to communicate with other nodes in their respective direct range. Each one can reach nodes those are outside of its direct range by communicating indirectly through intermediate nodes those forward packets towards the required destination. We assume that nodes use the same channel for transmitting with an omnidirectional antenna. A node \( j \) receives successfully a packet incoming from a node \( i \) if and only if there is no interference at the node \( j \) due to another simultaneous transmission. It also follows that a node cannot receive and transmit at the same time slot because of the use of a single channel. Each node \( i \) handles two separated buffers: buffer \( Q_i \) carries the own packets of \( i \) and buffer \( F_i \) carries packets originated from a given source, to be forwarded to neighbors till achieving the final destination. Fig. 1 depicts a typical example the associated double buffering network that can be used to study the multi-hop ad hoc network. These two queues are considered to have infinite storage capacity, packets inside are served with a First-In-First-Out fashion and are managed using Weighted Fair Queuing scheduling. The buffer \( F_i \) is selected for transmission with probability \( f_i \). Since we assume that each node has always packets to send from queue \( Q_i \), then it follows that queue \( Q_i \) is selected with probability \( 1 - f_i \), where \( f_i \) is the probability that queue \( F_i \) has at least one packet. The forwarding capability permits each node to behave as a router and this allows to relay packets originated from a source \( s \) to a destination \( d \). Routing tables that ensure the network reachability and define which neighbors to use to reach any given destination are periodically updated using a proactive routing protocol as OLSR (Optimized Link State Routing). We use throughout this paper the notation \( R_{s,d} \) to denote the set of intermediate nodes in a path between a source \( s \) and a destination \( d \) (\( s \) and \( d \) not included).

MAC layer protocols play doubtlessly the most important role in the communication chain. Consequently, studying medium access methods have got a particular attention by the researchers community from earlier years. This way, many access and resources reservation methods have been elaborated to ensure performance guarantee. There exists two major families of dynamic access methods: Deterministic access such as token ring and token bus, and random access such as aloha and CSMA with all their variants and improved versions. It has been shown that an IEEE 802.11-operated multi-hop ad hoc network is reasonably equivalent to a multi-hop ad hoc network operating with slotted aloha protocol. Indeed, this result becomes intuitive by considering the definition of virtual slot, i.e., the mean time (in slots) that the system stays in a given state (idle, busy, success and collision). This way, we only need to be careful about the individual transmission probabilities that should be the solution of Bianchi [1], Kumar et al. [10] and Yang et al. [17] fixed point problems. Henceforth, it is plausible to consider a channel access mechanism only based on a probability to access the network, i.e., when a node \( i \) has a packet to transmit, it accesses the channel with a probability \( P_i \). In IEEE 802.11 Distributed Coordination Function based ad hoc system, the attempt rate is given by [17]

\[
P_i = \frac{2(1 - 2\gamma)}{(1 - 2\gamma)(CW_{\text{min}} + 1) + \gamma CW_{\text{min}}(1 - 2\gamma)^m},
\]

where \( \gamma \) is the conditional collision probability given that a transmission attempt is made, \( CW \) is the contention window and \( m = \log (\frac{CW_{\text{max}}}{CW_{\text{min}}}) \) is the maximum of backoff stage. For example, in IEEE 802.11 standard, \( P_i \) depends on the number of neighbors, on the backoff mechanism and the probability of collision, see Yang et al. [17] for the ad hoc extension of Bianchi results [1]. Problems of hidden terminals or exposed terminals known with the IEEE 802.11 are included implicitly in the formula of \( P_i \). It is clear that the scheduler of transmission overall the network depends on \( P_i \). We assume that each node is notified about the success or failure of its transmitted packets. A transmission only fails when there is an interference on the intended receiver, in other terms, when a collision occurs on the receiver. The only source of packet loss is due to collisions.

For a reliable communication, we fix a limit number of successive transmissions of a single backlogged packet, after that it will be dropped definitively. We denote \( K_{\text{c,d}} \) the maximum number of successive collisions allowed for a single packet sent from the node \( i \) on the path \( R_{s,d} \). Unlike the OSI model where layers are clearly separated, we jointly consider network and MAC layer parameters, see Fig. 2. This allows communication and information exchange between different layers and henceforth is more powerful, flexible, allows global optimization and in particular permits manipulating the cooperation level. For ease of reading, we summarize the main notations in Table 1.

3. Delay distribution analysis

For soft real-time applications, which are delay-sensitive but loss tolerant, delay distribution is an important quality of service

---

**Fig. 1.** Queuing network model for multi-hop wireless ad hoc network with double buffering.

**Fig. 2.** Proposed cross-layer architecture: Accessing the channel starts by choosing the queue from which a packet would be selected. Then, this packet is moved from the network layer to the MAC layer, where it will be transmitted and retransmitted, if needed, until success or definitive drop. This way, it is clear that the end-to-end QoS (mainly, throughput and delay) depends on several layers as well as the cooperation level \( f_i \) of intermediate nodes, i.e., those that play relays role.
(QoS) measure of interest. In order to effectively support delay-sensitive applications such as video streaming and interactive gaming in an ad hoc wireless network, it is crucial and challenging to develop feasible methodologies and techniques for accurately analyzing, predicting and guaranteeing the end-to-end delay performance over multi-hop wireless networks. Our analysis takes into account the queuing delays at source and intermediate nodes of random access multi-hop wireless ad hoc networks. However, the delay is defined as the time taken by a packet to reach the destination after it has left the source. We aim behind this to determine the distributions of the number of forwarding packets and their sojourn times in the system. We focus our study on the forwarding queue $F_i$ of a given node $i$. In earlier work [4], we have shown that packets arrive to $F_i$ (i.e. packets received successfully by node $i$) according to a random arrival process with average $\lambda_i$, it can be written as:

$$\lambda_i = \sum_{s,d \in R_{i,d}} \lambda_{i,s,d}$$

$$= \sum_{s,d \in R_{i,d}} \frac{(1 - \pi_{s,d})P_{s,d}}{L_s} \prod_{k \in R_{i,s}} [1 - (1 - P_{k,s,d})^{K_{s,d}}],$$

where $\pi_i$ is the probability that queue $F_i$ has at least one packet to be forwarded in the beginning of each cycle, $\lambda_{i,s,d}$ is the arrival flow into relay $i$ concerning path $R_{i,d}$ and $P_{s,d}$ is the probability that the node $s$ generates and sends a packet to node $d$. We distinguish two types of cycles: The forwarding cycles related to the packets of $F_i$ and the source cycles related to the packets coming from $Q_s$. Moreover, due to asymmetric topology each transmission cycle has a different size for each path. Indeed, the beginning of each cycle defines the departure instants of forwarding packets in the two cases of the system (for $n_i = 0$ and $n_i \neq 0$). This leads to the following balance equation of the transmission probability of all neighbors of the next hop after $k$ on the path from $s$ to $d$. We denote that $\lambda_i$ is exactly the aggregate arrival rate of packets from different paths and different kind of connections to the forwarding buffer $F_i$ of node $i$.

When a packet leaves the network layer, it stays in the MAC layer (server) for some arbitrary number of slots depending on the attempt rate and the collision probability. One concludes that the forwarding queue $F_i$ constitutes a G/G/1 queue that has some special characteristics due to the presence of saturated queue $Q_i$. Furthermore, we will derive the desired distributions using the PGF approach. For sake of simplicity, in the following, we omit the index $i$ that identify the node $i$ itself to facilitate the notations and the reading, e.g., $F_i \equiv F$. Also, the notations indicating connections identities will be omitted until contraindicate. Each connection $R_{i,d}$ has its own service time which depends on the topology (set of neighbors), the transmission probability of nodes and the limit number of transmissions per packet.

- Let $r$ denote the number of arrival packets to the buffer $F$ during the residual service time of a packet which is picked from buffer $Q$ and seen by an arrival packet to buffer $F$.
- Let $d'$ denote the number of arrival packets to the buffer $F$ during a service time of a packet picked from forwarding buffer $F$.
- Let $d$ denote the number of arrival packets to the buffer $F$ during the $j$th packet service time of packets picked from the own buffer $Q$.
- Let $n_i$ be the instantaneous number of packets in the buffer $F$.

Fig. 4 shows the evolution of MAC service in terms of cycles (Q and F transmission cycles), it shows also the departure instants $\zeta_i$ of forwarding packets in the two cases of the system (for $n_i = 0$ and $n_i \neq 0$). This leads to the following balance equation of the number of packets in $F$ at departure instants:

$$n_{i+1} = \begin{cases} r + \sum_{j=1}^{m} d^0_j + d^0, & \text{for } n_i = 0, \\ n_i + \sum_{j=1}^{m} d^0_j + d^0 - 1, & \text{for } n_i \neq 0, \end{cases}$$

where $m$ represents the number of consecutive packets which are from buffer $Q$, taking service before the next packet from buffer $F$. For the case $n_i = 0$, the second term $\sum_{j=1}^{m} d^0_j = 0$ if there is no packet in the buffer $Q$ (except the residual service packet) which will take
for service before the next packet from buffer F. Putting the two cases of \( n_{i+1} \) together in one equation, we get

\[
n_{i+1} = n_i + r(I(n_i)) + \sum_{j=1}^{m} a^j + a^0 - 1 + I(n_i), \quad \forall i.
\]

(5)

where \( I(n_i) \) is an indicator defined by

\[
I(n_i) = \begin{cases} 
1, & \text{if } n_i = 0, \\
0, & \text{else}. 
\end{cases}
\]

(6)

We now focus on the solution of the difference Eq. (5) in the domain of the generating functions to derive all distributions of interest. The following proposition gives the distribution of the length of buffer \( F_i \).

**Proposition 1.** The PGF of the number of packet \( P(z) = \sum_{n=0}^{\infty} P_n z^n \) in the forwarding queue seen by a departure is given by

\[
P(z) = \frac{P_0(zR(z) - 1) fA'(z)}{z - (1 - f) z A(z) - f A'(z)}.
\]

(7)

**Proof.** See Appendix B. \( \square \)

For detail derivations of distributions of new arrivals in a service time denoted by \( A'(z) \) and \( A(z) \) and the distribution of new arrivals in residual time \( R(z) \), see Appendix A. We now turn to the calculation of how long a packet spends in an intermediate node. From queueing theory, there is a relation between the PGF of the number of packets in the buffer and the PGF of waiting time. Considering a First-In-First-Out fashion, it is clear that the packets left behind are precisely those arrived during its stay in the buffer.

**Remark 1.** Due to the presence of the saturated queue \( Q_s \), the minimum waiting time in the forwarding queue \( F_i \) is one time slot. Similarly, the service time is at least one time slot. It follows that the average delay (waiting + service) in the relay node \( i \) cannot be less than two time slots.

Thus, we have

\[
P(z|t) = \frac{\lambda^2}{n=0} z^n \left( \frac{t-1}{n} \right)^n = (1 - \lambda z)^{t-2}.
\]

Denote the total time spent in the system for this customer by the random variable \( D \) with distribution

\[
P(z) = \sum_{t=2}^{\infty} P(z|t) P(D = t) = \sum_{t=2}^{\infty} (1 - \lambda z)^{t-2} P(D = t)
\]

\[
= \frac{D(1 - \lambda z)}{(1 - \lambda z + \lambda z)^t}
\]

and the next result follows.

**Proposition 2.** The PGF of the waiting time in the system \( D(z) = \sum_{n=0}^{\infty} d_n z^n \) is given by

\[
D(z) = z^2 P \left( \frac{z}{\lambda} - 1 + \lambda \right)
\]

(8)

and then we can come out easily the expression of end-to-end delay PGF using \( D_{i,d}(z) = \prod_{i=0}^{k} D_i(z) \).

**Lemma 1.** The expected waiting time and the variance of waiting time at node \( i \) are, respectively, given by

\[
D'(1) = 2 + \frac{P'(1)}{\lambda}
\]

(9)

and

\[
\text{Var}[D] = D'(1) + D'(1) - [D'(1)]^2
\]

(10)

where \( D'(1) = 2 + \frac{2P(1)}{\lambda} + \frac{P(1)}{\lambda^2} \), where \( \phi(1) \) and \( \phi'(1) \), respectively, represent the first and second order derivatives of any PGF \( \phi(z) \) at \( z = 1 \). Since we are interested to derive the end-to-end delay of some given connection, we should not consider time elapsed due to dropped packets. Then, we have to deduct it from the total delay (waiting and service times) at node \( i \). The average number of successful transmissions and the average time spent by dropped packet at source \( s \) are, respectively, given by

\[
L_{s,t}^{\text{acc}} = \sum_{k=0}^{\infty} k (1 - P_{s,s}) k^{-1} P_{s,s}
\]

and

\[
L_{s,t}^{\text{drop}} = \sum_{k=0}^{\infty} (D'(1) - \frac{D'(1)}{k!})
\]

(11)

The first term is the average service time at the source \( s \), whereas the term inside symbol sum is exactly the average waiting and service time in intermediate nodes.

**4. Application: playout delay control**

Due to its vast potential for providing ubiquitous communication, ad hoc and the emerging mesh networking have received overwhelm interest over the last years. This way, several research works have been done to claim the ability of supporting multimedia applications over ad hoc networks [2] and [14]. In this section, we deal with multimedia applications over ad hoc networks. Our analysis is applicable for both conversational (e.g., VoIP, Gaming, . . .) and streaming services (e.g., VoD, TV, . . .). Supporting these classes of services over wireless medium is very challenging due to many factors that cause high error rate. In such interactive multimedia applications, packets loss and connection reliability deterioration are generally caused by delay, jitter (unexpected phase variation) and decoding errors. With a self-managing nodes such as ad hoc networks, the problem becomes more complicated due to the absence of a central entity that monitors the instantaneous changes in the network. In order to enable supporting real-time
services, some QoS demand should be stochastically fulfilled (e.g., the average goodput should be strictly guaranteed or a maximum delay should not be exceeded).

The newly designed H.264 video coding standard has been developed such as to support wireless medium [7]. Here, we assume that the ongoing application buffers received packets and plays them out after a given deadline. Henceforth, a packet arriving after the deadline will not be played out. Real-time Transport Protocol (RTP) defines a standardized packet format for delivering audio and video over the Internet. It is used extensively in communication and entertainment systems that involve streaming media. RTP time-stamps, see Fig. 5. Frames are held in the playout buffer for a specific playout buffer sorted by their RTP time-stamps, see Fig. 5. Frames are held in the playout buffer for a period of time to smooth timing variations generated while crossing intermediate nodes in the network. Holding the data in a playout buffer also allows the pieces of fragmented frames to be received and grouped, and it allows any error correction data to arrive. Potential remaining errors of the frames are then concealed, and the media is rendered for the user.

4.1. End-to-end goodput

If we denote the maximum tolerable delay for connection \( R_{sd} \) by \( \Delta_{sd} \), then the end-to-end goodput (effective throughput) can be written as

\[
goodput_{sd} = thp_{sd} \cdot P(D_{sd} \leq \Delta_{sd}).
\]

where \( thp_{sd} \) is the end-to-end throughput without the delay control which is given by \( thp_{sd} = \ell_{sd} / C_{sd} \), see Eq. (2), and \( P(D_{sd} \leq \Delta_{sd}) \) is the probability that the cumulative delay does not exceed the application layer threshold \( \Delta_{sd} \). We call it the end-to-end packets admission rate. It is given by next lemma.

**Lemma 2.** Let \( l = |R_{sd}| \) be the number of intermediate nodes (s and d not included) in route \( R_{sd} \). The probability that the end-to-end delay is exactly \( j \) slots is

\[
P(D_{sd} = j) = \sum_{l_1=2}^{j-2l} \sum_{l_2=2}^{j-2l} \cdots \sum_{l_l=2}^{j-2l} \prod_{i=2}^{l} P(D_{i,s,d} = j_i),
\]

\( j \geq 2l \).

**Proof.** We present here a mathematically non-rigorous, but intuitive, derivation of the probability that the e2e delay is exactly \( j \) time slots. It is easy to see that \( P(\bar{D}_{sd} \leq \Delta_{sd}) = \zeta(\Delta_{sd}) \), where \( \zeta(\cdot) \) is the Cumulative Distribution Function (CDF) of the e2e delay, i.e., the probability that the random variable \( \bar{D}_{sd} \) takes on a value less than or equal to \( \Delta_{sd} \). Since a packet cannot stay less than two slots on an intermediate relay, then the minimum delay on route \( R_{sd} \) after leaving the source s is 2l, it follows that

\[
P(\bar{D}_{sd} \leq \Delta_{sd}) = \sum_{j=2l}^{\Delta_{sd}} P(D_{sd} = j).
\]

where \( P(D_{sd} = j) \) is the probability that the e2e delay is exactly \( j \) slots. It can be computed considering the set of all partitions of the integer \( j \) and taking into account the possible permutations. It follows that

\[
P(D_{sd} = j) = \sum_{l_1=2}^{j-2l} \sum_{l_2=2}^{j-2l} \cdots \sum_{l_l=2}^{j-2l} \prod_{i=2}^{l} P(D_{i,s,d} = j_i),
\]

\( j \geq 2l \),

given that \( d_n = P(D_{i,s,d} = n) \) can be calculated by differentiating \( n \) times the polynomial \( D(z) = \sum_{n=0}^{\infty} d_n z^n \) at \( z = 0 \), provided by result (8).

**Example.** Let us consider a linear topology with five nodes. Node 1 is the source, nodes 2, 3 and 4 are relays and node 5 is the final destination. Consider the simple scenario where source 1 forwards to relay 2, relay 2 forwards to relay 3, relay 3 forwards to relay 4 and this latter forwards to the destination 5. The probability that the e2e delay is exactly 8 slots is

\[
P(D_{1,5} = 8) = \frac{2}{9}P(D_{2,15} = 2)P(D_{2,15} = 2)P(D_{4,15} = 4) + \frac{2}{9}P(D_{2,15} = 2)P(D_{3,15} = 3)P(D_{4,15} = 3) + \frac{2}{9}P(D_{2,15} = 2)P(D_{4,15} = 2)P(D_{4,15} = 2) + \frac{2}{9}P(D_{2,15} = 3)P(D_{3,15} = 3)P(D_{4,15} = 2) + \frac{2}{9}P(D_{2,15} = 3)P(D_{4,15} = 2)P(D_{4,15} = 2) + \frac{2}{9}P(D_{2,15} = 4)P(D_{4,15} = 2)P(D_{4,15} = 2).
\]
4.2. Distributed dynamic retransmissions algorithm (DDRA)

When a packet is traveling over a multi-hop ad hoc network, it experiences different transmission success probabilities due to the asymmetric topology. Intuitively, to enhance the multi-hop reliability and then to improve the success probability on the intermediate relays, each relay should fine-tune its intrinsic network/MAC parameters according to its instantaneous environment perception. This kind of auto-configuration scheme is quite simple but needs a relatively high amount of external information. Here, we describe a simple and fully distributed algorithm that was first described in [3]. We remark that giving more chance to packets that arrived near final destination or the ones whom accumulative de-retransmission scheme jointly with a technique, i.e., the retransmission delay, we propose to integrate the dynamic time (a threshold is to be defined by higher layers). To reduce the average queue size (or equivalently the effective load) of the connection that suffers from huge delay is set to its respective $K_s$.

As described in [3], we remark that giving more chance to packets that arrived near final destination or the ones whom accumulative de-retransmission scheme jointly with a technique, i.e., the retransmission delay, we propose to integrate the dynamic time (a threshold is to be defined by higher layers). To reduce the average queue size (or equivalently the effective load) of the connection that suffers from huge delay is set to its respective $K_s$.

Algorithm 1: Distributed Dynamic Retransmission Algorithm (DDRA)

```
1: for each relay node $i \in R_{s,d}$ do
2: \hspace{1cm} Update the retransmissions limit: $K_{s,d} = \min (K_{s,d} + h \cdot \eta_{s,d} K_{max});$
3: \hspace{1cm} Update the hop sequence: $h = h + 1;$
4: end for
```

5. Numerical examples

We now turn to study a typical example of ad hoc networks. We consider an asymmetric static network formed by 11 nodes as shown in Fig. 6. We established five connections (or streams) $a, b, c, d$ and $e$. Two nodes are neighbors if they are connected with a dashed or solid line. For illustrative purpose, we consider that the time slot duration is $\delta = 100$ $\mu$s and all nodes are supposed to have the same transmissions limit $K$ per packet. In order to get stability for all nodes, let $P_0 = P_3 = P_7 = P_8 = 0.3$, $P_4 = P_{10} = 0.4$ and $P_5 = 0.5$ over all the realized simulations. We present extensive numerical and simulation results to show the accuracy of our method. For that aim, a discrete-time simulator that implements the model of Section 2 is used to simulate the former network. In order to smooth out the simulation plots, we performed at least 10 runs per simulation and then took the average values.

5.1. Model validation

All involved nodes are considered to be cooperative and their forwarding probabilities (cooperation levels) are set to $f_i = f = 0.8$. While $P_i = P$ is varying for all nodes $1, 6, 9$ and $11$, Figs. 7 and 8, respectively, show (from analytical model as well as simulation results) the average delay in intermediate relays and the average end-to-end delay of considered connections. The results based on our analytical approach are close to the simulation results. This is also true in the case where nodes forward to different neighbors on different paths. However, one can see a sharp gap which is perhaps due to the approximation of the number of consecutive $Q$ cycles.

5.2. Impact of cooperation level $f_i$

Here, we address a crucial parameter that impacts significantly the end-to-end reliability. We fix here the transmission probabilities as follows $P_1 = P_{10} = P_{11} = 0.4$, $P_3 = 0.3$. For simplicity and without any loss of generality, we consider the same cooperation level for all nodes, i.e., $f_i = f$. If nodes were selfish ($f = 0$) then the e2e delay may go to infinity and the throughput is minimized. When nodes are altruistic ($f = 1$), the e2e delay is minimized and a maximum throughput can be achieved. But when $0 < f < 1$, we note that above some threshold (depending on transmissions probability vector) all forwarding buffers become stable. We refer to this region as the system stability region; therein throughput becomes insensitive to the cooperation level, see Fig. 9(a). At any fixed transmission probability vector, we come out that the end-to-end delay is strictly decreasing with $f$, see Fig. 9(b). We note here that connection $c$ outperforms other connections in term of throughput. Analyzing the topology, it is clear that connection $c$ has no common segments with other connections. However, it seems that this connection suffers from relatively high delay compared to other connections. This can be explained by high load of forwarding queues in path $c$, in particular forwarding queue of relay 10, see Fig. 10. Next, we depict the distribution of the average delay in Fig. 11(a), when cooperation level is set to 0.7. A similar behavior is observed at $f = 0.99$, but the curve is shifted to the left where probability to have small or average delay becomes greater. The delay distribution is as narrow as the cooperation level increases. This imply that the average e2e delay decreases with $f$ whereas the e2e throughput may remain constant. Furthermore, we see clearly that the e2e dropping probability is strictly decreasing with the forwarding probability of intermediate relays, see Fig. 10(a).

5.3. Impact of transmissions limit $K$

Another important factor that impacts e2e performances is the maximum number of transmissions per packet. It is clear that the waiting time does not depend on $K$ whereas the service time depends strongly on it. With $K = \infty$, the throughput is maximized $\text{thp}_i = \frac{(1 - \eta_i)}{c}$, see Eq. (2), corresponding to a huge delay (may go to infinity due to long service time caused by successive collisions, in particular when neighbor nodes are very aggressive). When $K = 1$, a minimum average throughput is obtained.
\[ d = \frac{1}{C \rho_s} P_s, \]
\[ d = \left\lfloor s P_k, s \right\rfloor, \]
where the delay is minimized.

Fig. 11(b) and Fig. 12(a) show the distribution of delay with \( f = 0.7 \) for \( K = 1 \) and \( K = 4 \), respectively. One note that the distribution becomes larger when increasing \( K \).

5.4. Static transmission vs. dynamic transmissions

Consider a step \( K_s = 2 \), i.e., \( K_{s,d} = K_{s,d} + 2 \), where node \( j \) is just before node \( i \) in route \( R_{s,d} \). We note that performances are improved since this new scheme gives more chances to packets.
arrived near the final destination, see Fig. 12 (b). Indeed, the delay distribution of dynamic case is more narrow than static retransmissions limit under same value of parameters. In contrast, a huge delay may be observed at intermediate nodes and the use of reset mechanism becomes crucial. Using this new routing, we achieve a better average delay (resp. throughput) for each connection without changing the average throughput (resp. delay). In extreme cases, a reset technique is introduced to reduce congestion and

![Diagram](image URL)
optimize the e2e performances. This scheme seems to be very interesting for delay-sensitive traffic.

5.5. A throughput-delay tradeoff

From proposed cross-layer point of view, see Fig. 2, the delay at an intermediate node can be written (with some abuse of notation and neglecting the transient effect of other parameters) as delay\(=\) waiting-time\(+\) service-time\(=\) delay\(=\) waiting-time\(+\) service-time\(=\) delay. Using a dynamic transmission scheme and based on Fig. 9(a) and (b), one can find an appropriate tradeoff between the throughput and the delay, so that the average delay will be less than some threshold while keeping the average throughput almost constant. This way, making the system running in such a region improves considerably the end-to-end reliability and makes the system able to support several classes of services with different QoS requirements, in particular real-time traffic. Another way to get an appropriate tradeoff between throughput and delay is to fully use the information of our cross-layer model. For instance, by exploiting the instantaneous length of the forwarding queue, a node may efficiently adjust its cooperation level as well as its maximum number of transmissions per packet.

We depict in Fig. 13(a) and (b), the complementary cumulative distribution function of e2e delay for different values of \(K\). As stated before, the e2e delay is minimized for small values of \(K\). This is not efficient since it results in low throughput and a loss probability close to 1 when the path length becomes large. Further, we compare the static retransmission and the dynamic retransmission schemes in terms of delay CCDF, see Figs. 13 and 14. On one hand, we note that the e2e delay of dynamic retransmission scheme is slightly higher compared to the static case. On the other hand, the e2e goodput is improved which confirms the interest of defining a throughput-delay tradeoff.

5.6. Delay control for real-time media streaming

We depict the variation of average goodput with respect to transmission probability for all established connections. We consider a service requiring a delay threshold value \(\Delta_{s,d} \approx 10\) ms (100 time slots). This means that a packet arrives after 100 slots is dropped and not played out. The goodput turns to decrease and vanishes when nodes become very aggressive (transmit at probability close to 1). This situation is similar to the well-known prisoners dilemma in game theory where cooperation between players is crucial. This control mechanism causes packets drop and therefore the goodput is deteriorated (Fig. 15(a)).

Next, we depict in Fig. 15(b) the dropping probability for connections \(a, b, c, d\) and \(e\) as a function of transmission probability.
It represents the amount of packets lost due to delay timeout. When we fix the forwarding probability at \( f_i = 0.8 \) and vary the transmission probability, we note a clear correlation with the corresponding e2e delay. Indeed, when average delay is huge, the dropping probability tends to increase and vice versa. One can note that the dropping probability may increase when the transmission probability goes to 1. This is not due to a huge delay but because of retransmissions expiration. Whereas for fixed transmission probability vector and variable forwarding probability, we note that the admission probability increases with \( f_i \).

6. Conclusion

We have presented a framework to derive the end-to-end delay in ad hoc networks taking into account the parameters related to several layers (cross-layer architecture). We have obtained the distribution of the forwarding queue size and then the average delay based on the probability generating function approach. As an application of our results, we have considered the case of real-time traffic which requires delay/jitter constraints. By using the delay analysis we calculate the admission rate/loss rate of this traffic. We also used the dynamic retransmission scheme first proposed in [3] to improve end-to-end QoS. Preliminary investigations show good match with our experimental illustrations. A part of future guidelines is to address the choice of cooperation level (forwarding probability and retransmission limit) in a game theoretical perspective and analyze the behavior of the selfish nodes. We are also interested in extending our results for wide Mesh networks as well as heterogeneous systems.

Appendix A. The number of new arrivals

We are interested here to compute the number of arrivals during a service time and the number of arrivals during the residual time. Let \( P(d^{s,d} = j), j \geq 0 \) denote the probability that \( j \) number of packets arrived in the queue \( F \) at node \( i \) during a service time of a packet (it may be \( F \) or \( Q \) depends on source \( s \)) on the path from \( s \) to \( d \). Then, for \( K_{s,d} = 1 \), we have

\[
P(d^{s,d} = j) = \left( \frac{l_i}{1-l_i} \right) \sum_{j=1}^{j} \binom{l-1}{j} (1 - \tilde{z}_i)^{j-1} \tilde{P}_i (1 - \tilde{P}_i)^{j-1},
\]

and, for \( K_{s,d} > 1 \), we have

\[
P(d^{s,d} = j) = \left( \frac{l_i}{1-l_i} \right) \sum_{j=1}^{j} \binom{l-1}{j} (1 - \tilde{z}_i)^{j-1} \tilde{P}_i (1 - \tilde{P}_i)^{j-1}.
\]
Then, we multiply both sides for the joint distribution $P_{h_1 h_2 h_3 h_4 h_5} = P(n_{h_1} = h_1, n_{h_2} = h_2, t = h_1, a^d = h_4, \sum_{j \not= h} d_j = h_5)$ and we sum over $h_1, h_2, h_3, h_4, h_5$. Note that on the left side the summations on $h_2, h_3, h_4, h_5$ can be exhausted; whereas on the right side the summation on $h_1$ can be exhausted. Therefore, we can write:

$$
\sum_{h_1 = 0}^{\infty} z^{h_1} P_{h_1} = \sum_{h_2 = 0}^{\infty} \sum_{h_3 = 0}^{\infty} \sum_{h_4 = 0}^{\infty} \sum_{h_5 = 0}^{\infty} z^{h_2 + h_3 + h_4 + h_5} \sum_{j \not= h} \sum_{h_1 = 0}^{\infty} z^{h_1} P_{h_1} h_j = \sum_{h_2 = 0}^{\infty} \sum_{h_3 = 0}^{\infty} \sum_{h_4 = 0}^{\infty} z^{h_2 + h_3 + h_4} P_{h_2 h_3 h_4 h_5}.
$$

Let $P(z)$ denote the PGF at regime of the state probability distribution at the imbedded instants. Then the left side of Eq. (16) can be written as:

$$
P_{h_1} = P(z).
$$

Further, let $A^d(z)$ and $A^q(z)$ denote the PGF of the number of arrivals at regime during the service time of a packet which are from buffers $F$ and $Q$, respectively. Also let $R(z)$ denote the PGF of the number of arrivals at regime during the residual service time of a packet from buffer $Q$. Then the right side of Eq. (16) can be written as:

$$
\sum_{h_2 = 0}^{\infty} \sum_{h_3 = 0}^{\infty} \sum_{h_4 = 0}^{\infty} \sum_{h_5 = 0}^{\infty} z^{h_2 + h_3 + h_4 + h_5} P_{h_2 h_3 h_4 h_5} = \sum_{h_2 = 0}^{\infty} \sum_{h_3 = 0}^{\infty} \sum_{h_4 = 0}^{\infty} \sum_{h_5 = 0}^{\infty} \sum_{j \not= h} ((1-f)A^q(z)) f A^d(z).
$$

If the number of consecutive packets from buffer $Q$ (remember that queue $Q$ is saturated, meanwhile, queue $F$ may be empty) becomes large enough, i.e., $m \to \infty$ then

$$
\sum_{h_2 = 0}^{\infty} \sum_{h_3 = 0}^{\infty} \sum_{h_4 = 0}^{\infty} z^{h_2 + h_3 + h_4} P_{h_2 h_3 h_4 h_5} = \frac{f A^d(z)}{1 - (1-f)A^q(z)}.
$$

Then the right side of Eq. (18) can be written as:

$$
P_{h_1} = \frac{P(z) - P_0}{z} \frac{f A^d(z)}{1 - (1-f)A^q(z)}.
$$

After some algebraic manipulations between (17) and (21), the proof follows.

### Appendix C. Computations of $P_0$, $P(1)$ and $P(1)$

The first order derivative of Eq. (7) at $z = 1$ allows to calculate easily the probability that queue $F$ is empty, i.e.,

$$
P_0 = \frac{f - (1-f)A^q(1) - f A^d(1)}{f(1+R(1))}.
$$

The second order derivative of Eq. (7) at $z = 1$ yields

$$
P(1) = \frac{(1-f)[2A^q(1) + A^q(1) + fA^d(1) + P_f(2 + R(1))A^d(1) + 2R(1) + R^2(1)]}{2[f - (1-f)A^q(1) - f A^d(1)]}.
$$
and, the third order derivative of Eq. (7) at \( z = 1 \) yields

\[
P''(1) = \frac{(1-f)3A''(1)+A''(1)+3(1-f)(2A''(1)+A''(1))}{3f(1-f)A''(1)-fA'(1)}
\]

(24)

where \( \phi'(1) \), \( \phi''(1) \) and \( \phi'''(1) \), respectively, represent the first, second and third order derivatives of any probability generating function \( \phi(z) \) at point \( z = 1 \).

References


