TCP Performance Evaluation over GEO and LEO Satellite Links between Performance Enhancement Proxies

Fei Peng[1], Ángel Salamanca Cardona[2], Kaveh Shafiee[1], Victor C. M. Leung[1]
[1]Department of Electrical and Computer Engineering, University of British Columbia
Vancouver, BC, Canada V6T 1Z4
[2] Facultad de Informática, Universidad Politécnica de Madrid

Abstract—**Transmission Control Protocol (TCP) is the most widely used protocol for end-to-end data transmissions over inter-networks. However, the congestion control algorithm of TCP was not created to match the special characteristics of satellite links, resulting in a drastically performance degradation. The employment of TCP split connections coupled with link level retransmissions and satellite link tailored TCP versions has been proven effective in hiding corruption losses from TCP to improve end-to-end performance. Using NS-2 simulations, this paper evaluates and compares popularly used TCP versions such as New Reno, Hybla, Vegas, DVegas and Westwood+ over GEO VSAT- and LEO-based satellite links employing performance enhancement proxy (PEP) based on snoop. Simulation results show that with the use of snoop proxy, TCP Hybla outperforms the rest of TCP versions, improving the usage of satellite links that are affected by losses not due to congestion. The second good candidate is DVegas over satellite PEP. When the Explicit Control Protocol (XCP) is employed as TCP PEPs, it would improve TCP performance to get high transfer rates and respond to congestion in a correct and scalable way.**

Keywords-Transmission Control Protocol (TCP); Satellite; Performance Enhance Proxy (PEP)

I. INTRODUCTION

In the next generation Internet, satellite links will play an important role by providing broadband Internet access and high-speed backbone network connectivity between remote networks through easily scalable architecture and multicast capabilities [1]. Besides, they are the only way Internet can reach isolated locations, covering large areas of the Earth.

Satellite networking began with the use of individual satellites in Geostationary Orbit (GEO). However, requirements for lower propagation delays and propagation loss, in conjunction with the coverage of high latitude regions for personal communication services, have pushed the development of new satellite communication systems called Low Earth Orbit (LEO) satellite systems. Differences between LEO and GEO satellites will be discussed later in the paper.

The Transmission Control Protocol (TCP) is the mostly used protocol for data transmissions over networks. However, TCP performs poorly in heterogeneous networks with satellite links because of their characteristics: long propagation delays, large bandwidth-delay products, high bit error rates and bandwidth asymmetry. As TCP needs to receive an ACK of every sent segment to increase its *CWND*, the long propagation delays over the satellite make slow start unable to reach the maximum achievable throughput. Secondly, the offered window size field in the TCP header is only 16-bit long [2], which restricts its value to 64K. The third problem is that TCP misinterprets packet losses as an indication of congestion and reduces the transmission rate, thus leading to TCP throughput degradation. Finally, the low bandwidth uplink path may become easily congested leading to delay of ACKs, which causes a TCP sender to transmit data segments in bursts.

This work is jointly supported by the National Natural Science Foundation under Grant 61102009 and the Natural Sciences and Engineering Research Council of Canada (NSERC).

The rest of the paper is organized as follows. Section 2 is the survey of related work, section 3 introduces the system architecture, Section 3 presents the performance evaluation and simulation results. Finally, we make the conclusion and future work in section 4.

II. SURVEY OF RELATED WORK

In light of the problems mentioned in Section I, many researchers have proposed solutions to improve TCP performance over satellite networks. The possible solutions can be classified into three categories: link layer solutions, end-to-end solutions and performance enhancement proxy (PEP) solutions. Link layer solutions include link layer retransmissions and forward error correction [3], to mitigate packet losses over satellite links due to transmission errors. However, these methods cannot solve the problems experienced by TCP over satellite links caused by latency. Some end-to-end protocols were designed specially for the satellite environment, we categorize them as TCP satellite-optimized protocols such as TCP NewReno, TCP Hybla, TCP Vegas, TCP Veno, TCP Westwood+, TCP Peach, and Non-TCP satellite-optimized protocols such as SCPS-TP, STP, XCP, SCTP; however, these protocols are not practicable for accessing the global Internet as not all end systems support such TCP extensions or modifications.

The PEP approach has been widely considered as an effective and practical solution to improve TCP performance over satellite links. The advantage of deploying PEPs is that they can act on behalf of end systems without changing their TCP configurations. Two kinds of TCP PEPs have been proposed in the literatures: TCP spoofing PEPs [4] and TCP split connection PEPs [5]. When a PEP ‘splits’ a connection, the proxy pretends to be the opposite endpoint of the connection in each direction, literally splitting the connection into two. When a PEP ‘snoops’ into a connection, Spoofing proxies controls the transmissions of the TCP segments in both directions, by ACK filtering and reconstructing the existing connection. However, the problems of long congestion control

978-1-4673-1881-5/12/$31.00 ©2012 IEEE

feedback delay and inability to differentiate between packet losses due to congestion or transmission errors are still present but transferred to the spoofing proxies. In this paper, we propose to apply TCP end-to-end protocols to a PEP snoop for the performance enhancement over satellite networks.

Concerning end-to-end solutions TCP Reno and TCP New Reno [6] are the most commonly implemented algorithms. SACK support is very common and is an extension to Reno/New Reno. The others are competing proposals which still need evaluation. Here we mainly introduce several TCP versions used at satellite PEPs.

For TCP satellite-optimized protocols, TCP Westwood+ [7] is an evolution of TCP Westwood. Its main novelty is the algorithm used to estimate the available end-to-end bandwidth. This improves TCP Westwood's original estimation algorithm, but does not work well in the presence of reverse traffic due to ACK compression (aliasing effects). TCP Hybla [8] employs a time-scale modification algorithm to increment the congestion window *CWND* independent of the round-trip time (RTT) during the slow start and congestion avoidance phases. However, it assumes that the transmission rate does not depend on *RWND*. The TCP Hybla algorithm employs the SACK and timestamp options to recover multiple losses and prevent delays in Retransmission Time-Out (RTO) timer update, respectively. Based on TCP Vegas, TCP DVegas is applied over the satellite links in split TCP connections, and employs active queue management at the medium access control layer for immediate cross-layer congestion feedback to the TCP virtual sources at the PEPs. This method substantially shortens congestion feedback delay and enables transmission losses to be effectively differentiated from congestion losses. Simulation results show that it improves the performance of TCP Vegas when traffic load is heavy and transmission losses increase [9].

The non-TCP satellite-optimized protocols require modification at intermediate nodes. Explicit Control Protocol (XCP) [10] and its enhanced variant P-XCP [11] employ explicit feedback to determine network conditions and decouple utilization and fairness control. This decoupling of congestion control's efficiency and fairness policies enables routers to quickly make use of available bandwidth while conservatively managing the allocation of bandwidth to flows. XCP is built upon a new principle: carrying per-flow congestion state in packets. XCP packets carry a congestion header through which the sender requests a desired throughput. Routers make a fair per-flow bandwidth allocation without maintaining any per-flow state. Thus, the sender learns of the bottleneck router's allocation in a single round trip.

As far as we know, only DVegas and XCP have been used in satellite PEPs among those TCP end-to-end satellite protocols. Their best results are obtained when used in satellite segments of split TCP connections compare to corresponding TCP protocols employed over the end-to-end satellite links. In [12] and [13], XCP is used with router-assisted Explicit Rate Notification (ERN) to improve end-to-end performance without split TCP connection over satellite PEPs. However, their simulations were either limited between XCP and P-XCP [12]; or focused on TCP fairness performance between flows [13] that is beyond the scope of the paper.

In [14], a performance analysis of three TCP variants of Sack TCP, Hybla TCP and Cubic TCP are carried out using the NS-2 simulator over a Digital Video Broadcasting – Second Generation (DVB-S2) satellite system. A new cross-layer TCP splitting architecture with an optimized congestion control algorithm XPLIT-TCP are specially devised for DVB-S2 network in [15]. However, both [14] and [15] only take into account of the challenge faced by GEO satellite systems. The evaluations in [14] and XPLIT-TCP in [15] mainly consider the Differentiated Service (DiffServ) architecture to provide quality of service guarantees, which is out of the paper's scope.

III. CONTRIBUTIONS

This paper aims at evaluating and comparing five control algorithms, which are New Reno, Hybla, Vegas, DVegas and Westwood+, over GEO and LEO satellite links between PEPs using NS-2 simulations.

We have chosen these TCP versions as they are among the best performance candidates available from the literature. TCP New Reno is always interesting to look at and compare to, since it is the most common implementation of TCP. TCP Westwood+ is a similar and also a widely recognized version of TCP. TCP Vegas and TCP Hybla, are specialized for long delay error prone networks. Finally, TCP DVegas improves TCP Vegas by adding an immediate cross-layer congestion feedback to the TCP virtual sources at the PEPs. TCP Reno and SACK are included as baselines for our comparisons.

IV. SYSTEM ARCHITECTURE

A. GEO VSAT- and LEO-based networks

Both the GEO VSAT- and LEO- based networks can work only when the terminal and the satellite "see" each other. The very small aperture terminals (VSAT) communicate with geostationary satellites over a 36.000 km distance, which can be seen only from a part of earth surface. GEO satellite's strong points are: the simple base station working mechanism (as GEO satellites are always on the same position, no tracking and handover procedures are needed), their higher reliability, long operational lifetime and lower development risks. Their main weaknesses are: onboard complexity, expensive cost and the large number of users affected if malfunction is experienced.

On the other hand, LEO satellite constellations are at lower distances of 1,500 km, and are visible practically from any point on the earth. LEO satellite's strong points are: its lower propagation delay, lower RTTs and its ability to communicate with small devices. The main weaknesses of LEO satellites are: the complex tracking procedures and handover mechanisms needed at the base station and the fact that a whole constellation has to be developed in order to provide broad coverage.

B. Split Connection with Performance Enhancement Proxy

A typical scenario for PEP implementation is a satellite link along the data path. The PEP agent is installed at the edge router bridging terrestrial and satellite networks. As a result, the end-to-end communication flow is split into two sections: a low-delay terrestrial link using standard TCP and a high-delay satellite link using a protocol specifically designed for satellite networks. For the satellite link, suitable scenarios based on two GEO and polar LEO satellite configurations were considered and implemented.

We also had to decide which kind of PEP we wanted to use in our simulation. NS-2 simulator offers the possibility to work with either spoofing or splitting PEPs. The latest modules available for NS-2 simulator are Snoop (spoofing) and SaTPEP (splitting). On the one hand, SaTPEP splitting proxy introduces specialized congestion control and loss recovery mechanisms based on link utilization measurements. On the other hand, the Snoop proxy is implemented in the data link layer but is capable of monitoring the flow of TCP segments in both directions. It stores in its cache all the TCP segments that have not yet been recognized by the receiver. The main advantage of this is that the proxy retransmits the lost segments locally. This way the sender doesn't have to recognize duplicated segments and no longer has to use congestion control mechanisms and fast recovery. There is not a dominant PEP implementation used with VSAT or LEO networks. For this simulation we decided to implement a Snoop network model.

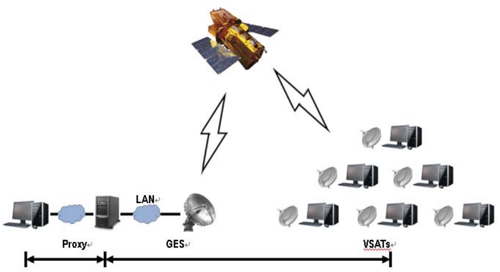

Figure 1. System Architecture I: GEO "bent pipe" satellite architecture.

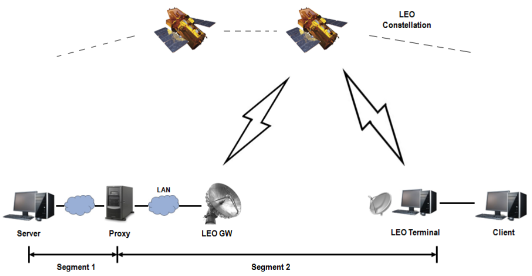

Figure 2. System Architecture II: polar orbiting LEO constellation architecture.

When working with Ns-2 simulator, we decided to follow a divide and conquer strategy to accomplish our final system architecture simulation. Nowadays, VSAT networks are implemented with geosynchronous satellites, most of them using a star topology. This way, our first step was to implement a star topology as shown on Figure 1. A large hub earth station is located at the centre of the star with VSATs used at the remote sites of the network. For the polar LEO configuration, Ns-2 simulator offers two network models: Iridium and Teledesic. Here, we use to model an Iridium polar orbiting LEO constellation in Figure 2. In both network architectures a number of VSATs/LEO GW terminals (GEO and LEO configurations, respectively) are located at the subscriber premises, which enable end hosts to access the Internet via satellite. In both GEO and LEO split TCP configuration, one Snoop proxy is located at the gateway earth station (GES), connected to the GES through a LAN. The GES is used to connect the satellite network to local web servers or the global Internet. When an end host wants to set up a TCP connection, the Snoop PEP will intercept this request and set up cascading TCP connections for it. The gateway is granted fixed bandwidth in a dedicated uplink channel. The Snoop PEP is responsible for local acknowledgments and retransmission over the satellite network.

V. Performance Evaluation

We use NS-2 version 2.31 as the simulation tool for performance evaluations. The different TCP versions available are compared when applied over the proposed satellite link between PEPs. Comparisons of the TCP versions with end-to-end TCP protocols are not included in the paper since it is well-known that a PEP scheme will out-perform end-to-end TCP over a satellite network. Since TCP Reno is the most popular version implemented in current networks, we use it as baseline for the comparison with PEP schemes.

As shown on Figures 1 and 2, the implementation of the VSAT-based network includes 5 VSATs at the user side, while the LEO-based network includes one LEO terminal connected to the potential client. The TCP throughput is defined as the number of received data bits divided by the simulation time. Tables I and II list the simulation parameters used in simulations, except otherwise noted. Considering that system bandwidth should be shared by burst traffic to take advantage of statistical multiplexing, instead of long-lived traffic, we use a traffic model that approximate World Wide Web traffic as the web represents the most popular application that transmits data over TCP. The comparison also takes into account File Transfer Protocol (FTP) and User Datagram Protocol (UDP) as background traffic for our simulations.

Our performance evaluation simulation results can be divided into five groups, depending on which situations and characteristics are studied. These evaluation groups are the following: bit-error ratio (BER), Uplink bandwidth, Traffic load, Snoop proxy and XCP as TCP PEP.

A. *Impact of BER*

We test the impact of BER on TCP performance in both scenarios with the BER of the VSATs and LEO terminals at 10^{-7}, which is common in satellite networks in which rain attenuation affects a subset of the earth stations. The other simulation parameters have the same values as shown in Tables I and II.

TABLE I. Simulation parameters for VSAT model

Parameter Items	Parameter Values
Bandwidth for Terrestrial Link	10 Mbps
Propagation Delay for Terrestrial Link	50 ms
Packet length for downstream (incl. TCP/IP headers)	1024 Bytes
Packet length for upstream (incl. TCP/IP headers)	128 Bytes
Propagation delay for satellite link	250 ms
Simulation time	25 s

In the GEO "bent pipe" satellite scenario the overall throughput does not suffer much degradation due to the adaptive congestion mechanism of TCP. Other TCP connections take bandwidth from those suffering from BER degradation. TCP Hybla is specifically designed to work with high packet error rates, so its performance was good expected. DVegas ranked second because of its successful decoupling of congestion control and error control. As shown on Figure 3, TCP Vegas offers a better performance than TCP Reno because TCP Vegas only cuts its congestion window by one fourth if a packet is lost for the first time.

TABLE II. SIMULATION PARAMETERS USED FOR MODELING THE TELEDESIC POLAR ORBITING CONSTELLATION SYSTEMS

	Teledesic
Altitude	1375
Planes	12
Satellites per plane	24
Inclination (deg)	84.7
Interplane separation (deg)	15
Seam separation (deg)	15
Elevation mask (deg)	40
Intraplane phasing	Yes
Interplane phasing	No
ISL per satellite	8
ISL bandwidth	155 Mb/s
Up/downlink bandwidth	1.5 Mb/s
Cross-seam ISLs	Yes
ISL latitude threshold (deg)	60

In the polar orbiting LEO constellations scenario, Figure 4 shows that when working in prone error satellite networks, TCP Hybla once more outperforms the rest of TCP variants. Since TCP DVegas adaptively reduces the *CWND* and SSTRESH by taking into account the immediate congestion feedback for error differentiation that mitigates the impact of random errors not due to congestion that provoke premature reductions of TCP New Reno and TCP Vegas's control windows. In comparison with other TCP versions, TCP Hybla achieves much more performance enhancement in GEO than LEO, since TCP Hybla was designed not only to counteract BER, but RTT-unfairness as well. However, Other TCP versions in the simulation, for example, TCP Westwood+ was mainly conceived to tackle random error networks, its performance approach to TCP Hybla more closely in LEO than in GEO configurations.

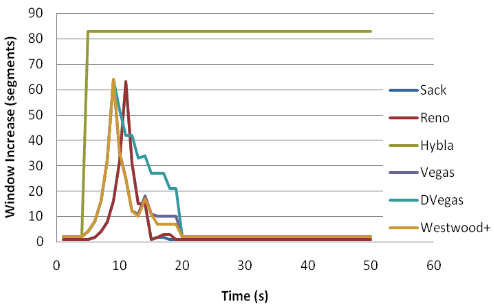

Figure 3. Impact of BER on Congestion Window increase of VSAT terminals experiencing BER=10^{-7}

B. *Impact of Uplink bandwidth*

In the polar orbiting LEO constellations scenario, the throughput results do not suffer much impact when the uplink bandwidth is modified. This can be explained by the lower propagation delay of the Inter Satellite Link (ISL) of LEO networks. This way, in this section we will focus on evaluating the results from the GEO "bent pipe" satellite scenario. We investigate the impact of the uplink bandwidth (from VSAT to proxy) on the TCP throughput as well as the round trip delay (satellite segment only). While the downlink bandwidth (from proxy to VSAT) is kept constant at 6 Mbps, the uplink bandwidth changes from 2Mbps to 10Mbps.

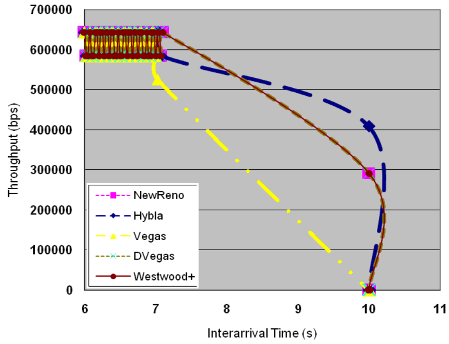

Figure 4. Impact of BER on Throughput of LEO terminals experiencing BER=10^{-7}

Figure 5 shows when the uplink bandwidth is 2Mb, all the TCP protocol suffer a clear degradation in throughput. TCP Hybla is the fastest in exploiting the link bandwidth, reaching the steady state before the other TCP variants. Packet spacing and initial bandwidth estimation, allows TCP Hybla to avoid the early buffer overflow produced when packets first arrive to the router. This buffer overflow affects all other TCP variants, except DVegas and TCP Westwood+.

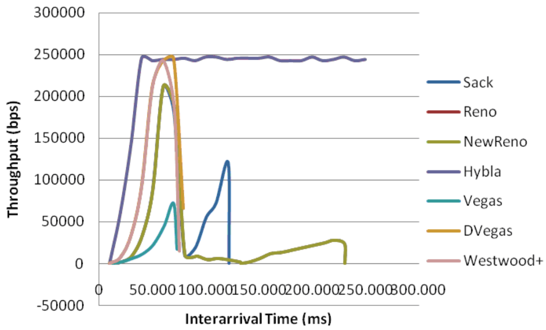

Figure 5. Impact of Uplink bandwidth. GEO "bent pipe" satellite scenario. Throughput. Uplink bandwidth = 2Mb, Download bandwidth = 6Mb, BER = 10^{-7}

C. *Impact of traffic load*

Figure 6 shows the simulation results when the load is heavy on the LEO scenario. As explained above, TCP Hybla outperforms the rest of the TCP variants. TCP Reno performs worst. This can be explained considering the fact that TCP Reno needs more time to recover from packets dropped by transmission errors, while SACK can recover quicker using selective acknowledgments. GEO produced similar results to LEO when traffic load is modified, however, TCP Vegas has the worst throughput, being unable to cope with the small thresholds ballast. On the GEO scenario, TCP Hybla's results have large differences from others, we omit the GEO figure since it is hard to appreciate the other TCP variants if we want to show TCP Hybla results.

D. *Impact of XCP as TCP PEP*

When considering the study of other protocols that are able to dynamically adapt to the congestion state of the network, XCP seems the best option. XCP represents a major advance in Internet congestion control, delivering the highest possible application performance over a broad range of network infrastructure, including extremely high speed and very high delay links that are not well served by TCP. XCP is built upon a new principle: carrying per-flow congestion state in packets.

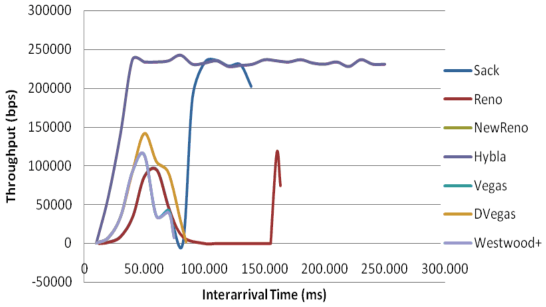

Figure 6. Impact of traffic load on Throughput. LEO Constellation satellite scenario.

XCP may be used in between two TCP PEPs, but its router implementation on Ns-2 simulator is beyond the scope of this paper. However, as a simple approach, we created a bottleneck scenario that reproduced the throughput simulation results obtained by the TCP variants in the GEO satellite bottleneck link and we tested XCP's performance in this scenario. From the results shown on Figure 7, we can state a hypothesis: when used as a TCP PEP, XCP would obtain high throughput performance since it helps the sender to learn about the bottleneck router's allocation in a single round trip without maintaining any per-flow state.

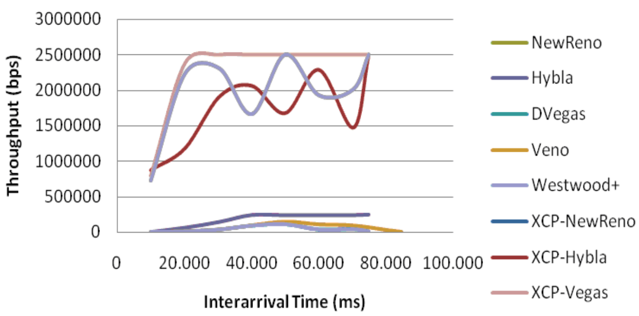

Figure 7. XCP Throughput over bottleneck satellite link.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we have evaluated and compared five TCP versions, which are New Reno, Hybla, Vegas, DVegas and Westwood+, over GEO and LEO satellite links employing Snoop PEPs. The obtained results show that TCP Hybla allows a better efficiency in utilizing the available satellite bandwidth, especially for high packet error rates and high satellite bandwidth values. From our results we can conclude that Snoop proxy improves end-to-end performance of the satellite link. We have made contributions in (1) sharing our experience on modeling suitable scenarios based on GEO bent-pipe satellites and polar orbiting LEO constellations with ns2 simulator, (2) describing a satellite network simulation methodology, and (3) presenting our results and gaining understanding of transport protocols in satellite networks.

Three research directions appear of major interest for further investigations on the subject: the first one concerns the inclusion of other types of PEPs (SaTPEP) and their enhancements in our system architecture and compare them with the results obtained by Snoop. The second one is to continue working on using XCP as TCP PEP, and the third one is to explore and impulse artificial intelligence approach to satellite networks.

REFERENCES

[1] A. Jamalipour, M. Marchese, H. Cruickshank, J. Neal, and S. Verma, "Broadband IP Networks via satellites-part I," IEEE J. Select. Areas Commun., vol. 22, no. 2, pp. 213–217, February. 2004.

[2] M. Allman, D. Glover, and L. Sanchez, "Enhancing TCP over satellite channels using standard mechanisms", RFC2488, January 1999.

[3] V. Jacobson and R. Braden, "TCP Extensions for Long-Delay Paths", RFC1072, October 1988.

[4] J. Ishac and M. Allman, "On the performance of TCP spoofing in satellite networks," Proc. IEEE Milcom, pp. 700-704, Vienna, Virginia, October. 2001.

[5] J. Border, M. Kojo, J. Griner, G. Montenegro and Z. Shelby, "Performance enhancing proxies intended to integrate link-related degradations", RFC3135, June 2001.

[6] S. Floyd, T. Henderson, and A. Gurtov, "The NewReno modification to TCP's fast recovery algorithm," RFC 3782, April. 2004.

[7] L. A. Grieco, S. Mascolo,"Performance evaluation and comparison of Westwood+, New Reno and Vegas TCP congestion control", ACM Computer Communication Review, April 2004, Vol. 34(2).

[8] C. Caini and R. Firrincieli, "TCP Hybla: a TCP enhancement for heterogeneous networks," Int. J. Satellite Commun. Netw., vol. 22, no. 5, pp. 547–566, September. 2004.

[9] L. Wu, F. Peng, V.C.M. Leung. "Dynamic Congestion Control for Satellite Networks Employing TCP Performance Enhancement Proxies", Proc. IEEE PIMRC, Barcelona, Spain, Sept. 2004.

[10] D. Katabi, M. Handley, and C. Rohrs, "Congestion control for high bandwidthdelay product networks," ACM SIGCOMM Comput. Commun. Rev., vol. 32, no. 4, pp. 89–102, October. 2002.

[11] K. Zhou, K. L. Yeung, and V. O. K. Li, "P-XCP: a transport layer protocol for satellite IP networks," in Proc. IEEE GLOBECOM, Dallas, TX, Dec. 2004.

[12] D.M. Lopez-Pacheco and E. Lochin. "Optimal Configuration for Satellite PEPs using a Reliable Service on Top of a Routers-Assisted Approach", Proc. IEEE IWSSC, Siena-Tuscany, Italy, Sep. 2009.

[13] Tuan Tran Thai, Dino Martin Lopez Pacheco, Emmanuel Lochin1, Fabrice Arnal, "SatERN: a PEP-less solution for satellite communications", in Proc. IEEE ICC, Kyoto, Japan, June 2011.

[14] E. Rendon-Morales, J. Mata-Diaz, J.J. Alins, J. L. Muñoz, and O. Esparza, "Analysis of TCP variants over a QoS DVB-S2 system", Proc. ACM PE-WASUN, Miami Beach, FL, Oct. 2011.

[15] J. Alins, J. Mata-Díaz, J. L. Muñoz, E. Rendon-Morales, O. Esparza, "XPLIT: A cross-layer architecture for TCP services over DVB-S2/ETSI QoS BSM", Computer Networks, vol. 56, pp. 412-434, Jan. 2012.