Making H-ARQ Suitable for a Mobile TCP Receiver over LEO Satellite Constellations

Bastien Tauran[1(✉)], Emmanuel Lochin[1], Jérôme Lacan[1], Fabrice Arnal[2],
Mathieu Gineste[2], Laurence Clarac[3], and Nicolas Kuhn[3]

[1] Université de Toulouse, ISAE-SUPAERO TéSA, Toulouse, France
{bastien.tauran,emmanuel.lochin,jerome.lacan}@isae-supaero.fr
[2] Thales Alenia Space, Toulouse, France
{fabrice.arnal,mathieu.gineste}@thalesaleniaspace.com
[3] Centre National d'Etudes Spatiales, Toulouse, France
{laurence.clarac,nicolas.kuhn}@cnes.fr

Abstract. This paper investigates strategies to carry out delay tolerant services over LEO satellite constellations for mobile receiver. In this context, LEO constellations are characterized by important delay variations where propagation impairments are mostly localized on the Land Mobile Satellite (LMS) channel (i.e. on the last hop). To cope with this issue, distinct reliability schemes can be introduced at the physical or link layers. Although their capacity to cope with transmission errors has been demonstrated, these recovery schemes may induce a high jitter that could severely damage TCP's internal timers and reliability schemes. As a matter of fact, transport and link layers' reliability schemes exhibit a clear discrepancy. Following temporal traces representing the delay between a mobile terminal and the last hop satellite from a LEO constellation, we assess how HARQ mechanisms impact on the RTO based retransmission and the duplicate acknowledgments of TCP. Based on *ns*-2 simulations, we propose a layer-2 buffer that let both link and transport layers to conjointly perform. Our evaluations show an end-to-end data rate increase and more generally illustrate the benefit of re-ordering packets at the link layer when link-layer erasure coding recovery mechanisms are used conjointly with TCP.

Keywords: Constellations · TCP · Latency · Adaptive-HARQ

1 Introduction

Constellations systems such as Iridium connect urban and rural areas to terrestrial Internet broadband. As a result, significant amounts of TCP traffic between end-hosts on the Internet is expected to be carried out by constellations of satellites [1]. However, LEO constellations are characterized by important delay variations [2–4], where propagation impairments are mostly localized on the Land

P. Pillai et al. (Eds.): WiSATS 2017, LNICST 231, pp. 33–42, 2018.
https://doi.org/10.1007/978-3-319-76571-6_4

Mobile Satellite (LMS) channel [5]. In particular, as link errors often occur on the last mile and strongly impact on the end-to-end transmission, link and physical layers error correcting codes are deployed to minimize these errors [6–8]. However, reliable transport protocols remain necessary to ensure end-to-end reliability but the inadequacy of the interactions between both transport and link layers retransmission schemes may result in high end-to-end latency. As opposed to GEO satellite communications, LEO constellations are expected to provide a lower propagation delay. That being said, this may not ensure a lower end-to-end delay and a better quality of experience for latency sensitive applications if, *e.g.*, there is no cross-layer considerations. With LEO constellations, the TCP connection may not require specific tuning and/or to be split using Performance Enhancing Proxies (PEP) as defined in RFC3135.

Considering high delay and its variability in LEO satellite constellations, space and aeronautical communications often lays on delay tolerant services. Such services rely to applications which are bounded by a delivery delay for a certain percentile of messages. For instance, reliable sensor data transfer or aeronautical message services might require that 95% of data messages respect a certain delay threshold [9]. To ensure this kind of service, and to cope with losses that might occur on LMS, several ARQ and Hybrid ARQ schemes have been proposed [8,10]. Basically, these solutions make more robust this last link by adding error correcting, erasure coding, or enhanced Automatic Repeat reQuest (ARQ) schemes. One novel and promising solution is Adaptive-HARQ, which is an evolution of type II HARQ [11]. Once again, although such link layer mechanisms greatly mitigate the percentage of packets loss seen by the transport layer, they do not ensure a fully reliable service.

We acknowledge that multiple studies have analyzed the relation between ARQ and reordering. These investigations have even led to standard documents (RFC3366, 3GPP standard for RLC). These contributions show the interest of introducing a reordering buffer, when layer 2 reliability mechanisms are deployed. However, considering the actual trend in **deploying constellations of LEO satellites, our study fills a gap by assessing the relevance of such solution in this specific context**.

In this paper, we study the performance of bulk data transfers carried out by either TCP NewReno (denoted NR in the results) and CUBIC over a LEO constellation. The objective is to assess the impact of error mitigation mechanisms deployed at the link layer on the transport layer, which may trigger spurious retransmissions when high jitter variation occurs. We first study a set of TCP metrics impacted by this variable delay (number of spurious retransmissions, DUPACK and timeout). Following these preliminary results, we identify the root cause of the problem and propose a re-ordering buffer scheme that greatly improves TCP data delivery ratio. We believe that this solution should be deployed in any case where H-ARQ mechanisms are conjointly working with TCP.

2 Scenario

This section presents how we simulate the satellite environment and the different schemes that are considered through out this paper.

2.1 Satellite Environment

To simulate the satellite environment, we used both Network Simulator 2 (ns-2) and `SaVi` [12] to simulate a constellation composed of 66 satellites on Low Earth Orbit, at an altitude of 800 km. This constellation ensures a global coverage of any point of the earth, at any time.

The transmission delay varies within the satellite constellation because of the satellites' movements and the route changes. Considering the characteristics of our constellation, the time from the gateway to the end user's terminal varies between 70 ms and 90 ms. We consider that apart from the forward LMS links, there are no transmission errors. This assumption eases the impact of reliability schemes analysis on the forward transmission (*i.e.* from the gateway to the terminal) and remains consistent as Inter-Satellite Links (ISL) do not usually exhibit transmission errors. We consider that the routing within the constellation prevents congestion drops inside the constellation. The topology simulated is described in Fig. 1.

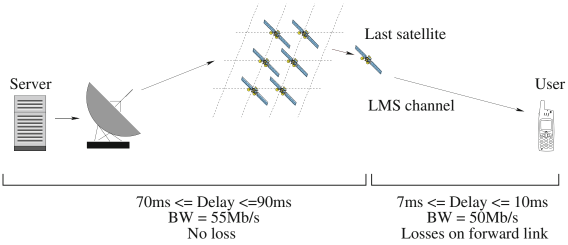


Fig. 1. Model for a satellite constellation

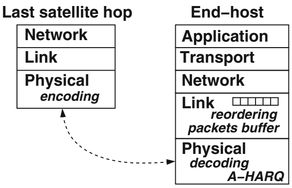


Fig. 2. Layering

2.2 Hybrid Automatic Repeat ReQuest

The Hybrid Automatic Repeat reQuest (HARQ) mechanism is a physical layer solution which aims at increase both physical and link layers reliability. In particular, this paper considers a novel HARQ scheme, Adaptive-HARQ [11] that has been designed to optimize the bandwidth utilization, as opposed to other HARQ mechanisms. We implemented this scheme as a new module for ns-2. This module, working on physical layer, allows the source node to send redundancy bits to the receiver, up to a fixed number of times, until the message is decoded. If the message is not decoded after the maximal number of (re)transmissions, it is then considered as lost and is dropped by the A-HARQ module. Following [11], we authorize up to 3 retransmissions. Figure 3 details how this module is working.

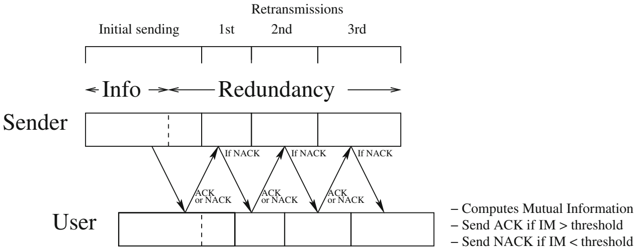


Fig. 3. Description of HARQ

The receiver side of the A-HARQ link uses a buffer to store packets not yet decoded. Each time useful or redundancy bits are received, the module computes the Mutual Information (MI) of the packet, depending on the state of the channel at this moment and the total number of bits already sent. The packet is considered as decoded if the value of the Mutual Information is higher than a given threshold.

As A-HARQ involves both the link and physical layers, we simulate with ns-2 the physical layer using mutual information (MI) and the channel characteristics following data contained in [11]. On the worst case, the packets are recovered at most 70 ms after entering A-HARQ module. This worst case corresponds to the time between a packet is sent from the last hop satellite and the packet is decoded after three A-HARQ retransmissions.

2.3 Version of TCP and Parameters

TCP has been showed to provide reasonable performances over LEO constellation [1,13], without requiring specific PEP optimization mechanisms, we thus consider TCP NewReno (NR) and CUBIC as TCP variants for our evaluations. Even though TCP NewReno is less and less populating the Internet, this protocol is of interest for our problematic since it features basic reliability functionalities of most TCP variants. We also consider CUBIC since (1) its error recovery is more aggressive than TCP NewReno and (2) it is enabled by default in GNU/Linux and OSX systems (since `10.9`).

We test TCP/SACK performance with standard GNU/Linux parameters. The first one is Duplicate SACK (DSACK), which is enabled by default on GNU/Linux systems. DSACK allows TCP sender to differentiate in some cases, if a Fast Retransmit is due to a reordering or to a packet loss. Thus, the TCP sender can adjust the duplicate acknowledgment threshold (defined in RFC2581), which is usually 3 by default in GNU/Linux.

We also evaluate the impact of Delayed Acknowledgment (DelAck) on our proposed solution. Delayed ACK (defined in RFC1123) can combine two in-order packets if they are received within a fixed time window, which is 40 ms in GNU/Linux systems.

2.4 Simulation Scenario

We designed a scenario with three nodes representing the sender, the last satellite on the message route, and the receiver. We simulate the constellation by changing the link delays, using a temporal trace (obtained with `SaVi`) giving at any moment the value of the delay between the sender and the last satellite, and the last satellite and the receiver. These delays have been measured on a initial *ns*-2 simulation using the LEO constellation described in Sect. 2.1. For all simulations, we use these delays traces obtained to mimic the constellation. Each simulation lasts 600 s with one single TCP performing. The simulations are run with a Land Mobile Satellite (LMS) channel [14], [15] between the last satellite and the ground gateway and an ITS (Intermediate Tree Shadowed Environment) environment. The LMS channel enables the Adaptive-HARQ module. We also vary the quality of this channel by setting an average SNR ranging from 7 dB to 13 dB. During the simulations, the link quality changes over time around this SNR average value according to a trace file representing the evolution of the LMS channel. For all other links (return link included), we assume that there are no errors to ease the interpretation of the results.

3 On the Need to Mitigate the Impact of Out-of-Order Packets

In this section and the next ones, the numbers of RTO, DUPACK and spurious retransmission have been normalized by the number of packets sent in order to have comparable results with the different graphs and tables.

Table 1 shows the goodput achieved by TCP without reordering mechanism. The RTO and DUPACK columns represent the proportion of recovery events (due to link errors or congestion events). We divided the number of times TCP received a request for retransmission by the total number of packets sent. For example, when $SNR = 7\,\text{dB}$, during 600 s, CUBIC transmitted 21406 packets, the RTO timer expired 54 times and 500 retransmissions have been triggered due to DUPACK. We have also measured 480 spurious transmissions. Thus, the proportion for RTO is 0.25%, for duplicate acknowledgments is 2.34% and for spurious is 2.24%.

We observe that we have a low goodput, whatever the value of SNR, whereas we could expect a goodput of 40 Mb/s considering no errors occurred in the network. On the other hand, the number of DUPACK and spurious retransmission remains high, and is not decreasing when the channel quality improves as we could expect. This is mainly due to out-of-order packets interpreted by TCP as congestion losses which trigger spurious retransmissions and halve the congestion window.

To let TCP exploit the available capacity, we need mechanisms that mitigate the effects of out-of-order packets. In the next section, we exploit a possibility that is adding a reordering mechanism after A-HARQ.

Table 1. No reordering mechanism

SNR (dB)	TCP Goodput (kb/s)		A-HARQ Success (%)		RTO (%)		DUPACK (%)		Spurious (%)	
	NR	CUBIC	NR	CUBIC	NR	CUBIC	NR	CUBIC	NR	CUBIC
7	188	265	95.30	95.11	0.62	0.25	2.84	2.34	1.81	2.24
8	242	349	97.22	97.08	0.23	0.05	2.85	2.19	1.86	2.00
9	261	385	97.80	97.72	0.16	0.01	2.77	1.99	1.63	1.82
10	290	431	98.32	98.37	0.08	0	2.72	1.82	1.69	1.75
11	312	469	98.84	98.85	0.04	0	2.59	1.63	1.46	1.71
12	328	494	99.14	99.21	0.02	0	2.56	1.54	1.48	1.73
13	346	511	99.38	99.45	0.01	0	2.44	1.52	1.59	1.75

4 Solution Proposed

The inconvenient of A-HARQ is that packets are delivered out-of-order to the transport layer, due to the changing number of retransmissions by this mechanism, and also by route changes. This strongly impacts on TCP performance and generates DUPACK causing spurious retransmissions. In the next section, we show that adding a reordering mechanism, after A-HARQ and before sending the packets to the transport layer, improves the performance of TCP.

4.1 Adding a Reordering Mechanism

This reordering mechanism is composed of a buffer storing TCP out-of-order packets. A decoded packet from A-HARQ is directly forwarded to the upper layer if in sequence otherwise stored. This mechanism has to deal with the case where the buffer limit is reached and if a packet has not been decoded by A-HARQ. The rationale for the later is to speed up TCP recovery procedure. For both cases:

- we limit the size of the reordering buffer to 125 packets. This value is computed to prevent RTO and results from an heuristic based on throughput and retransmission delay on the last hop. When full, all these out-of-order packets are forwarded to the upper layer. Note this case rarely occurs in our simulation scenario as the traffic load is low (i.e. we only send one long-lived flow) and the route changes in the satellite constellation considered does not lead to several out-of-order packets;
- concerning the case when H-ARQ decoding fails, we propose to flush the buffer even if not full. Thus, all packets are forwarded to the upper layer allowing TCP to quickly recover this missing packet with DUPACK or RTO.

A global view of the last hop is given in Fig. 2, with the A-HARQ module and the reordering buffer.

4.2 Results with Reordering Mechanism

Table 2 shows the results obtained when the reordering mechanism operates conjointly with A-HARQ. We observe an improvement in terms of goodput due to

Table 2. With reordering mechanism

SNR (dB)	TCP Goodput (kb/s)		A-HARQ Success (%)		RTO (%)		DUPACK (%)		Spurious (%)	
	NR	CUBIC	NR	CUBIC	NR	CUBIC	NR	CUBIC	NR	CUBIC
7	409	363	96.06	94.95	0.62	0.59	0.52	0.56	1.45	4.00
8	521	640	97.46	97.18	0.41	0.29	0.40	0.50	1.20	4.65
9	628	753	98.11	97.83	0.30	0.22	0.35	0.45	1.27	4.47
10	749	993	98.72	98.38	0.23	0.16	0.27	0.40	1.21	3.94
11	818	1447	98.97	98.94	0.19	0.09	0.25	0.32	1.13	4.02
12	958	1877	99.27	99.22	0.14	0.06	0.20	0.28	1.20	4.04
13	1134	2367	99.47	99.47	0.10	0.05	0.17	0.25	1.02	3.92

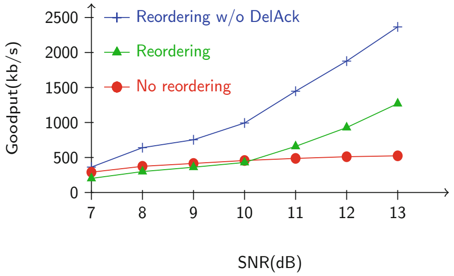
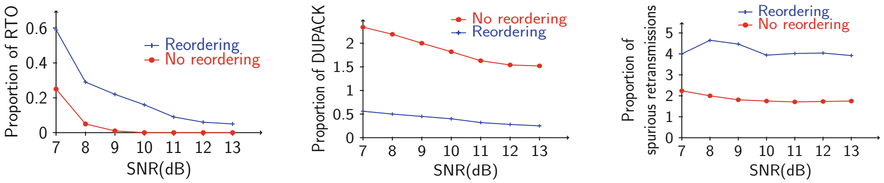


Fig. 4. Impact of reordering mechanism on end to end goodput



(a) Impact on RTO (b) Impact on DUPACK (c) Impact on spurious

Fig. 5. Impact of reordering mechanism on CUBIC performance

the decreasing number of retransmissions. We also observe a slightly increase of the number of RTO although the overall performance are much more better than without reordering as shown in Table 1. Actually, some losses are not recovered by DUPACK but by RTO because of decoding fails or when the queue is full as explained in the previous Sect. 4. We also observe an increase of the number of spurious retransmissions when the TCP goodput increases significantly. The reason is that TCP becomes more and more opportunistic as the network conditions are getting better and then congestion appears on other parts of the networks. As a result, thanks to the conjointly used of the A-HARQ and reordering mechanisms, the bottleneck is not the LMS channel anymore (Fig. 5).

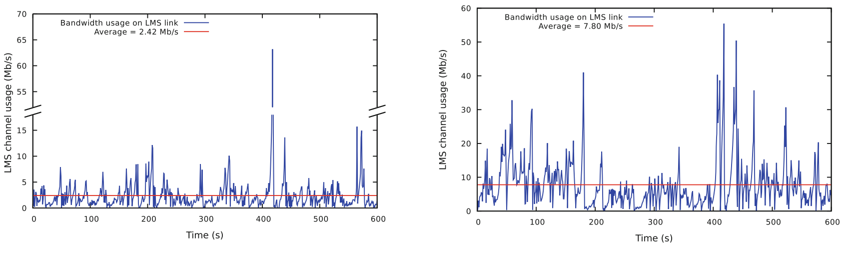
We recall that we only presented results with TCP NewReno and CUBIC. However, the same trend has been observed with TCP Westwood and should be also observed with other TCP variants. CUBIC is well-known to achieve better performance over high-delay bandwidth product networks. This explains the higher performance obtained by this protocol compared to TCP NewReno.

5 Analysis

The results presented in Sect. 3 highlight the interest for mitigating the number of out-of-order packets linked to the use of link layer reliability schemes. As shown in Sect. 4, link layer buffering results in a strong increase of the TCP goodput with a significant decrease of the number of DUPACK and spurious retransmissions for TCP NewReno. This is not only a gain for the use of the expensive satellite capacity, but also for the end-to-end latency. At last but not least, this makes it easier for delay tolerant services to respect their delivery rate constraints.

Concerning standard default TCP parameters tested, using DelAck with the reordering mechanism is counterproductive for TCP performance. As shown in Fig. 4, we observe that the goodput decreases when DelAck is added with the reordering buffer while better without this buffer. The problem is that DelAck might delay the acknowledgment pace while the reordering buffer might also delay the pace of data forwarding to the upper layer. As a matter of fact, both delayed schemes negatively interact between them. Our recommendation is thus to disable DelAck in this context to reach the highest performance if possible. Note that in another context of multipath communications over a satellite constellation, the authors in [1] showed that DelAck has also a negative impact on TCP performance.

With reordering mechanism and when the goodput is high enough, congestion appears on the network, and packets are dropped for reasons that are not due to satellite environment or A-HARQ. This means that A-HARQ is recovering enough packets to allow TCP to work in good conditions. So, TCP metrics are more impacted by congestion in the network than by the few packets dropped



(a) SNR=8 (b) SNR=13

Fig. 6. Usage of the LMS link, using CUBIC

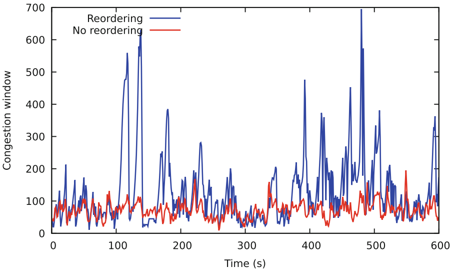


Fig. 7. Evolution of CUBIC congestion window, averaged each second, with SNR = 8

by A-HARQ. In these scenarios, we observe a good usage of the LMS link. We can see in Fig. 6b, corresponding to $SNR = 13$ and with reordering, that the maximum capacity of this link is reached sometimes, with an average usage of 7.8 Mb/s. As a reminder, we set the capacity of this link at 50 Mb/s. This usage corresponds to the real number of bits flowing through this link, including the redundancy bits and the retransmissions by A-HARQ. This usage is even better with higher SNR.

We also observe, as presented in Fig. 7, a real improvement of the congestion window with reordering mechanism. This evolution is correlated with the channel evolution, and confirms the goodput improvement brought out by this mechanism.

To conclude, the results we obtained for A-HARQ can be extended to all HARQ mechanisms, since they all imply varying delays and out-of-order packets in TCP layer. Thus, **adding a reordering mechanism after HARQ should be the standard for all HARQ mechanisms, in order to always guarantee good TCP performance.**

6 Conclusion

We investigated in this paper the performance obtained by TCP within a delay tolerant service over mobile satellite communications. We show that link-layer schemes used in the context of space or aeronautical communications have an impact on the TCP performance. We observe that using TCP over such link-layer schemes leads to weak performance due to high delay jitter. Basically, there is a clear discrepancy between TCP and these link-layer schemes while sharing the same goal. Using temporal traces representing the delay between a mobile terminal and the last hop satellite from a LEO constellation, we identified link-layer parameters that impact on the TCP performance. We proposed a buffer scheme allowing both link-layer and transport layers to conjointly perform. Simulations show that thanks to this simple buffer scheme, we can drastically increase the TCP performance without modifying TCP internal algorithms. In a future work, we seek to extend these preliminary measurements with various kind of traffic and in particular, to assess the impact on short-lived flows over a real satellite link provided by CNES agency.

References

1. Wood, L., Pavlou, G., Evans, B.: Effects on TCP of routing strategies in satellite constellations. IEEE Commun. Mag. **39**(3), 172–181 (2001)
2. Pratt, S.R., et al.: An operational and performance overview of the IRIDIUM low earth orbit satellite system. IEEE Commun. Surv. **2**(2), 2–10 (1999)
3. Taleb, T., Kato, N., Nemoto, Y.: Recent trends in IP/NGEO satellite communication systems: transport, routing, and mobility management concerns. IEEE Wirel. Commun. **12**(5), 63–69 (2005)
4. Diana, R., et al.: DTN routing for quasi-deterministic networks with application to LEO constellations. Int. J. Satell. Commun. Network. **35**, 1–18 (2015)
5. Blaunstein, N., Cohen, Y., Hayakawa, M.: Prediction of fading phenomena in land-satellite communication links. Radio Sci. **45**(6) (2010)
6. Sastry, A.: Performance of hybrid error control schemes of satellite channels. IEEE Trans. Commun. **23**(7), 689–694 (1975)
7. De Cola, T., Ernst, H., Marchese, M.: Application of long erasure codes and ARQ schemes for achieving high data transfer performance over long delay networks. In: Re, E.D., Ruggieri, M. (eds.) Satellite Communications and Navigation Systems. SCT, pp. 643–656. Springer, Boston (2008). https://doi.org/10.1007/978-0-387-47524-0_49
8. Kuhn, N., et al.: On the impact of link layer retransmission schemes on TCP over 4G satellite links. Int. J. Satell. Commun. Network. **33**(1), 19–42 (2015)
9. Gineste, M., Gonzalez, H.: On the usage of a dedicated data link layer and half duplex terminals in satellite systems for future air traffic management. In: 28th AIAA International Communications Satellite Systems Conference (2010)
10. Kuhn, N., et al.: On the impact of link layer retransmissions on TCP for aeronautical communications. In: 5th International Conference on Personal Satellite Services - PSATS (2013)
11. Ahmad, R.A., et al.: Enhanced HARQ for delay tolerant services in mobile satellite communications. In: The Seventh International Conference on Advances in Satellite and Space Communications SPACOMM, Barcelona, ES, pp. 1–6, April 2015
12. Wood, L.: SaVi: satellite constellation visualization. In: First Annual CCSR Research Symposium - CRS 2011 (2012)
13. Chotikapong, Y., Cruickshank, H., Sun, Z.: Evaluation of TCP and Internet traffic via low Earth orbit satellites. IEEE Pers. Commun. **8**(3), 28–34 (2001)
14. Perez-Fontan, F., et al.: S-band LMS propagation channel behaviour for different environments, degrees of shadowing and elevation angles. IEEE Trans. Broadcast. **44**(1), 40–76 (1998)
15. Fontan, F.P., et al.: Statistical modeling of the LMS channel. IEEE Trans. Veh. Technol. **50**(6), 1549–1567 (2001)