Analyze TCP Protocol Performance in Satellite Communications

Eng. Mahmoud Ayoub , Dr. Nabel Dahdouh

Abstract

This research is mainly about the influence of protocols on the performance of Satellite Communication. It is widely recognized that satellite communications are affected by some peculiar problems, which penalize heavily the performance and efficiency of the Transmission Control Protocol (TCP). In fact, wireless satellite channels are usually characterized by link-asymmetry and higher Round Trip Time and Bit Error Rate in comparison to wired links. It means that TCP, which was developed for wired channels, exhibits often poor performance in a satellite scenario (unfair bandwidth allocation, low throughput and long file-transfer delay).We have studied and compared many TCP variants recently proposed. In particular we have compared TCP-Reno standard implementation with TCP-SACK, TCP-Westwood, TCP-Vegas and TCP-Tibet. We tested the different protocols on a simulated satellite scenario, with the support of NS2, a well-know network simulator platform, annotating the advantages and drawbacks of various protocols in order to improve the transmission of IP data packets over satellite channels.

Keywords: Satellite link, Performance analysis, TCP behaviors.

1. INTRODUCTION

The use of satellites to transmit Internet data is a very encouraging proposition compared to wired networks. Modestly, any IP stream needs to pass 17 to 20 nodes, which means it will pass through several bottlenecks. On the other hand, the satellites need one jump to connect two sites far from each other.

Unfortunately, the use of satellites is limited by poor performance obtained from the usual TCP / IP protocols. In fact, delaying the long propagation of the optical channel has the same effect as signal transmission over several nodes, which is affected by the congestion control architecture of communication routing protocols. Moreover, the future communication system requires that the satellite be integrated with the ground [1], so the interaction between them will bring about new performance problems, which in part have not been fully explored or resolved. Satellite communications need to be fully integrated with the physical communication system without any technical barriers if they are to play an important role in the future world of communications, especially if the goal is to have extended coverage that extends to remote users, or to the sea or the air.

The main objective of this research is to evaluate the performance of typical mobile Internet applications using different TCP structures. Although the high efficiency of high-speed space links may eventually require the introduction of a new protocol, studies [2] suggest that the full use of the link can be achieved by introducing some improvements to the TCP protocol that have been approved or are being implemented Currently. The special interests of the satellite environment are performance enhancement, speed of retransmission and recovery, and optional notification of receipt. This research describes the protocols that take these concerns into consideration and the impact of their implementation on increasing the performance of the TCP protocol in space links. The analysis was carried out on typical satellite scenarios.

This research is organized as follows: Paragraph (2) reviews the role of satellites, and paragraph 3 concerns the characteristics of satellites that are likely to have an impact on TCP and other Internet applications. Paragraph (4) describes the different types of TCP protocol detailed and used in the search. In paragraph (5), the experiments conducted were presented, and in paragraph (6) the findings were presented. Paragraph (7) presents the conclusions and the conclusion.

2. THE ROLE OF SATELLITES

Future satellites will play an important role in providing wireless access to mobile users, in order to enhance coverage of existing terrestrial mobile systems, to ensure long-range mobility and high bandwidth. Existing satellite links provide redundant capacity, alternative routes, and less congested wireless and wired systems, thus providing a unique power to improve efficiency and reliability. The nature of satellite satellite broadcasting makes it suitable for multicast operations. In addition, the satellite is the only way to increase the extent of extraterrestrial communication, which connects space stations and space shuttle to the interplanetary network in the future. The growing interest in exploring the new world needs an affordable and affordable communications system for the success of space missions and full control of the Mission. It is possible for satellites to be integrated with short-range wireless networks (Bluetooth) to provide Internet services to mobile vehicles. In this scenario the satellite connects the vehicle to the Internet while the Bluetooth system will be able to connect the equipment inside the vehicle (car, bus, plane, ship).

The main satellite communication systems that offer a large direct packet display rely on users on the geostationary orbital configuration. One GEO can cover a very large area with several narrow beam beams using multi-ray antennas. This way you can get several reliable connections even with small terminals. GEO systems offer high data rate (up to 2Mbps or more) and a total gross capacity (10Gbps). Many of these systems are expected to cover specific areas in a few years, and eventually cover high except the poles.

In previous studies [8-12], TCP / IP performance was addressed through geostationary satellites and focused on evaluating the performance of different TCP structures (Tahoe, Reno, New Reno, SACK), which are connected between two fixed stations through a GEO satellite . In this paper, other TCP structures (Westwood [12], Tibet [7]) were processed, and TCP performance was evaluated when both the satellite and the earth station shared the same bottleneck to determine which protocol would have better behavior when used For long and short call-back communication. In addition, both types of communication were introduced to study UDP flow behavior to assess the possibility of maintaining productivity. A very realistic scenario was created to assess the performance of the satellite environment.

3. SATELLITE CHARACTERISTICS AFFECTING IP APPLICATIONS

Initially, when looking at the performance of real-time applications (such as audio and video), large propagation delays are a critical issue, and this problem is particularly acute for TVP applications over links with a delay-width multiplier. For effective use of the satellite link. A lost satellite link can cause a slow start to repeat with a significant productivity impact. The main characteristics of the satellite network, which must be considered to assess the impact of the satellite on the performance of Internet applications are: delay propagation, domain width delay (DBP), SNR ratio, satellite diversity, and routing strategy.

Satellite systems have changed their role from a bent-pipe model to a channel-oriented, packet-switched channel model. In this process each satellite is an element in the network, to show new design problems in both the network and transport layers.

By taking into account the performance of standard Internet protocols over routes that include satellite links with large propagation delays, the problem is particularly related to the TCP protocol because it is a time sensitive nature. For audio, and other real-time applications, delay is an important issue. Some basic features that need to be taken into account to assess the impact of system architecture on the performance of network protocols are:

Deployment delay

Because of the large packet distance needed to transmit via satellite networks, it can be subject to significant transmission delay (depending on the constellation and routing strategy).

Transmission delay in the GEO structure depends on the user-gateway distance (or user-user if direct dialing is allowed) when the satellite is set up without interference between satellites ISLs, or the communication strategy if ISLs are used. The delay can be considered constant for fixed users, while it is variable in the case of mobile users.

However, due to the queue queuing in the ground router, even fixed users can experience significant fluctuations in propagation delay. This phenomenon is more serious if ISLs are implemented because the routing strategy can also play a very important role.

Delay fluctuations are particularly important for TCP because it causes packet delivery time to be variable, and can affect the RTT estimate through TCP. Whenever a change occurs in RTT it takes some time for the TCP protocol to adapt to the changes. If RTT changes continuously, TCP will not be able to adequately update the RTT estimate, which may cause a resend or a time-out, thus reducing the overall packet width.

Delay-Bandwidth Product (DBP)

TCP performance over satellite connections is also affected by bandwidth-delay charges. This was assessed through simulations and experiments [9-11]. This performance can be estimated using techniques such as selection knowledge, window dimensions [9,11], and other techniques such as TCP spoofing, cascading and cascading can also be used [11].

Satellite diversity

Satellite diversity is a very effective technique for improving connection availability or SNR by using more than one satellite-user connection to establish communication. In an offline system of satellite diversity technology the advantage of increasing the probability of delivery of packets from and to the satellite network, to and from the ground. But this will increase the number of busy channels, which in turn reduces capacity.

Signal to noise ratio SNR

The SNR (or equivalent of the BER error rate) in the GEO group is characterized by large fluctuations due to loss of free space, the spread through the atmosphere (such as rain) of fixed communications, and shadowing in the case of mobile communications. Both the shadow and the deep rain fade caused packet loss.

The widespread SNR was studied for GEO satellites. Since TCP is primarily designed for wired connections, it is assumed that BER is too small (10-10 or less). When the packet is lost, TCP interprets that the network is congested, which causes TCP to implement congestion control algorithms, which reduces overall productivity. This is negative in satellite connections and causes undesirable reduction in productivity**.**

4. TCP PROTOCOL SPECIFICATIONS

TCP is primarily designed for the wired connection, so it relies on certain assumptions about the special properties of the wired connection. In addition, TCP relies on the assumption that the network does not provide any explicit back-up feedback to the source. Therefore, each source must estimate the characteristics of the network such as RTT or bandwidth used to achieve efficient end-to-end control of congestion.

In order to improve TCP connection performance to prevent congestion, and to achieve fair sharing of packet display between different communications, the TCP protocol has been extended with some add-ons that have been implemented in TCP Reno.

Slow Start and Congestion Avoidance

The slow start mechanism was introduced, avoiding congestion in 1998 [13] and was added as a condition for TCP implementation in RFC 1122 [14].

Slow start is used to gradually increase the rate at which the transmitter enters the data within the network. Slow start starts by sending one piece (piece) and waiting for the acknowledgment. With each receipt, the sender receives two parts of data within the network, which results in an exponential increase in the amount of data being sent. The slow start ends when the future announces a window or when a loss or loss is revealed. Given that the amount of time required for the slow start to reach the full bandwidth is dependent on the time to go, and the back, the satellite links are particularly sensitive to the productivity available during the slow start.

The congestion avoidance is used to check the network for the available packet width, by sending one additional piece at each time, and back (until the receivers announce a window). In the slow starting structure / blocking the original congestion When the TCP transponder discovers a lost part, it drops back to the slow start until the packet transmission rate is half the rate at which the loss detector was, and then the congestion phase begins.

Fast Retransmit and Fast Recovery

The TCP Reno protocol also includes fast retransmission and fast recovery (RFC 2001) [15]. Although these mechanisms have been present in many TCP proposals for several years, they have not been documented as RFC standards until 1997. The rapid retransmission reduces the time taken by the TCP transmitter to detect one part loss. Instead of waiting for the RTO retransmission timeout, the TCP sender can resend a portion if it receives three duplicate acknowledgment notifications for a portion was sent immediately before the segment was lost.

Quick restore works in tandem with fast retransmission. As mentioned above when the transmitter returns a part, it usually returns by moving initially to the slow start phase followed by the congestion phase. If the TCP sender detects that a portion of a fast retransmission has been lost, it uses a fast restore instead. The fast recovery divides the transmission rate by half, and begins to avoid congestion directly without retreating to the slow start.

Large windows

The original TCP standard limits the TCP receive window to 65535 B. The size of the TCP window is important in the satellite environment because the maximum TCP connection output is limited by the time to go, and the return [16] as shown in the following relationship:

$$troughput_{\max} = \frac{receive\_buffer\_size}{round\_trip\_time}$$

Without large windows, the TCP connection via a satellite synchronous to Earth is typically limited to the following productivity:

$$troughput_{\max} = \frac{64Kbytes}{580ms} \approx 900Kbps$$

Note that the TCP maximum limit is independent of the channel package width. A TCP connection running more than one full T1 channel (1536000b / s) can still achieve a maximum output of about 900 kb / s with a 64Kbyte reception window. According to RFC 1323 [13], large windows allow higher beam widths via large delay channels such as those used in satellite links.

In addition to these functions in order to improve native TCP, there are TCP protocol developments proposed, and their application to different TCP protocols. Thus to evaluate TCP performance over satellite connections, the original TCP with TCP Reno can be compared with other TCP protocols when the same simulation scenario. Other TCP protocols have the following characteristics:

TCP SACK

This protocol has enabled TCP to use SACK. This aspect is very important in the satellite scenario because cumulative acknowledgment, applied by TCP is not suitable for the satellite environment with long delays due to the time it takes to obtain information about the loss of the part. Using the selective acknowledgment mechanism SACK and defined by RFC 2018 [4]. SACK generates an explicit message to the sender of the parts that have arrived, none of which has been lost. This gives the sender more information about the parts, which require re-sending, and the parts that do not need to be transmitted through a long delay link such as satellite links .

TCP Newreno

This protocol applies some modifications to the REST algorithm to improve the partial acknowledgment response, initially proposed by Janey Hoe [6]. In the original application of the Fast Restore algorithm, the TCP data sender resends the packet only after the retransmission timeout occurs, or after the receipt of three acknowledgment notifications, resulting in the start of the retransmission algorithm. A single retransmission time may result in the re-sending of several data packets, but re-calling the RPC algorithm results in a single data packet being sent back. TCP Newreno defines a "quick restore procedure" that begins when three notifications are received by repeated acknowledgment and ends either when the retransmission timeout occurs or when a acknowledgment message arrives informing all of the data, including data that was pending when a fast restore procedure was initiated. This way you can retrieve the case where several packets fall from one data window and faster than TCP-Reno.

TCP Vegas

In order to increase productivity and reduce packet loss, TCP Vegas offers three technologies [18]. The first is more retransmission of the lost parts. This is obtained by using a more accurate RTT guess. Ie by reading the system clock signal every time a portion is sent, and again when the ACK arrives. When a non-duplicate ACK is received after retransmission, Vegas again checks to see if the interval is greater than the timeout timeout value. If this is done, Vegas anticipates the retransmission of that packet. In addition, Vegas reduces the busy frame only if the retransmission portion was previously sent after the last reduction. In fact, the loss that occurred before the last frame decreases does not mean that the network is congested for the size of the congestion frame.

The second technique is to expect congestion through TCP and adjust the transmission rate accordingly. Using Vegas in case of congestion avoidance the frame size increases each RTT per part, and the completed productivity compares with the part obtained when the frame is smaller within one part. If the difference is half of the productivity achieved when it is only one part, the window is reduced by one part.

The third technique is to adjust the slow start mechanism so that packet loss is avoided while trying to find the available packet width. This is obtained by allowing the exponential increase only for another RTT while the center does not change the congestion window.

TCP Westwood

This protocol [12] distinguishes that it can only be implemented at TCP connection sources, and when different SACKs are not needed worldwide to operate correctly. The guessing of the available packet width is done by measuring the acknowledgment rate. This guessing is used to determine the speed-recovery mechanism parameter in order to avoid half of the blind transmission rate as in TCP Reno after packet loss. Resetting the frame To accommodate the available packet width, TCP Westwood makes it more stringent for the sporadic losses caused by wireless channel problems. This causes expensive TCP to overwork, reducing the unnecessary frame. Guess the obvious bandwidth has a profound impact on TCP performance, especially in the presence of randomized, sporadic losses, and high-latency bandwidths typical of satellites.

TCP Tibet

This protocol requires some modifications to the TCP Westwood working mechanism to guess the available bandwidth. In particular, the technology offers double filtering to obtain a true guess for the packet width used by TCP sources without the error provided by sporadic losses. This estimation is achieved by measuring and filtering the low pass length of packets, and the intervals between transmission periods. This is due to the fact that dual filtering technology has a beam-guessing mechanism that is biased to the ACK compression theory, which usually occurs when the packets are transmitted over long delay links, and the bandwidth guesses are obtained around a fair share value when all TCP sources are exposed to the same conditions Track almost.

5. SIMULATION SCENARIO

The network simulator NS-2 [19] was used to form a scenario in which the performance of the various protocols mentioned above could be tested. In order to differentiate our work from other comparisons similar to different TCP protocols in special scenarios [2,3,8-12], the effect of interactions between short-terrestrial and short-distance communications was emphasized. When competing for a neck tie one bottle. The graphology was used in Figure (1).

The satellite link was used to transmit data between two ground stations located in Palermo, Houston and the satellite on the Atlantic Ocean at a length of 34.5 degrees west. This link has 20Mb / s channels for the upstream download, and the downstream download from the satellite.

Each source node, "Scr A" and "Scr B", establishes 15 FTP connections to the "Dest S" target for the satellite connection, and the target "Dest T" for terrestrial connection The bottleneck is between the routers and has a capacity of 2Mb / S, and delay propagation of the 10 ms link. All other terrestrial connections have a 20Mb / s link capacity, and delay propagation of the 10 ms link.

The simulation was run for 6000 seconds, with three cases affecting how the bottleneck capacity is divided between connections. More detailed: the first time at the second 0 start with 15 FTP connection towards the target of the space "Dest S". After 2000 seconds, we start with 15 FTP connections to the target "Dest D", and finally after 4000 seconds we start with a User Datagram Protocol (UDP) towards the terrestrial target. In this way, we can evaluate the performance of each TCP protocol in a wide range of situations: the first when there is only a satellite connection to the bottleneck, then there is a satellite connection, grounding the same bottleneck, and finally with UDP streaming, , And space.

In addition, different satellite and terrestrial communication protocols were used to estimate system behavior when new satellite protocols were exposed to the impact of older, non-wireless TCP protocols widely deployed in actual IP systems.

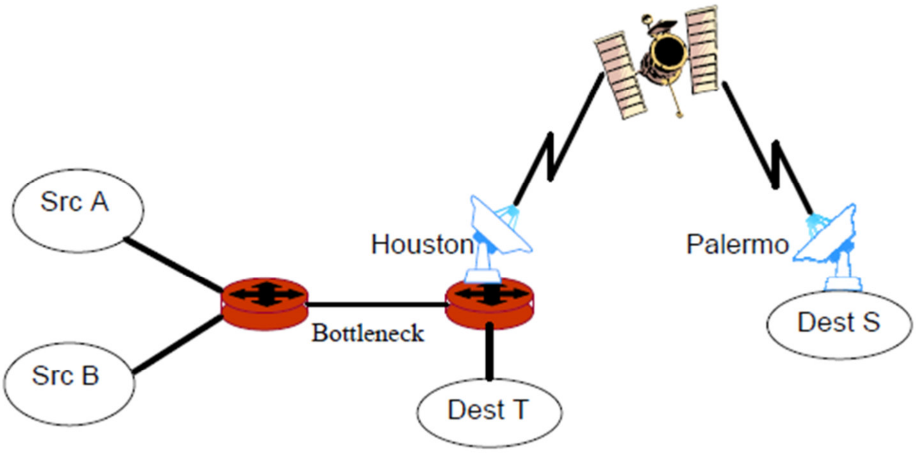

Figure (1): Description of the simulation scenario, the link of the directed bottle reaches 2Mb / s transfer speed.

6. PRACTICAL RESULTS

In this section, simulation results were presented based on the previous scenario. The formats were obtained from (2) to (8) assuming that all sources use the same version of the TCP protocol, whereas in (9) to (11), a comparison was made between the best protocols assigned to the satellite link and the most common TCP protocol In the actual distribution of IP network devices.

By comparing the results obtained during the first part of the simulation we can prove that during this period, which only has satellite connections that all the TCP protocols offer a very good connection and close to 100% of the bottleneck capacity, reaching the maximum limit for Vegas, Westwood, Tibet. This is very important for Westwood, Tibet, which has been studied for satellite communication because it demonstrates that TCP connections are capable of using the network efficiently when using the correct protocols.

During the second period of simulation (between 2000 to 4000 seconds) satellites suffer from the existence of ground connections which have short RTT. In this case, the SACK, Westwood, Tibet protocols are only capable of serving part of the bottleneck of the satellite link. In this scenario, TCP Tibet has a special feature because it has the ability to adjust the estimated packet width to the previous value after entering the ground link after a small period of time to maintain the use of space links, to stay connected to the maximum. Good performance is also achieved through TCP Westwood, which tried to distribute the bandwidth in a fair way between terrestrial communications and satellites, although satellite communications received a fraction more.

At the end of the third simulation period (4000 to 6000 seconds), UDP streaming in most cases achieves some advantages for satellite links. In fact, in all cases where satellite connections were thwarted through terrestrial connections, the UDP flow eliminated the channel's use of terrestrial TCP connections, and assisted satellite links to obtain some parts of the bottleneck. Even in this case, the best performance of the satellite links is obtained through the Westwood, Tibet protocols, which reach the state of total blockage of terrestrial communication for the service of those satellites.

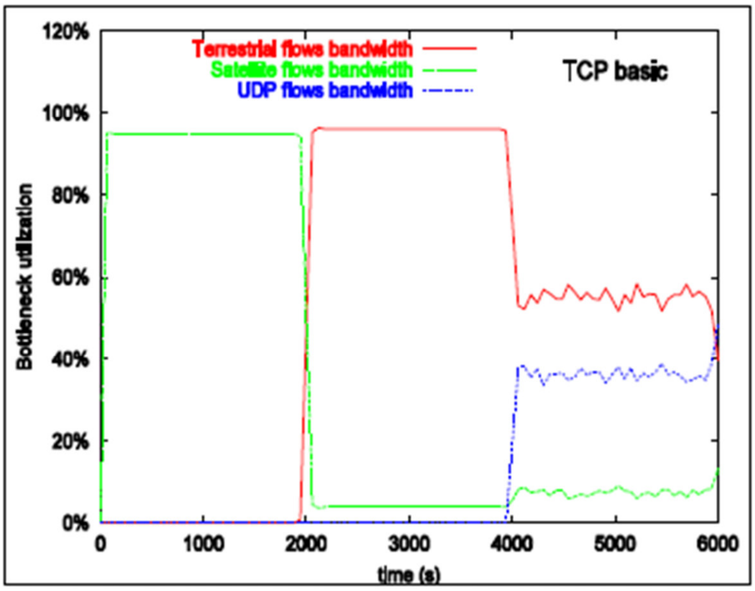

Figure 2: Simulation results of the original TCP for both space and terrestrial communications.

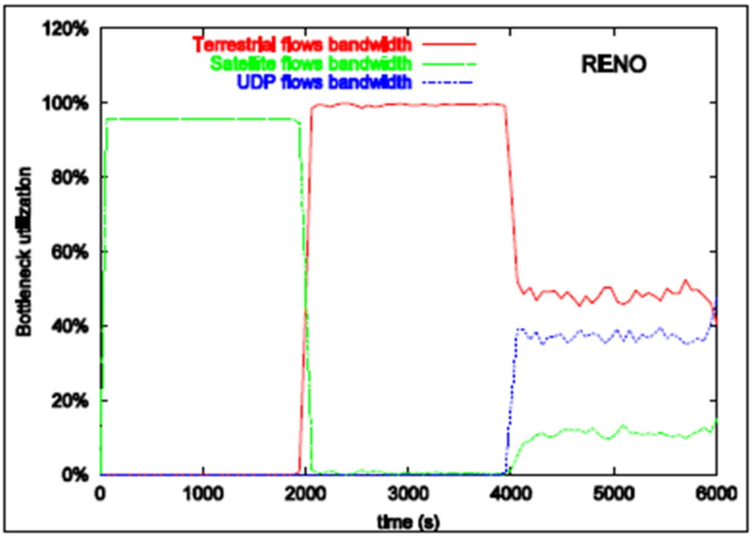

Figure 3: Simulation results of the original TCP Reno for both space and terrestrial communications.

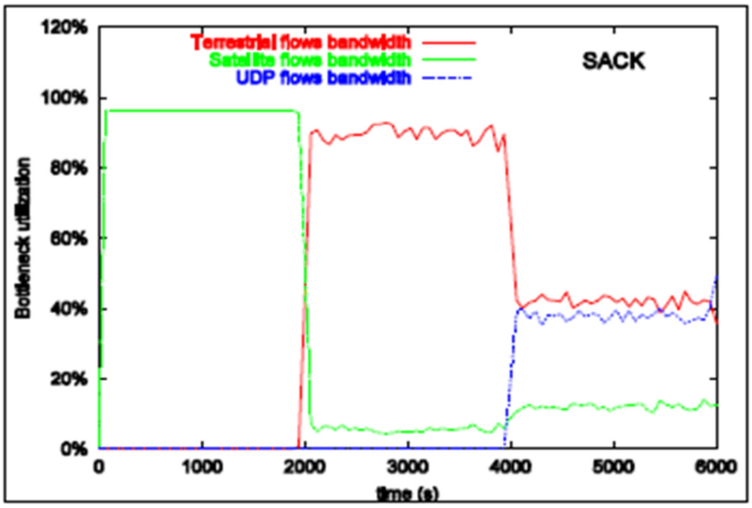

Figure 4: Simulation results of the original TCP SACK for both space and terrestrial communications

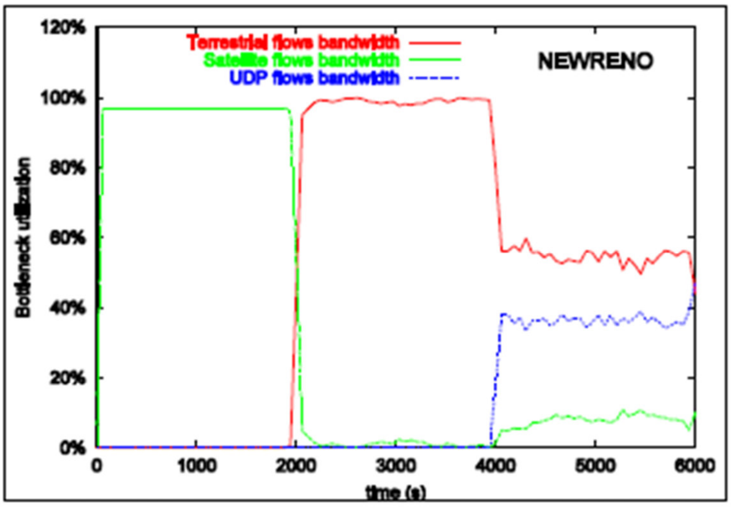

Figure 5: Results of the original TCP Rewreno simulation for both space and terrestrial communications.

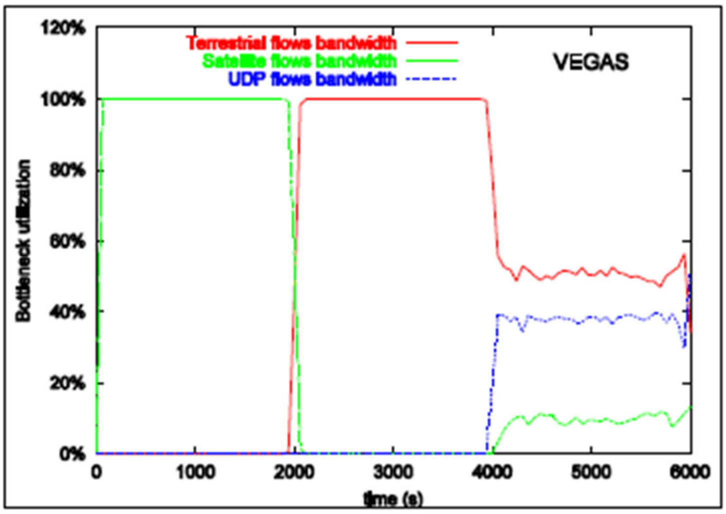

Figure 6: Simulation results of the original TCP Vegas protocol for both space and terrestrial communications.

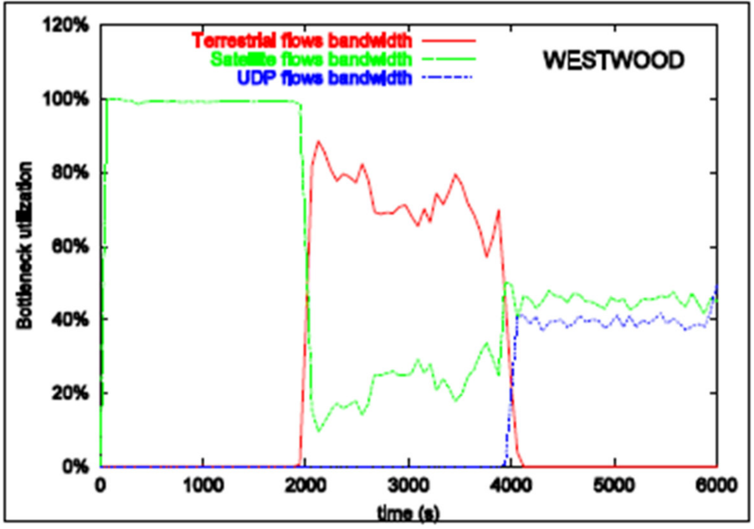

Figure (7): Simulation results of the original TCP Westwood protocol for both space and terrestrial communications.

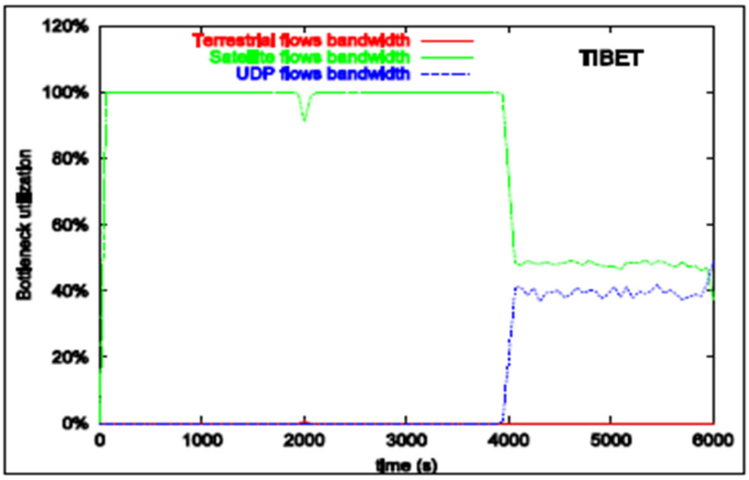

Figure 8: Simulation results of the original TCP Tibet for both space and terrestrial communications.

Therefore, after obtaining the first results, a comparison was made between the Tibet protocol, which proved to be the best protocol to protect the bandwidth for satellite communications, and the most widely used TCP version of the network devices: TCP Reno. The following formats illustrate the results reached for the mixed TCP scenario that uses the TCP Tibet satellite connection, and the terrestrial connection uses the TCP Reno protocol.

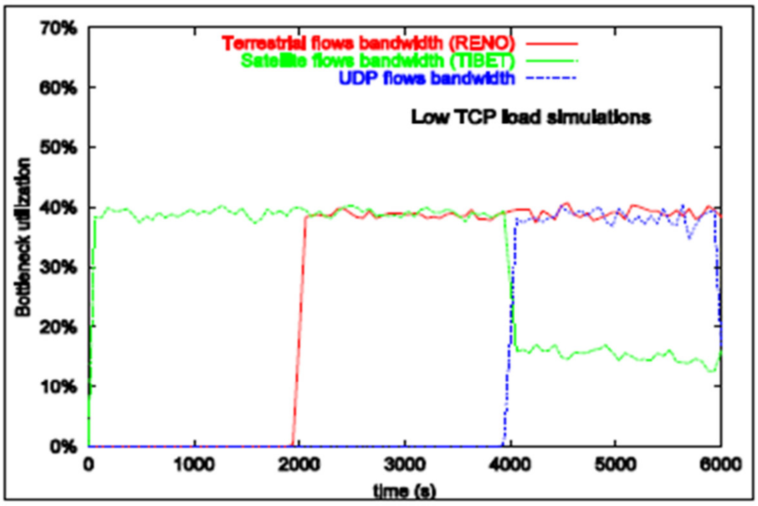

Figure 9: Simulation results with TCP Tibet satellite connection and terrestrial TCP Reno connection in low load working conditions (about 40% for each pattern).

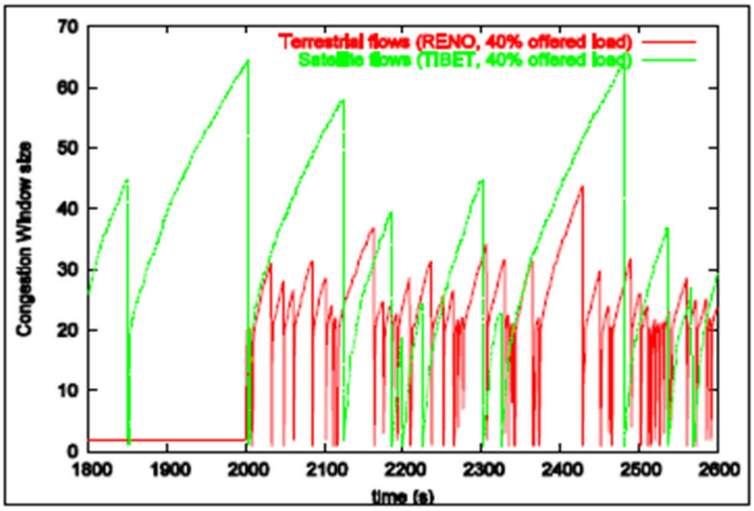

Figure 10: Simulation results for TCP congestion windows in low load working conditions.

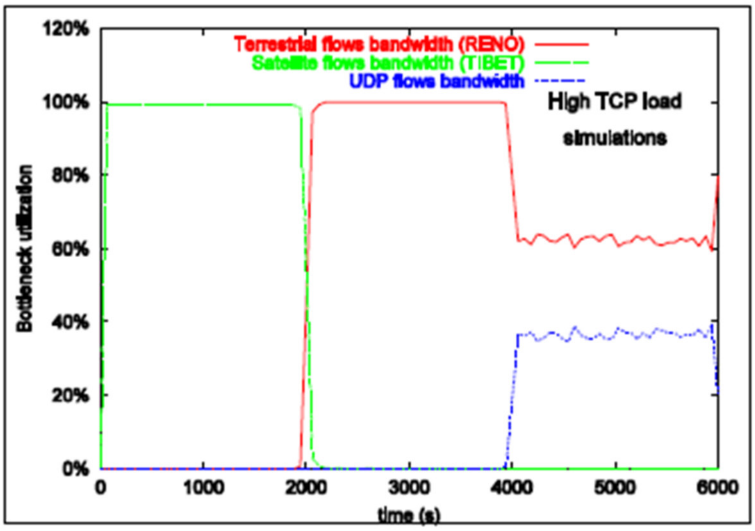

Figure 11: Simulation results with TCP Tibet satellite connection and terrestrial TCP Reno connection in high load working conditions.

7. –CONCLUSION

Satellites have a very important role in the future because of their ability to cover large areas, and in some cases are the only way to expand communications beyond the borders of the earth. Currently, satellite communications are affected by poor performance partly due to a series of insufficient protocols used in TCP / IP.

In this research, a comparison was made of some of the proposed extended TCP protocols, and their impact on the performance of satellite links, whose link capacity with other terrestrial connections must compete, was evaluated. The simulation results show that it is possible to introduce some mechanisms to improve satellite productivity such as fast recovery and guess the available bandwidth as described in TCP Tibet, TCP Vegas.

Unfortunately, simulations performed on a mixed scenario where different TCP protocols work on sharing the same link, it was found that - including the TCP Tibet protocol that provided the best results for a homogeneous scenario for satellite links - it was not able to compete with the previously completed TCP protocol for capacity Satellite connection connection.

In conclusion, this work provided evidence of actual TCP protocols developed as inappropriate for satellite scenarios. In addition, we suggest that further research is needed to access a new version of TCP capable of compensating long RTT for improved productivity and obtaining fair behavior of TCP protocols for the previous version before satellite use becomes a daily task.

References

1. D. O'Mahony, *UMTS: the fusion of fixed and mobile networking,* IEEE Internet Computing, vol. 21, Jan.-Feb. 2009.

2. M. Allman, C. Hayes, H. Kruse, S. Ostermann, *TCP Performance over Satellite Links*, Ohio University

3. P. Loreti, M.Luglio, R. Kapoor, J. Stepahnek, M. Gerla, F. Vatalaro, and M. A. Vazquez-Castro*, Satellite Systems Performance with TCP-IP Applications*, IWDC 2010, Taormina, September 2010.

4. S. Floyd, M. Mathis, J.Mahadavi, and A.Romanov, *TCP Selective Acknoledgement Option,* RFC 2018, April 1996

5. S. Floyd, and T.R.Henderson, *The NewReno Modifications to TCP's Fast Recovery Algorithm,* IETF RFC 2582, April 2005

6. J.C: Hoe, *Improving the Start-up Behaviour of a congestion Control Scheme for TCP,* ACM SIGCOMM Computer communications Review, October 2005.

7. A. Capone, and F. Martignon, *Bandwidth Estimates in the TCP Congestion Control Scheme,* IWDC 2001, Taormina, September 2001.

8. W. D. Ivancic, D. Brooks, B. Frantz, D. Hoder, D. Shell, D. Beering, *NASA's Broadband satellite networking research*, IEEE CommunicationsMagazine, n. 7, July 1999, pp. 40-47.

9. C. P. Charalambous, V. S. Frost, J. B. Evans, *Performance evaluation of TCP extensions on ATM over high bandwidth delay product networks,* IEEE Communications Magazine, n. 7, July 1999, pp. 57-63.

10. H. Kruse, S. Ostermann, M. Allman, *On the Performance of TCP-based Data transfers on a faded Ka-Band Satellite Link,* Proceedings of the
6th Ka-Band Utilization Conference, June 2000. 11. M. Allman, H. Kruse, S. Ostermann, *A History of the*

Improvement of Internet Protocols Over Satellites Using ACTS, Invited paper for ACTS Conference 2000, May 2000.
12. C. Casetti, M. Gerla, S. S. Lee, S. Mascolo, M. Sanadidi, *TCP with Faster Recovery,* UCLA CSTechnical Report #200017, 2000
13. V. Jacobson, *Congestion Avoidance and Control*, In Proceedings ACM SIG-COMM '88, ACM, August 2003
14. R. Braden, *Requirements for Internet Hosts- Communication Layers*, October 2004, RFC 1122 15. W. Stevens, "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms, January 1997, RFC 2001
16. J. Postel, *Transmission Control Protocol*, September 1981, RFC 793
17. V. Jacobson, R. Braden, and D. Borman, *TCP Extensions for High Performance*, May 2007, RFC 1323
18. L. S. Brakmo, S. W. O'Malley, L. L. Peterson, *TCP Vegas: New Techniques for Congestion Detection and Avoidance*, 2009 ACM SIGCOMM Conference, pages 24--35, May 2009.
19. "Network Simulator (NS-2)", www.isi.edu/nsnam/ns/.