Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Computer Networks 47 (2005) 307–326
COMPUTER NETWORKS
www.elsevier.com/locate/comnet

Performance analysis of AIMD mechanisms over a multi-state Markovian path

Eitan Altman [a], Konstantin Avrachenkov [a], Chadi Barakat [a], Parijat Dube [b,*]

[a] *INRIA Sophia Antipolis, 2004, Route des Lucioles, 06902 Sophia Antipolis, France*
[b] *IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA*

Received 24 June 2003; received in revised form 16 June 2004; accepted 26 July 2004
Available online 11 September 2004

Responsible Editor: S. Palazzo

Abstract

We analyze the performance of an Additive Increase Multiplicative Decrease (AIMD)-like flow control mechanism. The transmission rate is considered to increase linearly in time until the receipt of a congestion notification, when the transmission rate is multiplicatively decreased. AIMD captures the steady state behavior of TCP in the absence of time-outs and in the absence of maximum window size limitation. We introduce a general fluid model based on a multi-state Markov chain for the moments at which the congestion is detected. With this model, we are able to account for correlation and burstiness in congestion moments. Furthermore, we specify several simple versions of our general model and then we identify their parameters from real TCP traces.

Keywords: AIMD; TCP; Multi-state Markov chain; Laplace Stieltjes transform; Performance analysis; Experimentation

1. Introduction

We present a framework to study the performance of Additive Increase Multiplicative Decrease (AIMD) type flow control mechanisms. This is the kind of control used by TCP, the widely used transport protocol of the Internet [26]. However, we anticipate that our results will also be applicable for other flow control mechanisms (e.g., the ABR mechanism in ATM networks). We employ a fluid approach [1,2,4–6] to model the controlled flow. Our model studies a general window-based fluid AIMD mechanism. Our model applies

* Corresponding author.
E-mail addresses: altman@sophia.inria.fr (E. Altman), k.avrachenkov@sophia.inria.fr (K. Avrachenkov), cbarakat@sophia.inria.fr (C. Barakat), pdube@us.ibm.com (P. Dube).

1389-1286/$ - see front matter
doi:10.1016/j.comnet.2004.07.014

to the TCP protocol when the window size is large enough so that the packet nature of TCP is effectively diluted. The transmission rate of the source is assumed to grow linearly at a rate α. In the case of TCP where the flow is controlled via a congestion window, the transmission rate at any instant is equal to the window size divided by the Round Trip Time (RTT) of the connection. The growth of the transmission rate continues until the source receives a notification of congestion from the network or until the maximum window size is reached. In the case of TCP, the congestion is inferred from the loss of packets. It is an implicit notification compared to the explicit notification used by other flow control protocols as the ABR service in ATM or the ECN proposal in the Internet. We call the moment at which the source reduces its transmission rate a loss event. Upon detection of a loss, the transmission rate is *scaled down* by a (possibly random) factor $a \in [0, 1]$. The scaling factor depends on many factors. In the case of TCP, it depends on the version used, on the number of packet losses in the congestion period and on the way by which the loss is detected (e.g., duplicate ACKs or Timeout [26]). The Reno version of TCP divides its window by two for every packet loss [12]. The Newreno and SACK versions do not divide their windows by more than two in a RTT, regardless of the number of packet losses during the RTT [12].

We adopt an end-to-end approach for modeling the AIMD congestion control mechanism [8]. The end-to-end approach considers the network as a black box whose output is the process of loss events. The physical characteristics of the network (topology, capacity, etc.) and the parameters of the traffic of other users are all summarized by the process of loss events that we shall consider in our analysis. The opposite of the end-to-end approach is the network specific approach [8] which considers directly the characteristics of the network when modeling the AIMD type protocols (e.g., [3,10,21] for TCP). The advantage of the end-to-end approach is that it can be applied to all networks resulting in the same loss process as that considered by the model. It is clear that the more general the loss process is, the larger the number of networks the model is able to cover.

The process of loss events can be seen as a point process, where the appearance of a point corresponds to the appearance of a congestion signal, interpreted as a loss in the context of TCP, causing a reduction in the transmission rate. Different models have been proposed to study the performance of an AIMD mechanism using the end-to-end approach [6,11,23,20], but these models make in general simple assumptions on the loss process, as periodic, Poisson, iid, etc. These assumptions may not hold on some Internet paths where losses are clustered or when the rate of the loss process changes following some underlying Markov chain. Our aim in this paper is to consider such paths. For example in Fig. 1, one can observe a scenario where the moments of transmission rate reduction are clustered together. This figure corresponds to the window size evolution of a New Reno [12] TCP connection running between two sites at the technology park Sophia Antipolis in south of France.

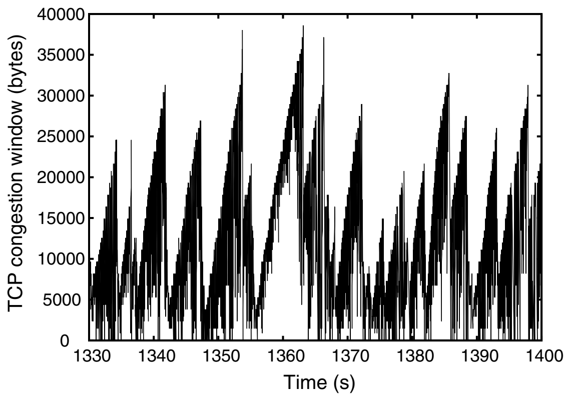

Fig. 1. TCP window evolution.