Link-Layer Fragmentation and Retransmission Impact on TCP Performance in 802.11-based Networks

Fethi Filali
Institut Eurcom
2229 Route des Crtes, BP-193,
06904 Sophia-Antipolis, France
filali@eurecom.fr

Contact details of the first author:

Fethi Filali
Institut Eurcom
2229 Route des Crtes, BP-193,
06904 Sophia-Antipolis, France

Tel.: +33 4 93 00 29 16
Fax.: +33 4 93 00 26 27
Email: filali@eurecom.fr

Abstact

Our objective in this paper is to examine the behavior of TCP in 802.11-based networks. The focus of this work is mainly on the effect of the fragmentation and frames retransmission handled at the MAC layer on the TCP end-to-end performance. The packet loss rate and the end-to-end delay of TCP connections are analyzed and evaluated. Some preliminaries relationships between the estimated TCP retransmission time-out and the maximum number of retransmissions at the MAC level are derived. The obtained results show how the number of fragments and the maximum number of retransmissions should be tuned in order to reach a desired values of both TCP delay and loss rate. The main contribution of this work is the development of an analytical framework that could be used as a basis for the development of a cross-layer cooperative scheme between the retransmission mechanisms TCP and MAC protocols.

Keywords

TCP performance, 802.11-based networks., link-layer fragmentation, frames retransmissions, cross-layer design.

Link-Layer Fragmentation and Retransmission Impact on TCP Performance in 802.11-based Networks

Fethi Filali
Institut Eurcom
2229 Route des Crtes, BP-193,
06904 Sophia-Antipolis, France
fi lali@eurecom.fr

***Abstract*— Our objective in this paper is to examine the behavior of TCP in 802.11-based networks. The focus of this work is mainly on the effect of the fragmentation and frames retransmission handled at the MAC layer on the TCP end-to-end performance. The packet loss rate and the end-to-end delay of TCP connections are analyzed and evaluated. Some preliminaries relationships between the estimated TCP retransmission time-out and the maximum number of retransmissions at the MAC level are derived. The obtained results show how the number of fragments and the maximum number of retransmissions should be tuned in order to reach a desired values of both TCP delay and loss rate. The main contribution of this work is the development of an analytical framework that could be used as a basis for the development of a cross-layer cooperative scheme between the retransmission mechanisms TCP and MAC protocols.**

Keywords: TCP performance, 802.11-based networks., link-layer fragmentation, frames retransmissions, cross-layer design.

I. Introduction

TCP performance in wireless networks has been the subject of extensive research work. Most of them have focused on node failures, mobility, and multi-hop effects on TCP performance. [4] reviews the first proposals to enhance the TCP throughput and fairness in wireless links and [1] gives an up-to-date survey of protocols that have been developed for TCP optimization over mobile ad hoc networks.

In this work, we focus on how TCP interacts with the 802.11 MAC ARQ retransmission schemes. The 802.11 MAC's ARQ (Automatic Repeat reQuest) is a stop-and-wait ARQ with positive acknowledgments after each packet. The detection of lost frames is achieved by an ACK timeout. The standard uses two different retry limits for short and long frames specifying how many times a packet should be retransmitted before it is dropped, can be set for each packet. On the other hand, after the expiration of a time out, a TCP source resends the TCP segment for which it is waiting of an acknowledgment from the destination.

The goal here is to derive the relationship between TCP and MAC retransmission scheme. The MAC fragmentation is taken into account in computing the end-to-end delay and the packet loss rate. To the best of our knowledge, despite the large amount of research works that studied the behavior of TCP in wireless networks, no one of them has examined the effect of MAC retransmissions and fragmentation on TCP performance.

The obtained expressions are able to lead to the development of cross-layer scheme aiming to include a cooperation between TCP and MAC in order to reduce the both parameters (delay and loss rate). Indeed, the cross-layer approach is an emerging design paradigm specially in wireless networks where redundant and similar mechanisms may occur at different layers. We believe that avoiding this redundancy by allowing layers cooperation is a good promising challenge that leads to an enhancement of the network and traffic performance without adding complexity. This was the final target of this work.

The remainder of this paper is organized as follows: we provide a description of DCF and TCP operations in Section II. TCP analysis in 802.11-based networks for the packet loss rate is given in Section III. The end-to-end TCP delay is analyzed in Section IV. Section V concludes this paper and outlines our future works.

II. DCF and TCP Operations

A. DCF operations

A 802.11 datalink layer is divided in two sublayers: Logical Link Control (LLC) and Media Access Control (MAC). The LLC sublayer is the same in 802.11 and other 802 LANs and can easily be plugged in into a wired LAN, but 802.11 defines a different MAC protocol. For Ethernet LANs, the CSMA/CD protocol regulates the access of the stations. In a WLAN collision detection is not possible. The 802.11 standard defines the protocol and compatible interconnection of data communication equipment via the air, radio or infrared, in a local area network (LAN) using the CSMA/CA medium sharing mechanism. This basic access method for 802.11 is called Distributed Coordination Function (DCF) and its mandatory for all stations.

CSMA/CA needs each station to listen to other users. If the channel is idle the station is allowed to transmit. If it is busy, each station waits until transmission stops, then enters into a random back off procedure. This prevents multiple stations from owning the medium immediately after completion of the preceding transmission. Packet reception in DCF requires acknowledgments (ACK). The period between completion of packet transmission and start of the ACK frame is one Short Inter Frame Space (SIFS). ACK frames have a higher priority than other traffic. Fast acknowledgment is one of the features of the 802.11 standard, because it requires ACKs to be handled at the MAC sublayer. Transmissions other than ACKs must wait at least one DCF inter frame space (DIFS) before transmitting data. If a transmitter senses a busy medium, it determines a random back-off period by setting an internal timer to an integer number of slot times. Upon expiration of a DIFS, the timer begins to decrement. If the timer reaches zero, the station may begin transmission. If the channel is seized by another station before the timer reaches zero, the timer setting is retained at the decremented value for subsequent transmission. When, after a network error, an ACK is not received, the source station contends again for the channel to transmit the unacknowledged packet and, in case of further error, retries until a maximum retry limit is reached.

B. Packets retransmission in TCP

TCP is a reliable transport protocol that performs efficiently in wire-line networks. It uses a Go-back-N protocol and a timer-based retransmission mechanism. The timer period (the timeout interval) is calculated based on the estimated round-trip delay. Packets whose acknowledgments are not received before the timer expires are retransmitted. In the presence of frequent retransmissions, TCP assumes that there is a congestion and invokes its congestion control algorithm. The algorithm reduces the transmission (also called congestion) window size. As the window size is reduced, the transmission rate is also reduced. This window size adjustment technique prevents the source from overwhelming the network with an excessive number of packets.

In the presence of high bit error rates in wireless links, TCP reacts the same way as in a wired link: it reduces the window size before packet retransmission. This adjustment results in an unnecessary reduction of the bandwidth utilization causing significant performance degradation (poor throughput and long delays). The bandwidth of the wired-link segments is especially under-utilized because the high error rates occur only on the wireless links. Nevertheless, the window reduction affects all transmission links.

In this work, we focus on TCP New Reno which is an extension to TCP Reno which avoids multiple decreasing of the congestion window size when several segments from the same window have been lost [7]. Nowadays, New Reno is the leading Internet congestion control protocol [10].

III. Analysis of the packet loss rate

In 802.11, a MAC service data unit (MSDU) could be partitioned into a sequence of smaller MAC protocol data unit (MPDUs). Fragmentation creates MPDUs smaller than the original MSDU length to increase reliability, by increasing the probability of successful transmission of the MSDU in cases where channel characteristics limit reception reliability for longer frames. Fragmentation is accomplished at each immediate transmitter. The process of recombining MPDUs into a single MSDU is defined as defragmentation. Defragmentation is accomplished at each immediate recipient.

Only MPDUs with a unicast receiver address shall be fragmented. Broadcast/multicast frames shall not be fragmented even if their length exceeds `aFragmentationThreshold`.

When a directed MSDU is received from the LLC with a length greater than `aFragmentationThreshold`, the MSDU shall be fragmented. The MSDU is divided into MPDUs. Each fragment is a frame no larger than `aFragmentationThreshold`. It is possible that any fragment may be a frame smaller than `aFragmentationThreshold`. An illustration of fragmentation is shown in Figure 1.

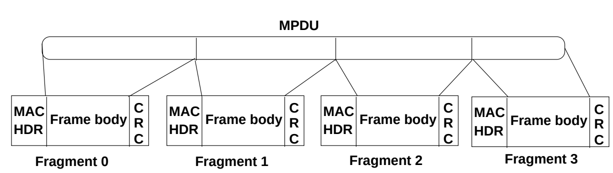

Fig. 1. MSDU partitioned into several MPDUs

The MPDUs resulting from the fragmentation of an MSDU are sent as independent transmissions, each of which is separately acknowledged. This permits transmission retries to occur per fragment, rather than per MSDU. **Unless interrupted due to medium occupancy limitations for a given PHY, the fragments of a single MSDU are sent as a burst during the CP, using a single invocation of the DCF medium access procedure. The fragments of a single MSDU are sent during a CFP as individual frames obeying the rules of the PC medium access procedure.**

Now a days with the need to achieve higher and higher rates and the need for to large networks in terms of size as well as the number of users, DSSS becomes the obvious option. That is why IEEE 802.11b, IEEE 802.11g and IEEE 802.11e have all chosen DSSS PHY instead of FHSS PHY. It was this fact that motivated us to choose IEEE 802:11 DSSS PHY, because it seems to be choice of the future.

An MSDU is considered successfully transmitted if all of the MPDUs have been transmitted with success. When the MAC fails to transmit an MPDU after the max number of retransmissions, the MSDU is discarded and remaining MPDUs are not transmitted.

Assume that P is the end-to-end packet loss rate. $P = P_{wd} + (1 - P_{wd}) \cdot P_{wl}$ where P_{wd} and P_{wl} are the packet loss rate in the wired and wireless portion of the path, respectively. Hereafter we provide an analysis of the packet loss rate in the wireless portion.

Let P_l denote the unsuccessfully transmission probability of one attempt. The probability of successfully transmission of a fragment after i attempts is $P_l^{i-1} \cdot (1 - P_l)$ with $i \geq 1$. The probability of successfully transmission of a given fragment is then $\sum_{i=1}^{i=M} P_l^{i-1}(1 - P_l)$ where M is the maximum number of retransmissions.

The 802.11 MAC starts attempting to send the fragment number k if all previous fragments have been transmitted with success. Therefore, the probability to send fragment k with success is

$$P(success) = \left(\sum_{i=1}^{i=M} P_l^{i-1}(1 - P_l) \right)^k \quad (1)$$

with the assumption that the same maximum number of retransmissions is applied for all fragments.

The loss probability of transmitting a TCP packet fragmented at the MAC layer into N fragments is

$$P_{wl} = 1 - \left(\sum_{i=1}^{i=M} P_l^{i-1}(1 - P_l) \right)^N = 1 - (1 - P_l^{M-1})^N \quad (2)$$

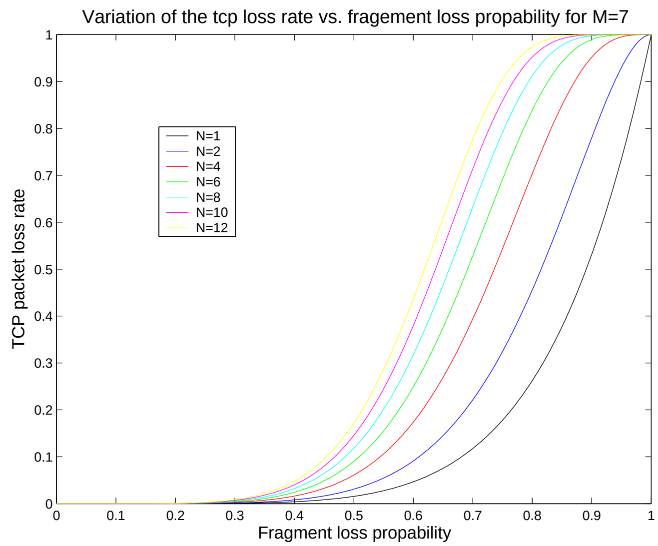

Fig. 2. Variation of the tcp packet loss rate at the MAC layer as a function of the fragment loss rate when the number of the maximum number of transmissions is equal to 7 for different number of fragments.

Figure 2 depicts the variation of the TCP loss rate as a function of the fragment loss probability during each transmission retry for distinct number of fragments and for $M = 7$. As we can easily observe, the fragment loss rate has a small effect on the packet loss rate when it remains less that 0.4 but when it goes up to 0.4, the packet loss rate increases significantly. Moreover, a higher is the number of fragments, higher the packet loss it is because when at least a fragment is lost, the TCP packet is considered lost.

In Figure 3, we plot the P_{wl} vs. the number of retransmissions attempts M, where N is set to 10. When the number of retries is very low, the packet loss rate increases rapidly even for low fragment loss probability. In contrary, when the MAC layer uses a high number of retries, the packet loss rate remains almost not affected for a fragment loss probability low that 0.4.

The loss of a fragment could be due to two events: collision or error. So we can write: $P_l = P_c + (1 - P_c) \cdot P_e$ where P_c and P_e are the packet loss rate due to collision and error, respectively.

According to [16], P_c is the solution of the following equation

$$P_c = 1 - \left(1 - \frac{2(1 - 2P_c)}{1 - P_c - P_c(2P_c)^{\log_2(CW_{max}/CW_{min})}} \cdot \frac{1}{CW_{min}} \right)^{n-1}$$

where n is the number of stations, CW_{min} is the minimum contention window size, and CW_{max} is the maximum contention window size ($CW_{max} = 2^m \cdot CW_{min}$).

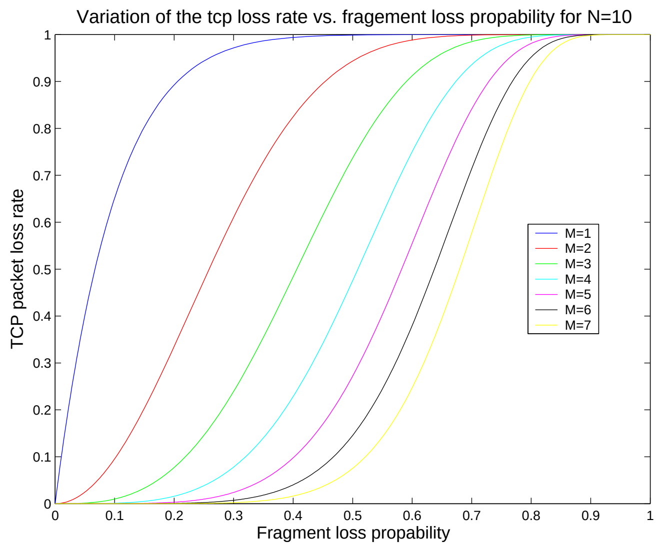

Fig. 3. Variation of the TCP packet loss rate at the MAC layer as a function of the fragment loss rate when the number of fragments is fixed to 10. The maximum number of retransmissions varies between 1 and 7.

Given that IEEE 802.11 does not do any encoding, the fragment error rate and the bit error rate are related by the following simple relation $P_e = 1-(1-BER)^s$, where s is the fragment size and BER is the Bit Error Rate.

For the case of fading channel, we assumed Rayleigh faded channel. This is some sort of worst case scenario as direct line of sight path may exist in some cases resulting in Ricean fading and better performance. For the case of Rayleigh fading the equation relating BER and SINR (Signal to Interference and Noise Ratio) for the case of receiver with just one channel is given by the following equation [13]:

$$BER = \frac{1}{2} \cdot \left(1 - \frac{\mu}{\sqrt{2-\mu^2}}\right) \quad (3)$$

where $\mu = \frac{SINR}{1+SINR}$.

Assuming the same transmission power (P) for all the nodes and consider a transmission from node X_i to node X_j. Let us denote the path loss from node X_i to X_j by L_{ij}. Then the received power at node X_j is given by $R_{ij} = P - L_{ij}$. Suppose that Δ is the set containing all the senders whose transmission overlaps with the above transmission. Then the total interference power is $\Sigma_{k\in\Delta}(P - L_{kj})$. The Average $SINR$ is then obtained as follows:

$$SINR = \frac{R_{ij}}{\Sigma_{k\in\Delta}(P-L_{kj})\alpha_{kj} + N_p} \quad (4)$$

where N_p is the noise power calculated before and α_{kj} is the percentage overlap between the transmission from X_i to X_j and the transmission from k.

L_{ij} is usually computed using the common LAN's path loss model described in [14], which has been experimentally justified for indoor environment to be computed as follows

$$L_{ij} = 40 + 35 \cdot \log(d_{ij}) \quad (5)$$

where d_{ij} is the distance between nodes i and j in meters. Note that this model is valid for distances greater than 7-8m and it assumes a propagation exponent of 3.5.

IV. Analysis of the average end-to-end delay

In this section, we provide an analysis of the TCP end-to-end delay in wired/wireless hybrid networks. The average end-to-end delay T is equal to $T_{wd}+T_{wl}$ where T_{wd} and T_{wl} are the average end-to-end delay in the wired and the wireless portion of the path, respectively.

Assume that all fragments are sent in burst separated by SIFS.

T_{wl} is computed as function of SIFS and transmission delay. To attempt to transmit a fragment a delay $T_{frag}^{attempt}$ is required $T_{frag}^{attempt} = SIFS + T_{tr}^{frag} + SIFS + T_{tr}^{ack}$ where $T_{tr}^{frag} = \frac{\text{data fragment size}}{\text{data transmisison rate}}$ and $T_{tr}^{ack} = \frac{\text{ack frame size}}{\text{ack transmisison rate}}$. Without loss of generality, we assume that the MSDU of TCP packet is divided into several fragments with equal size.

To successfully transmitting a fragment an average number M_{avg} of retransmission attempts is needed. M_{avg} is computed as follows

$$M_{avg} = \sum_{k=1}^{M} k \sum_{i=1}^{i=k} P_l^{i-1}(1-P_l) = \sum_{k=1}^{M} k(1-P_l)^{k-1} \quad (6)$$

The average delay required to successfully transmit a fragment is

$$T_{wl} = T_{rts/cts} + N \cdot M_{avg} \cdot T_{frag}^{attempt} \quad (7)$$

$$T_{wl} = T_{rts} + SIFS + T_{cts} + SIFS + N \cdot M_{avg} \cdot T_{frag}^{attempt}$$

where N is the number of fragments.

The round trip time (RTT) is computed as follows

$$RTT_{tcp} = 2 \cdot (T_{wd} + T_{wl}) \quad (8)$$

and the retransmission time out (RTO) can be approximated by $5 \cdot RTT_{tcp}$ [8]. So the RTO is expressed as follows

$$RTO_{tcp} = 10 \cdot \left(T_{wd} + T_{cts/rts} + N \cdot M_{avg} \cdot T_{frag}^{attempt} \right) \quad (9)$$

V. Conclusion and Future Work

In this paper, we studied the behavior of TCP over wireless links. Our focus was on the impact of fragmentation and retransmission at the 802.11 MAC layer on the TCP performance.

We first derived an analytical formula showing the relationship between the TCP loss rate and the number of MAC fragments as well as the maximum number of retransmissions. We then turned our attention to the end-to-end delay and we came up with a simple expression showing how the MAC parameters impact the TCP delay in the wireless segment. This is valid for infrastructure and ah hoc modes of 802.11 networks.

In our future works, we are planning to investigate how the TCP and 802.11 protocols should cooperate in order to avoid retransmissions at both transport and MAC layers. We have already identified a new problem concerning the performance of TCP may be affected dramatically with duplicated packet that may appear in the wireless medium. This affects not only the concerned TCP connection as the source slow down its window, but also competing connections that will have much less available bandwidth.

References

[1] A. Al-Hambali, E. Altman, P. Nain, A survey of TCP over mobile ad hoc networks, INRIA research report N. 5182, May 2004.

[2] M. Allman, V. Paxson and W. Stevens, TCP congestion control, IETF RFC 2581, April 1999.

[3] F. Anjum and R. Jain, Performance of TCP over lossy upstream and downstream links with link-level retransmissions, in Proc. of IEEE ICON 2000, September 2000.

[4] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and R. H. Katz, A comparison of mechanisms for improving TCP performance over wireless links, IEEE/ACM Transactions on Networking, 5(6):756-769, 1997.

[5] A-F. Canton and T. Chahed, End-to-end reliability in UMTS: TCP over ARQ, in Proc. of IEEE Globecom 2001, 2001.

[6] C.F. Chiasserini and M. Meo, A reconfigurable protocol setting to improve TCP over wireless, IEEE Transactions on Vehicular Technology, Vol. 51., No 6., November 2002.

[7] S. Floyd and T. Henderson, NewReno modification to TCP's fast recovery, RFC 2582, April 1999.

[8] P. Karn et al., Advice for Internet subnetwork designers, IETF, RFC 3819, July 2004.

[9] R. Ludwig and K. Sklower, The Eifel retransmission timer, ACM Sigcomm Computer Communication Review, v.30 n.3, p.17-27, July 2000.

[10] J. Padhye and S. Floyd, On interfering TCP behavior, in Proc. of ACM Sigcomm 2001, August 2001.

[11] V. Paxson and M. Allman, Computing TCP's retransmission timer, IETF RFC 2988, November 2000.

[12] S. Pilosof, R. Ramjee, D. Raz, Y. Shavitt, and P. Sinha Understanding TCP fairness over wireless LAN, in Proc. of IEEE INFOCOM 2003, April 2003.

[13] J. G. Proakis, Digital communications, pp. 831 and 275, 2001.

[14] T. S. Rappaport, Wireless communications - Principles & Practice, IEEE Press 1996.

[15] P. Sarolahti, M. Kojo, and K. Raatikainen, F-RTO: an enhanced recovery algorithm for TCP retransmission timeouts, ACM Sigcomm Computer Communication Review, V. 33 , N. 2, pp. 51-63, 2003.

[16] J. Yeo and A. Agrawala, Packet error model for the IEEE 802.11 MAC protocol, In Proc. of IEEE PIMRC 2003, September 2003.

[17] F. Vacirca, A.D. Vendictis, and A. Baiocchi, Investigating interactions between ARQ mechanisms and TCP over wireless links, in Proc. of European Wireless 2004, Barcelone, February 2004.

[18] F. Vacirca, A.D. Vendictis, A. Todini, and A. Baiocchi, On the effects of ARQ mechanisms on TCP performance in wireless environments, in Proc. of IEEE Globecom 2003.