Rubrique

Investigating end-to-end delay measurement in IEEE 802.11 ad hoc networks

SARR Cheikh

Département Mathématiques Informatique
UFR Sciences et Technologies
Université de Thiès
Sénégal
csarr1979@hotmail.com

RÉSUMÉ. Dans ce papier, nous proposons un modèle analytique afin d'évaluer le delai moyen de bout en bout dans les réseaux ad hoc multi-sauts. Notre modèle fournit une expression de ce délai en fonction du temps de service. Ce modèle se base sur une file d'attente M/M/1/K à partir de laquelle nous dérivons les expression du temps moyen de service grâce à la théorie des files d'attente. Cette estimation est ensuite combinée à un protocole appelé DEAN capable de fournir des garanties en terme de délai à des applications QoS. A travers des simulations, nous comparons les performances de notre protocole avec d'autres approches telles que le protocole DDA

ABSTRACT. In this paper, we present a new analytic model for evaluating average end-to-end delay in IEEE 802.11 multihop wireless networks. Our model gives closed expressions for the end-to-end delay in function of arrivals and service time patterns. Each node is modeled as a M/M/1/K queue from which we can derive expressions for service time via queueing theory. By combining this delay evaluation with different admission controls, we design a protocol called DEAN (Delay Estimation in Ad hoc Networks). DEAN is able to provide delay guarantees for QoS applications in function of the application level requirements. Through extensive simulations, we compare performance evaluation of DEAN with other approaches like, for instance, DDA.

MOTS-CLÉS : Réseaux ad hoc, IEEE 802.11, Estimation du délai, Bande passante

KEYWORDS : Ad hoc Networks, IEEE 802.11, Delay estimation, available bandwidth

1. Introduction

Using Voice-over-IP services in wireless mesh networks is becoming a reality [1]. Such VoIP applications are delay sensitive and require some delay guarantees. Offering and maintaining delay guarantees is a difficult task in wireless mesh networks and more generally in multihop wireless networks.

Recently, the works on quality of service (QoS) guarantees in ad hoc networks have attracted more and more attention. Most of these works assume that the underlying technology used in these networks is the IEEE 802.11 technology [2]. This technology is widely available, not so expensive and provides a distributed radio medium access that can be easily used in an ad hoc network. On the other hand, the random radio medium access provided by IEEE 802.11 offers few control on the emissions and makes the radio medium sharing difficult in a multihop context [3]. Many works offer quality of service to IEEE 802.11-based ad hoc networks by providing either throughput guarantees or delay guarantees or both. Among these studies, most of them have focused on throughput guarantees (like for instance [4, 5, 6]) and few of them have proposed solutions for delay guarantees.

If solutions for providing throughput guarantees are still not perfect, they are nonetheless more and more efficient. Ensuring the delay seems to be a more challenging task. As mentioned in [8], it is very difficult to predict the expected delay due to the strong dependency between the flows in a wireless multihop setting.

In this article, we propose a new protocol to achieve delay guarantees in wireless multihop networks. With this study, we show that it is possible to design an efficient measurement-based admission control protocol for the delay parameter. The proposed protocol, called DEAN (Delay Estimation in Ad Hoc Networks), is based on a *a priori* estimation of average end-to-end delay. This estimation is derived from a simple model of IEEE 802.11 nodes and from an accurate evaluation of each link's collision probability. By combining this estimation with accurate admission controls, the estimated delay is guaranteed after a new flow starts. Such guarantees depend mainly on a strong correlation between the estimated delay and the available bandwidth as an efficient estimation of available bandwidth. This latter is estimated with the protocol ABE (Available Bandwidth Estimation) that provides an accurate evaluation [6]. Finally, extensive simulations show that our protocol DEAN is very efficient to provide delay guarantees.

The remainder of this paper is organised as follows : Section 2 describes our end-to-end delay estimation mechanism. In Section 3 simulation results are presented and Section 4 concludes our work.

2. Mean delay estimation

Delay indicates the time to send a packet from a source to a destination node. Contrary to bandwidth, delay is an additive metric. Thus, the delay along a path is equal to the sum of the delays on one-hop links of this path.

With the use of IEEE 802.11, the mean packet delay on a specific one-hop link, denoted by D, can be divided into three parts :

– The *mean queueing delay* which represents the interval between the time the packet enters in the queue of the link's emitter and the time that the packet becomes the head of line packet in this node's queue. We denote it by D_q.

– The *mean contention delay* is the interval between the time the packet arrives at the head of line and the time the packet is sent to the physical medium. We denote it by D_c. This interval reflects the fact that a node may contend to access to the channel because of other transmissions in its carrier sensing area.

– The *mean transmission delay* is the time to transmit the whole packet including possible retransmissions in case of collisions. We denote it by D_t.

Therefore we have the relation :

$$\boxed{D = D_q + D_c + D_t} \tag{1}$$

In the remainder of this section, we made some assumptions in order to simplify the analysis and to give analytical expressions for D_q, D_c and D_t.

2.1. Assumptions and general idea

We model, as described in [7], a IEEE 802.11 node as a discrete time M/M/1/K queue. The properties of this queue are :

– The packet arrival follows an exponential law of parameter λ. The value of λ represents two parameters : the source traffic initiated by the tagged node and the routed traffic from other nodes.

– The service rate follows also an exponential law of parameter μ.

– The size of the queue is limited by the value K. When a new packet arrives and if there are already K packets in the queue, then this one is dropped.

– The queue is a classical FIFO (First In First Out).

The parameter λ represents the number of packets arriving in the queue per second which depends on the application throughput (if such an application exists on the node) and the traffic routed by this node. The service time μ represents the number of packets that the MAC Layer can offer.

Our initial goal is to provide delay guarantees to delay sensitive flows. To this end, we need to estimate the mean delay that the packets of such a flow will achieve before transmitting this flow. Therefore, we need to estimate the service rate that can be offered to this flow on each node passed through by this flow. It is also important to remind that the acceptance of a new flow may impact the delays of existing flows. Our goal is then also to minimize such an impact in order to achieve the delay guarantees of existing delay sensitive flows. We could define the **available service rate** of a node as *the service rate that can be offered to a new flow without increasing the delay of any ongoing flow in the network*.

In order to limit the impact on the mean delay of existing flows, a congestion control must be realized. Therefore, the service rate that can be offered by a node to a new flow is directly correlated to the residual bandwidth seen by this node and already computed in [6]. This value of μ captures the effect that after the queuing process, a packet which arrives at the head of line of the MAC layer should wait until the channel is free in order to gain the access. More precisely, we model μ_{res}, the service rate that can be offered to a new flow, as the available bandwidth computed by the node rescaled in packets per second.

2.2. Estimating the mean queueing delay D_q

In this section, we estimate D_q. When $\mu > \lambda$, the service rate of a node is higher than the arriving process and there will be no accumulation in the queue. This situation involves a very low queueing delay which can be easily obtained through simulations.

When $\mu \leq \lambda$. Let's denote by $p(n)$ the probability to have n packets in the queue ($n \leq K$). A packet arrives with rate λ and exits with rate μ. So :

$$p(n) = \frac{\lambda}{\mu} \times p(n-1) = \left(\frac{\lambda}{\mu}\right)^n \times p(0)$$

$$\text{Using } \rho = \frac{\lambda}{\mu} \Longrightarrow p(n) = \rho^n \times p(0)$$

The sum of the probabilities being equal to 1 : $\sum_{n=0}^{K} p(n) = 1$

We can simply express $p(n)$ in function of ρ and K :

$$p(n) = \begin{cases} \rho^n \dfrac{1-\rho}{1-\rho^{K+1}} & \text{if } \rho \neq 1 \\[2ex] \dfrac{1}{K+1} & \text{if } \rho = 1 \end{cases}$$

The mean number of packets Q in the queue is therefore : $Q = \sum_{n=0}^{K} n \times p(n)$

Using queueing theory and according to Little's law, the parameter D_q is equal to the *mean waiting time* :

$$D_q = \frac{Q}{\lambda}$$

So the final expression is :

$$D_q = \begin{cases} \dfrac{\rho}{1-\rho} \dfrac{1-(K+1)\rho^K + K\rho^{K+1}}{1-\rho^K} \dfrac{1}{\lambda} & \text{if } \rho \neq 1 \\[2ex] \dfrac{K}{2\lambda} & \text{if } \rho = 1 \end{cases}$$

2.3. Estimating the mean contention delay D_c

We define the mean contention delay D_c of a packet to be the time the node senses the channel busy due to transmission of other nodes within the carrier sensing area as illustrated in Figure 1. The computation of this value is not trivial, therefore we performed it by simulations.

Before a successful transmissions by a specific node, other nodes may successfully transmit a number of packets or may be involved in a number of collision, each of which add to the channel access time of the specific node. It is also important to note that transmission attempts by the specific node which result in collisions and retransmissions also increase the mean contention delay D_c.

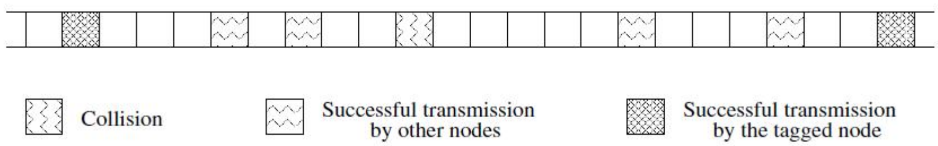


Figure 1 – Busy time

In the following, we use the backoff slot length δ as the time unit. Note that in real networks the packet arrival process is a continuous time process and the arrival process may occur anywhere in the slot. However, as δ is in order of $20\mu s$, the error introduced by the discretization is small.

The value of D_c is correlated to two parameters :

– The available bandwidth E_{avb} which reflects the medium occupancy around the tagged node

– The probability of collision which reflects the number of retransmissions before a successful one occurs

In our simulations, we find a specific function F depending of these two parameters such as :

$$D_c = F(E_{avb}, p)$$

2.4. Estimating the mean transmission delay D_t

The mean transmission delay is the time to transmit the whole packet. In IEEE 802.11 DCF, when this operation is successful, a positive acknowledgment is sent back to the emitter. However, there is a chance, even for a single frame that when a packet is emitted, the medium is not idle at the receiver's side, provoking a collision. These collisions involve retransmissions of the same packet and increases the contention window size, all these phenomena resulting in an increase of the mean transmission delay. Moreover, when the channel is occupied by other transmissions, the backoff counter is paused and the node has to wait until the channel becomes free again, all these features increase the transmission delay.

2.4.1. Modeling the exponential backoff mechanism

Let us consider that an arbitrary wireless link suffers from collisions with a probability p (evaluated with the method given in [6]). For every frame, the transmission is successful at the first attempt with probability $(1-p)$. It succeeds at the second attempt with probability $p \cdot (1-p)$. After C unsuccessful attempts, C depending on the frame size, the IEEE 802.11 standard specifies that the frame should be dropped.

If we denote by X the random variable representing the number of attempts performed for the correct transmission of a given frame, we have the following probabilities :

$$P(X = k) = \begin{cases} p^k \cdot (1-p) & \text{if } k < C \\ p^k & \text{if } k = C \\ 0 & \text{if } k > C \end{cases}$$

The expected number of retransmissions n for a given frame can be expressed as follows :

$$n = \sum_{k=0}^{+\infty} k \cdot P(X = k) = \sum_{k=1}^{C} k \cdot P(X = k) = \sum_{k=1}^{C-1} k.p^k.(1-p) + Cp^C$$

$$n = \frac{p.(1-p^C)}{1-p}$$

Now, we need to compute the expected backoff that impacts the delay transmission. First, let us consider that there is no collision. Then the backoff is drawn according to a uniform law in the interval $[0; CW_{min} - 1]$, CW_{min} being determined by the MAC protocol specification. On a large observation window, the backoff can be approximated by its average value $\frac{CW_{min}-1}{2}$.

When collisions happen, the exponential backoff mechanism is triggered. After each unsuccessful transmission, the contention window size is doubled up to a maximum value denoted by CW_{max}. In this situation, the average backoff value increases way above $\frac{CW_{min}-1}{2}$ and it is necessary to model the time consumed by the exponential backoff process. At the end of k unsuccessful attempts, $CW(k)$ the mean backoff at stage k is given by :

$$CW(k) = \frac{\min(CW_{max}; 2^k \cdot CW_{min}) - 1}{2}$$

Using this expression, we can evaluate the number of backoff slots decremented on average for a single frame :

$$\overline{backoff} = \sum_{k=0}^{C} P(X = k).CW(k)$$

$$\overline{backoff} = \sum_{k=0}^{C} P(X = k) \cdot \left(\frac{\min(CW_{max}; 2^k \cdot CW_{min}) - 1}{2} \right)$$

To simplify the expression, let us define M so that $CW_{max} = 2^M \cdot CW_{min}$ with $M \leq C$. If we further replace $P(X = k)$ by its expression, we get :

$$\overline{backoff} = \frac{CW_{min} \cdot (1-p) \cdot (1-(2 \cdot p)^{M+1})}{2 \cdot (1 - 2 \cdot p)} + \frac{1}{2} \cdot \left(p^{M+1} - 1 + (CW_{max} - 1) \cdot (p^{M+1} - p^{C-M-1} + p^C)\right)$$

2.4.2. Mean transmission delay computation

The different points mentioned above can be combined to estimate the mean transmission delay on a wireless link, i.e. between an emitter and a receiver. To summarize, the mean transmission delay between two neighbor nodes can be estimated by the following formula :

$$D_t = \overline{backoff}.\delta + n.T_c + T_m + T_{busy} \qquad (2)$$

where T_m is the time to successfully transmit a whole packet of m bytes with IEEE 802.11, T_c is the collision duration, n is the mean number of retransmissions depending on collision probability, $\overline{backoff}$ is the expected number of backoff slots, δ is the duration of a slot and T_{busy} is the duration the node senses the channel as busy due to other transmissions. It is important to notice that the backoff is decremented as long as the channel is sensed idle, otherwise the backoff scheme is paused until the channel becomes free again. This supplementary duration is captured by T_{busy}.

To sum up, the mean delay D of a one-hop link consists of :

– the mean queuing delay experienced by a packet, the time the packet enters in the queue of the link's emitter and the time it becomes the head of line packet in this node's queue denoted by D_q.

– the mean contention delay D_c of a packet which captures the channel busy time due to transmission of other nodes within the carrier sensing area.

– the mean delay experienced by a packet during the transmission (D_t). It also includes the potential retransmissions induced by collisions.

3. Estimation

In this section, we evaluate and compare, by simulation, the performances of our solution with other approaches. We use the network simulator 3 (NS-2.34) and the IEEE 802.11 implementation provided with the simulator.

The parameters used for scenarii are presented on Table 1. Unless specified, the queue of each node (K) can contain at mot 100 packets of 1000 bytes. We have implemented a protocol version of our delay estimation scheme called DEAN (Dean Estimation in Ad hoc Networks). We compare the performances of DEAN with DDA [8] another end-to-end delay estimation. We chose DDA because it is one of the most recent solutions that outperforms several measurement-based admission control protools like SWAN for instance. The comparison between DDA and DEAN is focused on the delay estimation and the delay guarantee since these two approaches are based on the same bandwidth admission control and the same available bandwidth estimation. We also compare DEAN with AODV as a baseline for comparison.

Parameters	**Values**
HELLO interval	1 s
Packet size	1000 bytes
Radio medium capacity	2 or 11 Mb/s
Communication range	250 m
Carrier sensing range	550 m
Grid size	1000×1000 m
C (Number of retransmissions)	6

Tableau 1 – General parameters for simulations

AAC overestimates the available bandwidth, it forwards more route requests during the route discovery phase.

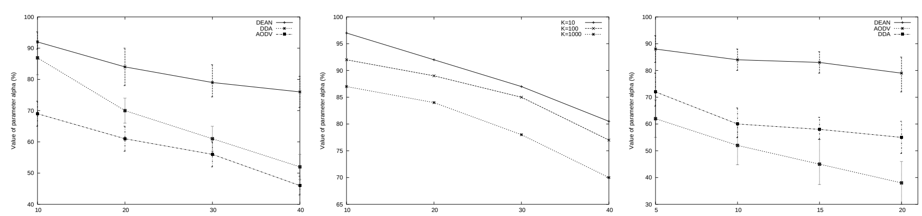
3.1. Accuracy of the estimation

Let us now investigate more scenarii. To evaluate the accuracy of the end-to-end delay estimation in several scenarii, we use the following metric :

$$\alpha = \frac{n}{N}$$

where n is the number of packets that don't experience delay violation and N is the total number of packets correctly received. We consider that a packet, for which the end-to-end delay may be higher than 5% of the delay requirements when it is emitted, experiences a delay violation. This metric reflects the fact that a falsely admitted flow either degrades delay of close flows or is not able to achieve its own desired end-to-end delay.

We measure the value of α by simulation on networks composed of 10 to 40 nodes. The radio medium capacity is of 11 Mb/s. Each simulation lasts 100 seconds and three random sources try to establish delay-sensitive CBR connections towards three random destinations. These three CBR connections are competing with 5 best effort CBR flows, started at the beginning of the simulation. The throughput of each connection is uniformly drawn between 100 kb/s and 300 kb/s and the end-to-end delay requirements is fixed to 50 ms for all delay-sensitive flows. In the following, we evaluate the values for α in function of three parameters : the used protocol ; the queue size and the number of delay-sensitive flows [1]. All the results presented below are the average of 30 simulations with different random seed.



(a) Value of α in function of the used protocol

(b) Value of α in function of queue size for DEAN

(c) Value of α in function of the number of delay-sensitive flows

Figure 2 – Values of α

Influence of the used protocol : Figures 2(a) gives the values of α in function of the number of nodes present in the network. When the network is not too loaded, α is relatively high for both DEAN and DDA (respectively about 92 and 86% for 10 nodes). By increasing the density of the network, we see that DEAN is more accurate on delay guarantees than DDA. For example, when there are 30 nodes, the value of α is about 79% for DEAN and 61% for DDA.

DDA proposes a stateless approach, meaning that no estimation of residual resources are carried out. Each node of DDA regulates dynamically the contention window size in function of its local perception of close transmissions. This regulation is also based on some asumptions. The obtained results let us think that some of these asumptions are maybe too strong, especially when the density of the network increases. Indeed, when the desity becomes higher, more collisions may appear involving packets retransmissions. Moreover, the delay on a path may very probably not evenly broken down among the

1. In this last case, the number of delay sensitive flows will not be three.

hops of this path. Therefore, DDA handles less efficiently these collisions in the delay evaluation than DEAN. In all cases, the performances of DDA and DEAN are better than AODV for which no admission control and regulation are done.

Influence of queue size : Figure 2(b) shows the values of α in function of the queue size of all nodes in the network. Increasing the queue size decreases the value of α. For example, for a density of 10 nodes, α is equal to 97% for a queue size of K =10 and 87% for a queue size of K =1000. It can be explained by the fact, with small queues, the queueing delays will be smaller as the variance in delay. Therefore, the delays are more stable when nodes have small queues. Note that, even with small queues, the number of dropped packets is very low. For instance, this number is less that 4% when K is equal to 10.

Influence of the number of delay-sensitive flows : For this evaluation, the number of nodes is fixed to 20. We increase gradually the total number of delay-sensitive flows from 5 to 20 and compute the values of α. As shown on Figure 2(c), for DDA, α decreases when the number of delay-sensitive flows increases. Actually, DDA only performs an admission control on bandwidth but not on delay and the regulation performed by DDA becomes more and more difficult when the total amount of traffic increases in the network. Therefore, these results show that an admission control on delay that eliminates potential routes is of some interest when the network becomes loaded. For example, when there are 20 delay-sensitive flows, α is equal to 85% for DEAN and 55% for DDA. This part illustrates the scalability of DEAN.

4. Conclusion and future works

In this article, we propose a technique to evaluate *a priori* the delays that flows will achieve in a wireless multihop network. By combining this technique to admission controls based on available bandwidth and on delay, we design a protocol DEAN that can guarantee delays to delay sensitive flows with a high level of accuracy. Simulations compare the performance of AODV, DDA and DEAN and show that, under different configurations, DEAN is able to provide strict guarantees.

In the future, we plan to work towards three main directions. First, in the presented study, we haven't considered the co-existence of delay sensitive traffic and best effort traffic. If, in simulations, some best effort flows were present, they were transmitted before the starting of delay sensitive flows. Our solution is based on an admission control that ensures that congestion will not appear. It is therefore important to take care that best effort flows started while delay sensitive flows are transmitting will not impact the delays of these latter. To be efficient, it will also important to let the possibility to use the whole bandwidth to best effort traffics if no delay sensitive flows are transmitted. In [**?**], we have already designed a mechanism to ensure the co-existence between best effort flows and bandwith sensitive flows. We plan to extend this approach to delay sensitive flows. Second, we plan to merge the different QoS solutions we have designed in order to provide a unique solution that will handle best effort traffic, bandwidth sensitive traffic and delay sensitive traffic. Our goal is to provide accurate guarantees while using the bandwidth efficiently. Finally, we plan to test our solution with real delay sensitive applications, like for instance, VoIP applications. Other mechanisms like VoIP packet aggregation and header compression, as mentionned in [1], should also be considered.

5. Bibliographie

[1] S. GANGULY, V. NAVDA, K. KIM, A. KASHYAP, D. NICULESCU, R. IZMAILOV, S. HONG AND S. R. DAS, « Performance Optimizations for Deploying VoIP Services in Mesh Networks », *IEEE Journal on Selected Areas in Communications*, vol. 24, n° 11, Nov 2006

[2] IEEE STANDARD FOR INFORMATION TECHNOLOGY - TELECOMMUNICATIONS AND INFORMATION EXCHANGE BETWEEN SYSTEMS, « Local and Metropolitan Area Network - Specific Requirements - Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications », *The Institute of Electrical and Electronics Engineers*, 1997

[3] C. CHAUDET, D. DHOUTAUT AND I. GUÉRIN LASSOUS, « Performance Issues with IEEE 802.11 in Ad Hoc Networking », *IEEE Communication Magazine*, vol. 43, n° 7, Jul 2005

[4] Y. YANG AND R. KRAVETS, « Contention Aware Admission Control for Ad Hoc Networks », *IEEE Transactions on Mobile Computing*, vol. 4, n° 4, pages 363-377, Jul 2005

[5] C. CHAUDET AND I. GUÉRIN LASSOUS, « BRuIT - Bandwidth Reservation under InTerferences influence », *In Proceedings of European Wireless*, Feb 2002, Florence Italy

[6] C. SARR, C. CHAUDET, G. CHELIUS AND I. GUÉRIN-LASSOUS, « Bandwidth Estimation for IEEE 802.11-Based Ad Hoc Networks », *IEEE Transactions on Mobile Computing*, vol. 7, n° 10, pages 1228-1241, 2008

[7] R.S. WONG, S. SHANKAR AND N. VAN DER SCHAAR « Integrated application MAC modeling for cross-layer optimized wireless video », *IEEE International Conference on Communications*, vol. 2, n° 10, pages 1271-1275, 2005

[8] Y. YANG AND R. KRAVETS « Achieving Delay Guarantee in Ad Hoc Networks by Adapting IEEE 802.11 Contention Windows », *In Proceedings of IEEE Infocom*, October 2006