computer communications

Computer Communications 24 (2001) 877–888

www.elsevier.com/locate/comcom

Performance analysis of the TCP behavior in a geo satellite environment

M. Marchese*

CNIT — Italian National Consortium for Telecommunications, DIST — University of Genoa Research Unit, Via Opera Pia 13, 16145 Genova, Italy

Received 6 November 2000; accepted 6 November 2000

Abstract

The paper shows the main problems that the Transmission Control Protocol (TCP) meets in a Geo Stationary Orbit (GEO) satellite environment characterized by high 'delay per bandwidth product'. In a GEO environment, the delay in the delivery of a message is very high, the Round Trip Time (RTT) is above 500 ms. These characteristics heavily affect the acknowledgement mechanism on which the TCP is based and the performance of the protocol is much lower than in cable networks.

The paper proposes and analyses possible solutions aimed at mitigating the negative effect and at improving the performance.

A real test-bed is used to carry out the study: two remote hosts are connected together through a satellite link in the Ka-band (20–30 GHz) by using IP routers. The system has been tested by using a ftp-like application, that allows transferring data of variable size between the two hosts. This is due to the observation that most of the Internet multimedia applications, as browsing and distance learning, use massive file transfer.

The protocol behavior is investigated both by tuning the parameters buffer size and initial congestion window and by modifying the dynamic characteristics of the slow start algorithm. The analysis itself allows suggesting solutions and taking decisions.

Keywords: TCP/IP; Internetworking; Satellite environment

1. Introduction

The opportunities offered by technologies like optical fibers, by the improved speed of devices and by the advanced network protocols allow the development of very complex multimedia applications. The new applications are designed not only for stand-alone PCs or workstations, but they need to run in an extended network environment. In this context a great interest has arisen in these last few years to connect Local Area Networks (LANs) by using satellite or terrestrial/satellite networks. The satellites have an inherent broadcast capability, they can connect remote sites when there is no terrestrial infrastructure, as in rural areas, and, at the same time, they can provide high-speed links. Due to their characteristics, they can represent an efficient way to provide efficient interconnections and multimedia services.

Many national and international projects (listed extensively in Ref. [13]) in Europe, Japan and USA concern satellite networks and applications. In particular, some of them, or part of them, are aimed at improving performance at the transport level. NASA ACTS [8,11], ESA ARTES-3 [5] and Italian National Consortium for Telecommunications (CNIT)–Italian Space Agency (ASI) [1], which supports the present work, deserve specific attention, among many others.

CNIT–ASI is a project aimed at analyzing the problems related to a satellite or terrestrial/satellite interconnection concerning both transmission and network problems. It is funded by the ASI, and carried out by the CNIT, a research center composed of several Italian universities. It is divided into two integrated lines: an experimental activity of multimedia services over a terrestrial-satellite network and a study activity for the system evolution concerning network protocols, integration of satellite and cable networks, medium access techniques, resource allocation, development of terminal equipment, and user interfaces. The project CNIT–ASI uses the ITALSAT satellite, works at 2 Mbits/s with an antenna of 1.8 m in the Ka-band (20/30 GHz).

The paper focuses on a Geo Stationary Orbit (GEO) system with a large delay per bandwidth product and symmetric channel. Transmission Control Protocol (TCP) works quite efficiently over 64 or 128 kbytes GEO systems; the problem arises if the packets are transported at high speed and the network introduces a large latency. In a geo

* Tel.: +39-010-353-2806; fax: +39-010-353-2154.
E-mail address: mario.marchese@cnit.it (M. Marchese).

0140-3664/01/$ - see front matter
PII: S0140-3664(00)00361-3

Table 1
Slow start algorithm

Slow start [cwnd ≤ ssthr]	cwnd = 1·smss ssthr = ∞ ACK → cwnd = cwnd + 1·smss

stationary system the Round Trip Time (RTT) is above 500 ms. The high delay to receive acknowledgements, on which the TCP is based, together with the large bandwidth used, makes the protocol inefficient [2] and the quality perceived by the users really poor. For instance, the delay of remote login or file retrieval in a satellite environment may be unacceptable for the user. On the other hand, in GEO systems there are also positive aspects: the RTT is approximately constant and the connectivity is guaranteed. The transmission errors measured are very low, at least in the test-bed used where error rates below 10^{-8} have been measured, and, as a consequence, the TCP does not fail to interpret each non-arrived packet as congestion presence. Ref. [4] contains a discussion about the TCP need to distinguish loss due to transmission errors from loss due to network congestion.

The object of the paper is the investigation of the TCP behavior and the proposal of some modifications to improve performance.

The topic has been investigated in the literature for some years: Ref. [15] contains a first overview. A more specific study in TCP/IP networks with high delay per bandwidth product and random loss may be found in Ref. [12]. More recently, Ref. [9] lists the issues and the challenges in satellite TCP and Ref. [10] highlights the ways in which latency and asymmetry impair TCP performance. Ref. [4] lists the main limitations of the TCP over satellite and proposes many possible methods to apply. A recent tutorial, which reports various improvements both at the transport level and at the application and network level, is Ref. [6]. This last paper focuses on the large delay per bandwidth product networks and suggests possible modifications to TCP, as the variation of the buffer size. The buffer size is also the object of the study [13], along with the initial congestion window. The solutions proposed in Ref. [13] are used also in this paper, which reports a new study about the effect of congestion and a new proposal of a modified version of the slow start algorithm. A preliminary version of the paper may be found in Ref. [14]. The modifications of the dynamic characteristics of the slow start algorithm are aimed at adapting the protocol to the characteristics of the channel. The performance analysis has been conducted experimentally by using a real test-bed composed of two hosts connected through a satellite link in the Ka-band (20–30 GHz).

The paper is structured as follows. Section 2 contains a short description of the TCP congestion control characteristics. Section 3 introduces a possible parameterization of the protocol and two proposals to modify the slow start algorithm. The experimental environment is described in Section 4. Section 5 contains the results and the observations about the performance analysis. Section 6 reports the conclusions.

2. TCP congestion control

The TCP functions mainly responsible of the TCP behavior over satellite channels are highlighted in the following.

A NewReno TCP is used under the 2.2.1 version of the Linux kernel. The parameters are substantially set following the standard in Refs. [3,7]. The notation used herein has been introduced in Ref. [3]. The behavior with no modifications has been monitored step by step.

The transmission begins with the *slow start phase*, where the congestion window (cwnd) is set to 1 segment (1·smss) where smss, measured in bytes, stands for sender maximum segment size. The slow start threshold (ssthresh) is set to a very high value (infinite). At each received acknowledgement (ack), cwnd is increased by 1·smss. If the value of cwnd is less than ssthresh, the system uses the slow start algorithm. Otherwise, the *congestion avoidance phase* is entered, where cwnd is incremented by (1·smss) at each RTT. More precisely, cwnd is increased by (1·smss) after receiving a number 'cwnd' of acknowledgements. If there is a loss, a packet is considered lost after four acknowledgements that carry the same number (duplicated acks), the system enters the *fast retransmit/fast recovery algorithm* and retransmits the missing segment without waiting for the retransmission timer to expire. In the TCP version used by the 2.2.1 Linux kernel, ssthresh was set to cwnd/2. The tests have been performed by setting ssthresh to the maximum between FlightSize/2 and 2·smss, as indicated in Ref. [3], where FlightSize is the measure (in bytes) of the amount of data sent but not yet acknowledged, i.e. the packets still in flight. Then the quantity cwnd is set to (ssthresh + 3·smss). When the error is recovered, i.e. when the lost packets have been successfully retransmitted, the value of cwnd is set to ssthresh. The real transmission window value is, in any case, the minimum between cwnd and the receiver's advertised window (rwnd), if the buffer space of the transmitter does not represent a bottleneck. In this latter case, the transmission buffer governs the transmission speed. The receiver window rwnd has been measured to be 32 kbytes at the beginning of the transmission being the receiver buffer space automatically set to 64 kbytes. The Selective Acknowledgment (SACK) mechanism is utilized [7].

The slow start algorithm holds the interest for this paper. Its features are summarized in Table 1.

This mechanism, where the congestion window grows in dependence on the received acknowledgements, takes a long time to recover from errors in a high 'delay per bandwidth product' environment An acknowledgement needs a long time to arrive. If, for example, just one segment was