Available online at www.sciencedirect.com

SCIENCE DIRECT®

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 timeouts 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 © 2004 Elsevier B.V. All rights reserved.
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.

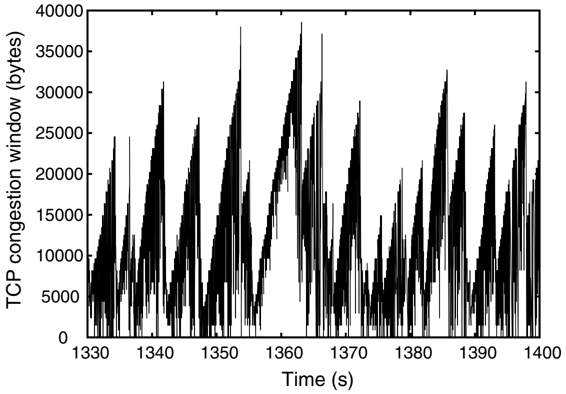

Fig. 1. TCP window evolution.

Here we would like to explicitly mention that *our model provides a framework for analyzing AIMD type flow control mechanisms in general and in particular the AIMD behavior of TCP congestion avoidance phase.* We are *not* claiming to model other features of TCP protocol. In our analysis we only take the loss events from the real TCP connection, and shall employ this real loss process to construct our AIMD mechanism against which we compare our model.

Remark 1. Note that Fig. 1 shows, in addition to the slow oscillations in the congestion window caused by the congestion control algorithms of TCP, some quick oscillations. These quick oscillations are caused by the burstiness of TCP. Indeed, we measure the congestion window as the number of packets transmitted by the source and not yet acknowledged (packets in the pipe). But, this number presents quick oscillations due to the arrival in bursts of ACKs and consequently, the transmission in bursts of packets. These quick oscillations can be absorbed by modifying the operating system and reading the content of the variable describing the congestion window of TCP.

Normally in the analysis of AIMD mechanisms it is assumed that the window is divided by a constant factor upon congestion detection (e.g., by a factor of two in the case of TCP), but we see in Fig. 1 a more severe reduction due to multiple consecutive divisions of the congestion window by two. When a congestion appears, the network continues dropping packets over multiple round-trip times resulting in these multiple consecutive divisions of the congestion window. In a previous paper [1], we presented a two-state Markovian model to account for such a burstiness of losses. In that paper, we considered a lossy path with two states *Good* and *Bad* together with potential loss events. The transmission rate may be reduced upon potential losses. A potential loss can transform into a real loss with probability p_G in the Good state and with probability p_B in the Bad state ($p_G \leqslant p_B$). The time between potential loss events is assumed to be independently and identically distributed. Our main contribution in [1] was to show that the throughput of the flow control mechanism increases with the increase in burstiness of losses when keeping the average loss rate unchanged. We validated the model via simulations, but did not provide any algorithm for the identification of its parameters from real traces.

We would like to highlight here that by burstiness of losses, we mean loss events that appear close to each other, and not packets that are lost in bursts. It is known that TCP congestion control suffers when packets are lost in bursts [12]. A loss event in our model can be the result of one or more packet losses. It is an event that results in a reduction of the congestion window. The factor by which the window is reduced can model the number of lost events due to network congestion (or due to some other phenomenon, e.g., noise in wireless networks).

The present work is an extension of our previous work [1] to a multi-state Markovian case. Being motivated by some experimental results (e.g., Fig. 1), we allow the path of the connection to be described by more than two states. The need for more than two states for describing the path is also motivated by modeling results from [25,28] on mobile satellite channels, where it was shown that one needs typically at least four states. In [1], the scaling factor a is a random variable equal to either 0.5 (the potential loss becomes a real loss) or 1 (a potential loss is not transformed into a real loss). Here, we propose to study the scaling factor with a general distribution that depends on the state of the path. We present then some applications of our general model. These applications can be seen as different ways to infer the parameters of the general model from a real TCP trace. In particular, we provide a method for the parameter identification of our (two-state) model in [1]. A comparison among the different applications is provided to see which one is the most efficient in predicting TCP performance.

In the following section, we overview related works on AIMD modeling in general and on TCP modeling in particular. In Section 3, we present our general multi-state multi-reduction model for the AIMD mechanism. This general model is analyzed in Section 4. In Section 5, we provide several particular cases of the general model as well as their application to TCP modeling. We finally conclude in Section 6.

2. Related works

AIMD mechanism portrays the behavior of a TCP connection in its steady state and in the absence of timeouts and limitation on the throughput caused by the receiver window. Thus, the literature on TCP modeling overlaps with the literature on AIMD modeling. AIMD modeling approach (to which the current paper belongs) normally assumes an exogenous packet loss process, thus neglecting the impact of a single TCP connection on the loss process. This describes the situation of an interaction with a very large number of other connections. One of the most important papers [23,24] makes assumption of independence between losses of packets and obtains explicit expressions for the throughput of TCP as a function of the loss probability taking into account also time-outs, and provides an extensive validation through experimentation (through mainly very long TCP connections). Rather than providing an exhaustive list of previous works on TCP modeling we shall discuss works which deals with fluid models for AIMD mechanism, in the spirit of the modeling approach followed in this paper. Interesting fluid models that can be seen as limits of the independent packet loss process are given in [11,22]. In [4,5] and references therein, related modeling approaches have been used with the assumptions of independent times between losses, focusing however, on non-linearities in the window growth dynamics of TCP (which is induced by queueing delays and the receiver window size limitation). In [2] we present a general model that accounts for any stationary ergodic process of loss events. No Markovian assumptions are made. The advantage of using Markov chains is that it permits us to reduce the number of parameters to infer from real traces. Moreover, a Markovian model is easy to map to some networks whose loss processes are known to possess some Markovian properties (wireless and satellite networks [25,28]).

A second modeling approach for AIMD consists of predicting simultaneously the loss process together with the throughput. To do that, a precise knowledge of the network topology and the number of ongoing connections (or their distribution) is required. The throughputs of all connections are obtained using some fixed-point approach. This technique has been used in [21], where the authors propose a model for a network of Active Queue Management routers, and further used in [3].

An abstraction of AIMD protocols, based on deterministic fluid models, especially well adapted to ECN marking, has been introduced recently. This approach is mainly due to F. Kelly (see [15] and references therein). This, as well as other related models, have been extensively investigated by many researchers, see e.g. [18,16,27], and have been related to fairness issues, to pricing and to utility optimization. Further, interested readers can find a comprehensive survey on packet and fluid level modeling and analysis of TCP in [8].

3. The model

Let $X(t)$ be the transmission rate at time t. $X(t)$ is equal to the current window size divided by the Round Trip Time of the connection. Let $K = \{1, 2, \ldots, N\}$ be the set of possible states of the path. We allow losses to occur in any of the N states; the probability of the occurrence of losses in each of these states may be different. To that end, we define a series of potential losses occurring with a certain distribution of times between potential losses. Let T_n denote the time at which the nth potential loss occurs and let X_n denote the transmission rate of AIMD just prior to T_n. The pair $\{T_n, X_n\}$ can be considered as a marked point process [7]. Let D_n, $n \in \mathbb{Z}$ be a sequence of times between potential losses: $D_n = T_{n+1} - T_n$. D_n are assumed to be iid with expectation d, second moment $d^{(2)}$ and Laplace Stieltjes Transform $D^*(s) = E[\mathrm{e}^{-sD_n}]$. Let Y_n be the state of the channel at the nth potential loss instant. We assume further that the sequences $\{Y_n\}$ and $\{D_n\}$ are independent. We assume that $\{Y_n\}$ is an ergodic Markov chain with the following transition probabilities:

$$p_{ij} = P\{Y_{n+1} = j \mid Y_n = i\}, \quad 1 \leqslant i, \ j \leqslant N.$$

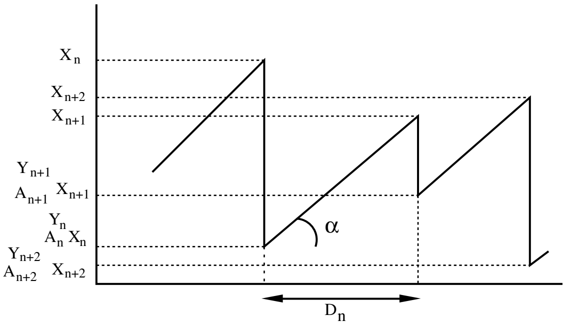

Fig. 2. Window evolution of the TCP model.

Let $P = \{p_{ij}\}_{i,j=1}^{N}$ and let π be the stationary distribution of the Markov chain associated to the path. Next we define N random variables (discrete or continuous), $\{A_n^j; 1 \leqslant j \leqslant N\}$, which describe the behavior of the transmission rate when a potential loss occurs: is it reduced and if so by how much. These variables $\{A_n^j; 1 \leqslant j \leqslant N\}$ correspond to the N possible states of the model for losses. Each random variable A_n^j, $1 \leqslant j \leqslant N$, takes real values in the interval $[0,1]$. The choice of the interval $[0,1]$ stems from the fact that we are *scaling down* the transmission rate at the instant of losses. The set includes 1 since it corresponds to the case when a potential loss is not transformed into a real loss and so the transmission rate is unaltered. A_n^j, $1 \leqslant j \leqslant N$ has a distribution function $F^j(a)$ for all $n \in \mathbb{Z}$. That is, we take the distribution of A_n^j to be time homogeneous. Denote

$$a_i := \int_0^1 a \, \mathrm{d}F^i(a), \quad 1 \leqslant i \leqslant N.$$

We assume that there is at least one i for which $a_i < 1$. The dynamics of the system can be given by the following stochastic recurrent equation:

$$X_{n+1} = \sum_{j=1}^{N} A_n^j X_n 1\{Y_n = j\} + \alpha D_n, \tag{1}$$

where in the case of TCP $\alpha = 1/\mathrm{RTT}^2$ (or $\alpha = 1/2\ \mathrm{RTT}^2$ if the delayed acknowledgment mechanism is enabled). An example of a typical window size evolution described by (1) is shown in Fig. 2.

4. Performance analysis

First we observe that Eq. (1) is a particular case of stochastic linear difference equations of type $X_{n+1} = \beta_n X_n + \gamma_n$, where $\{\beta_n, \gamma_n\}$ is a stationary and ergodic processes (one can consider the Markov chain $\{Y_n\}$ in the stationary regime). It follows from [9,14] that such equations have a stationary solution X_n^* given by

$$X_n^* = \sum_{k=0}^{\infty} \left(\prod_{i=n-k}^{n-1} \beta_i \right) \gamma_{n-k-1}.$$

Moreover, for our problem, the stationary regime exists under the assumption that there is at least one i for which $a_i < 1$. Moreover, for any arbitrary starting point X_0, the sequence $\{X_n\}$ will converge almost surely to this stationary regime, that is

$$\lim_{n\to\infty} |X_n - X_n^*| = 0, \quad \text{P-a.s.}$$

Therefore, we can assume without loss of generality that the process $\{X_n\}$ is in the stationary regime in order to compute the limit distribution. Next we compute the moments of X_n in this regime. Let us denote

$$x_i = E[X_n 1\{Y_n = i\}], \quad 1 \leqslant i \leqslant N.$$

Obviously, the expectation of X_n is given by

$$E[X_n] = \sum_{i=1}^{N} x_i.$$

To compute x_i, $1 \leqslant i \leqslant N$, we use the Laplace Stieltjes Transform approach. Namely, define the following Laplace–Stieltjes Transforms:

$$W(s,i) = E[\mathrm{e}^{-sX_n} 1\{Y_n = i\}], \quad 1 \leqslant i \leqslant N,$$

where we assume that X_n is in the stationary regime.

Theorem 1. *The Laplace–Stieltjes Transforms $W(s,j)$, $1 \leqslant j \leqslant N$, are solutions of the following implicit equations*:

$$W(s,j) = D^*(\alpha s)\left[\sum_{i=1}^{N} p_{ij} \int_0^1 W(as,i)\,\mathrm{d}F^i(a)\right], \quad 1 \leqslant j \leqslant N. \tag{2}$$

Proof. We write for any j, $1 \leqslant j \leqslant N$,

$$\begin{aligned}
E\left[\mathrm{e}^{-sX_{n+1}} 1\{Y_{n+1} = j\}\right] &= \sum_{i=1}^{N} E\left[\mathrm{e}^{-sX_{n+1}} 1\{Y_{n+1} = j\} \mid Y_n = i\right] P(Y_n = i) \\
&= \sum_{i=1}^{N} E[\mathrm{e}^{-sX_{n+1}} \mid Y_n = i] E[1\{Y_{n+1} = j\} \mid Y_n = i] P(Y_n = i) \\
&= \sum_{i=1}^{N} E\left[\mathrm{e}^{-s(A_n^i X_n + \alpha D_n)} \mid Y_n = i\right] p_{ij} P(Y_n = i) \\
&= D^*(\alpha s) \sum_{i=1}^{N} \int_0^1 E\left[\mathrm{e}^{-saX_n} \mid Y_n = i\right] \mathrm{d}F^i(a) p_{ij} P(Y_n = i) \\
&= D^*(\alpha s) \sum_{i=1}^{N} p_{ij} \int_0^1 E\left[\mathrm{e}^{-saX_n} 1\{Y_n = i\}\right] \mathrm{d}F^i(a).
\end{aligned}$$

This results in the implicit equations (2). □

Although the Laplace–Stieltjes Transforms in Theorem 1 are only given as solutions of implicit equations, all moments of $X_n 1\{Y_n = i\}$ for $1 \leqslant i \leqslant N$ (in the stationary regime) can be obtained explicitly. Note that

$$E\left[X_n^k 1\{Y_n = i\}\right] = (-1)^k \left.\frac{\mathrm{d}^k W(s,i)}{\mathrm{d}s^k}\right|_{s=0}.$$

We shall now proceed to the calculation of expressions for the first and second moments of $X_n 1\{Y_n = i\}$ for $1 \leqslant i \leqslant N$ from the implicit expressions of the Laplace Stieltjes transforms. Upon differentiating the implicit expressions (2) and using the following relations:

$$W(0,i) = \pi_i, \quad 1 \leqslant i \leqslant N, \quad D^*(0) = 1, \quad \left.\frac{\mathrm{d}D^*(\alpha s)}{\mathrm{d}s}\right|_{s=0} = -\alpha \mathrm{d},$$

we get N linear equations in N unknowns:

$$x_j = \sum_{i=1}^{N} p_{ij} a_i x_i + \alpha d \pi_j, \quad 1 \leqslant j \leqslant N. \tag{3}$$

Remark 2. Observe that in the stationary case, by multiplying both sides of (1) by $1\{Y_{n+1} = j\}$ and taking expectations we can obtain (3).

We shall now write the above N equations in matrix notation. Let

$$x = [x_1, x_2, \ldots, x_N] \quad \text{and} \quad A = \begin{bmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_N \end{bmatrix}.$$

Then, Eqs. (3) take the form

$$x = xAP + \alpha d \pi. \tag{4}$$

Recall that $0 \leqslant a_i \leqslant 1$ for all i. Furthermore, we assume that there is at least one i for which $a_i < 1$. The latter guarantees that the matrix AP is sub-stochastic (there is an i for which $\sum_{j=1}^{N} p_{ij} a_i < \sum_{j=1}^{N} p_{ij} = 1$). Recall that moduli of all eigenvalues of a sub-stochastic matrix are strictly less than one. Therefore, matrix $I - AP$ has no zero eigenvalue, and consequently, Eq. (4) has a unique solution. Thus, we can state the following result:

Theorem 2. *Let X_n be in the stationary regime. Then $E[X_n]$ is given by*

$$E[X_n] = xe = \alpha d \pi (I - AP)^{-1} e,$$

where e is a column vector of ones.

To compute the second moment of X_n, we first define

$$x_i^{(2)} = E[X_n^2 1\{Y_n = i\}], \quad 1 \leqslant i \leqslant N.$$

Clearly,

$$E[X_n^2] = \sum_{i=1}^{N} x_i^{(2)}.$$

Also let

$$x^{(2)} = \left[x_1^{(2)}, x_2^{(2)}, \ldots, x_N^{(2)}\right] \quad \text{and} \quad A^{(2)} = \begin{bmatrix} a_1^{(2)} & 0 & \cdots & 0 \\ 0 & a_2^{(2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_N^{(2)} \end{bmatrix},$$

where

$$a_i^{(2)} = \int_0^1 a^2 \, \mathrm{d}F^i(a), \quad 1 \leqslant i \leqslant N.$$

Then, in the next theorem we give an explicit expression for $E[X_n^2]$.

Theorem 3. *Let $\{X_n\}$ be in the stationary regime and there is at least one i for which $a_i < 1$. Then, $E[X_n^2]$ is given by*

$$E[X_n^2] = x^{(2)}e = (2\alpha d(xAP) + \alpha^2 d^{(2)}\pi)(I - A^{(2)}P)^{-1}e.$$

Proof. Differentiating twice the implicit expressions (2), we obtain

$$\frac{\mathrm{d}^2 W(s,j)}{\mathrm{d}s^2} = D^*(\alpha s)\left[\sum_{i=1}^N p_{ij} \int_0^1 \frac{\mathrm{d}^2 W(as,i)}{\mathrm{d}s^2} \mathrm{d}F^i(a)\right] + \frac{\mathrm{d}^2 D^*(\alpha s)}{\mathrm{d}s^2}\left[\sum_{i=1}^N p_{ij} \int_0^1 W(as,i) \, \mathrm{d}F^i(a)\right] + 2\frac{\mathrm{d}D^*(\alpha s)}{\mathrm{d}s}\left[\sum_{i=1}^N p_{ij} \int_0^1 \frac{\mathrm{d}W(as,i)}{\mathrm{d}s} \mathrm{d}F^i(a)\right].$$

Now evaluating the above derivatives at $s = 0$, we get

$$x_j^{(2)} = \sum_{i=1}^N p_{ij} a_i^{(2)} x_i^{(2)} + 2\alpha d \sum_{i=1}^N p_{ij} a_i x_i + \alpha^2 d^{(2)} \pi_j.$$

Next, we rewrite the equations in matrix notation

$$x^{(2)} = x^{(2)} A^{(2)} P + 2\alpha d(xAP) + \alpha^2 d^{(2)}\pi.$$

Solving for $x^{(2)}$, we get

$$x^{(2)} = \left(2\alpha d(xAP) + \alpha^2 d^{(2)}\pi\right)(I - A^{(2)}P)^{-1}.$$

The existence of $(I - A^{(2)}P)^{-1}$ is guaranteed, because $A^{(2)}P$ is again sub-stochastic as the sum of the elements of the ith row of $A^{(2)}P$ is $\sum_{j=1}^N p_{ij} a_j^{(2)} < \sum_{j=1}^N p_{ij} = 1$. □

Observe that we computed the expectation of the transmission rate with respect to loss instants. This expectation is also referred to as Palm expectation in the context of marked point processes [7]. Of course, the most interesting is the calculation of the expectation of the transmission rate at an arbitrary time moment. For ergodic processes, the latter expectation coincides with the following time average P-a.s.,

$$\bar{x} = \lim_{T \to \infty} \frac{1}{T} \int_0^T X(t) \, \mathrm{d}t.$$

This is no other than the throughput of the TCP transfer. It is the total volume of transmitted data over the transfer time. We proceed to evaluate this throughput by employing the concept of Palm probability.

Theorem 4. *The throughput, or the time-average transmission rate, is given by*

$$\bar{x} = E[X(t)] = \sum_{i=1}^N a_i x_i + \tfrac{1}{2}\alpha \frac{d^{(2)}}{d} = ax^T + \tfrac{1}{2}\alpha \frac{d^{(2)}}{d}, \tag{5}$$

where $a = [a_1, a_2, \ldots, a_N]$, and x is given in Theorem 2.

Proof. To compute $E[X(t)]$, one can use the following inversion formula (see e.g., [7, Ch. 1, Sec. 4]),

$$E[X(t)] = \frac{1}{d} E^0 \left[\int_0^{T_1} X(t)\,\mathrm{d}t \right], \tag{6}$$

where $E^0[\cdot]$ is an expectation associated with Palm distribution. Thus, we can write

$$E[X(t)] = \frac{1}{d} E^0 \left[\int_0^{T_1} \left(\sum_{i=1}^{N} A_0^i X_0 1\{Y_0 = i\} + \alpha t \right) \mathrm{d}t \right].$$

Because of the independence of X_n and $\{D_k, k \geqslant n\}$ and also because of the independence of $\{D_n\}$ and $\{Y_n\}$ we can write

$$E[X(t)] = \frac{1}{d} \left[\sum_{i=1}^{N} \left(E^0[A_0^i] E^0[X_0 1\{Y_0 = i\}] \right) E^0[D_0] \right] + \frac{\alpha}{2d} E^0[D_0^2] = \sum_{i=1}^{N} a_i x_i + \tfrac{1}{2}\alpha \frac{d^{(2)}}{d} = a x^T + \tfrac{1}{2}\alpha \frac{d^{(2)}}{d}. \qquad \square$$

In the next theorem we evaluate the second moment of the transmission rate at an arbitrary time instant. This second moment describes how much the transmission rate varies. It could be used by TCP-friendly real-time applications (e.g., audio or video transmissions) [13]. These applications may choose to change their parameters for increasing the window and decreasing it so as to minimize the oscillation of the transmission rate while keeping the same throughput as in the case of TCP. A smoother transmission is a requirement for a better quality of service in case of such delay-sensitive applications. The latter requirement on the throughput stems from fairness arguments.

Theorem 5. *Let $d^{(3)}$ be the third moment of the time between potential losses. The second moment of the input rate over a long time interval is equal to*

$$\bar{x}^{(2)} = \lim_{t\to\infty} \frac{1}{t} \int_0^t X^2(t)\,\mathrm{d}t = \tfrac{1}{3}\alpha^2 \frac{d^{(3)}}{d} + \frac{1}{d}\alpha d^{(2)} a x^T + a^{(2)} x^{(2)^T},$$

where $a^{(2)} = [a_1^{(2)}, a_2^{(2)}, \ldots, a_N^{(2)}]$ and $x^{(2)}$ are given in Theorem 3.

Proof. Again by the inversion formula from Palm probability,

$$\begin{aligned} E[X^2(t)] &= \frac{1}{d} E^0 \left[\int_0^{T_1} X^2(t)\,\mathrm{d}t \right] = \frac{1}{d} E^0 \left[\int_0^{T_1} \left(\sum_{i=1}^{N} A_0^i X_0 1\{Y_0 = i\} + \alpha t \right)^2 \mathrm{d}t \right] \\ &= \frac{1}{d} E^0 \left[\frac{\alpha^2 D_0^3}{3} + \alpha D_0^2 \sum_{i=1}^{N} A_0^i X_0 1\{Y_0 = i\} + \sum_{i=1}^{N} (A_0^i)^2 X_0^2 1\{Y_0 = i\} D_0 \right] \\ &= \tfrac{1}{3}\alpha^2 \frac{d^{(3)}}{d} + \frac{1}{d}\alpha d^{(2)} \sum_{i=1}^{N} a_i x_i + \sum_{i=1}^{N} a_i^{(2)} x_i^2 = \tfrac{1}{3}\alpha^2 \frac{d^{(3)}}{d} + \frac{1}{d}\alpha d^{(2)} a x^T + a^{(2)} x^{(2)^T}. \qquad \square \end{aligned}$$

Having obtained the expressions for the general case of N states, we shall now focus on some particular cases in the following sections. We will show how the parameters of our model can be inferred from a real trace of a TCP connection. Different possible applications of the model to the same trace are presented and the results are then compared to show which method is the most efficient.

5. Specifications of the general model

In this section, we present different ways for the application of our general model to predict the performance of a TCP-like flow control mechanism. We chose to work with real loss processes. From the trace of a TCP connection, we determine the moments of window reduction. We reconstruct then the evolution of TCP congestion window over time under the assumption that the window increases linearly between two consecutive losses. We call this reconstructed window evolution the Exact Fluid Model and we use it below as a reference. We then try to derive simple closed form expressions for the throughput of the exact fluid model, and therefore for the throughput of TCP, using simple versions of our general model.

We ran long-lived TCP connections between INRIA and different machines over the Internet. Our total results, which are summarized in [2], have shown that the process of loss events is indeed bursty in metropolitan networks. Fig. 1 is a proof of this burstiness. In wide area networks however, the process of loss events is close to Poisson and a simple model can then be used to predict the performance of TCP. Hence, to validate our present model that accounts for correlation and burstiness of loss events, we only consider the results obtained in a metropolitan network. These results are collected from a TCP connection running between machines in Sophia Antipolis Park, 1 km apart from each other.

The reason for which we chose to use the Exact Fluid Model as a reference and not TCP itself, is that on the connection we considered, the evolution of the congestion window has been found to be strongly sub-linear. Sub-linearity in TCP dynamics, which is very hard to model (see [5] for a modeling of this sub-linearity in the case of a Poisson process of loss events), appears when the round-trip time of the TCP connection increases with the congestion window. It seems to be impossible to get explicit expressions for TCP throughput in presence of such sub-linearity. Given that our main objective is to well model the process of loss events, we decided to eliminate such sub-linearity in TCP dynamics by using the Exact Fluid Model. As it is explained in [5,8], keeping this sub-linearity can make a wrong model for loss events gives better results than a right model. The introduction of sub-linearity into our present model will be the topic of our future research.

Our experimentation consists then of a long-lived New-Reno TCP connection running between `clope.inria.fr` at INRIA and `nessie.essi.fr` at ESSI, both located in the technology park Sophia Antipolis in the south of France. The two machines are connected to the same metropolitan network. The TCP connection is run eleven times for approximately 20 min each at the most busy periods (between 10 a.m. and 2 p.m.). The choice of 20 min as a duration of every run of the TCP connection is only made by our will to ensure around 500 loss events per run, which allows a low estimation error for moments and transition probabilities. The trace of the connection is captured at the source using the `tcpdump` tool [17] and a program is developed to analyze the traces in order to find the moments at which the congestion window is divided by two. We noticed that most of the time, the loss of packets is detected with the Fast Retransmit algorithm (3 Duplicate ACKs) [26]. We also noticed that the maximum window advertised by the receiver is rarely reached due to working at busy periods.

5.1. The basic model

We consider here the very simple case where the path has a single state and where the transmission rate is divided once by two at every potential loss event. We assume that the times between losses are iid. This gives the following expression for the throughput:

$$E[X(t)] = \alpha \mathrm{d} + \tfrac{1}{2}\alpha \frac{d^{(2)}}{d}. \tag{7}$$

Obviously, if times between losses are really iid, this model must give a very close throughput to that given by the exact fluid model. And indeed, in our experiments, we did not find a significant correlation among inter-loss times. Fig. 3 confirms this conclusion. The throughput given by formula (7) follows closely

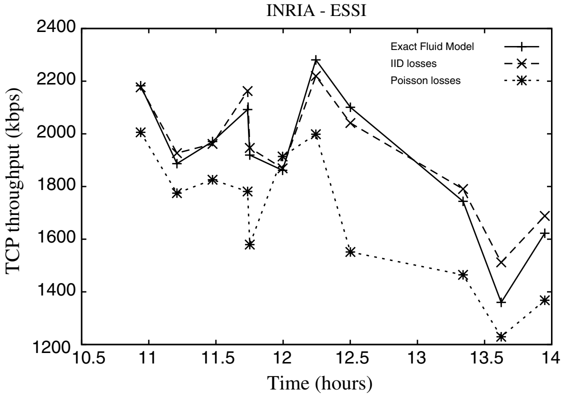

Fig. 3. Comparison of Poisson, iid and exact fluid models.

the one given by the exact fluid model. However, to use formula (7) for the throughput calculation, one must know the second moment of inter-loss times. In Fig. 3, we compute this second moment from our measurements. Usually, this quantity is difficult to find since it requires the knowledge of all inter-loss times for the modeled connection. Note that, by contrast, d can be easily calculated by dividing the total time of the connection by the number of loss events. The number of losses in turn can be calculated using the packet loss probability. One way to eliminate $d^{(2)}$ is to express it as a function of d. For example, one can assume that inter-loss times form a Poisson process and hence take $d^{(2)} = 2d^2$. The problem with this solution is that it hides the impact of burstiness and expresses the throughput only as a function of the average loss rate. Indeed, in Fig. 3, the throughput calculated according to the Poisson assumption does not match well the throughput of the exact fluid model. The reason for this mismatch is clearly explained by Fig. 4 where we plot the histogram of inter-loss times. This figure shows the deviation of the inter-loss time distribution from the exponential shape. This deviation is caused by the appearance of bursts of losses, which causes the pulse of probability around the origin. Indeed, we noticed from the real traces of a TCP connection that the congestion window is divided multiple times by two when a congestion occurs and this due to the loss of packets in multiple consecutive Round Trips (see also Fig. 1). However, the important notice we made from Fig. 4 is that the time between bursts can still be well approximated by the exponential distribution. Fig. 5 shows the distribution of times between losses after the elimination

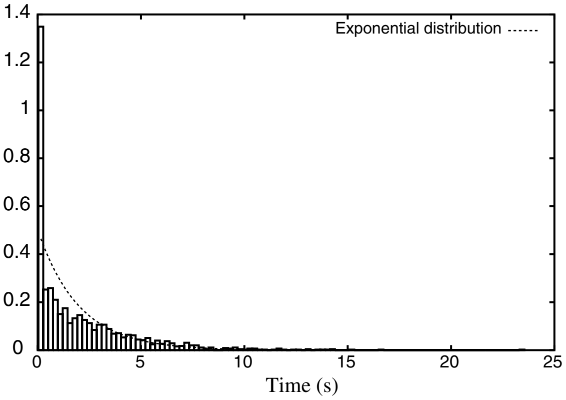

Fig. 4. Histogram of inter-loss times.

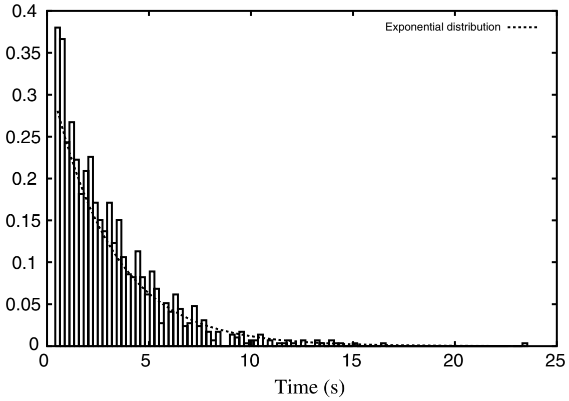

Fig. 5. Histogram of times between bursts.

of the pulse around the origin. In the next two sections, we will present two methods to account for this bursty behavior of losses.

5.2. The aggregate loss method

As was noticed in Fig. 4, the inter-loss time distribution is a mixture of two distributions, one around the origin represents the time between losses within bursts and another away from the origin represents the time between bursts. This prompts us to aggregate the losses inside a burst into a single loss and to divide the transmission rate upon an aggregate loss occurrence (or a burst occurrence) by two powers the number of aggregated losses inside the burst. The aggregate loss process can be considered now as a Poisson process. Upon the arrival of an aggregate loss, the transmission rate is divided by a random factor that can be greater than two. The question that one may ask here is how to characterize a burst, in other words how to decide that two consecutive losses are within the same burst or within two different bursts. In this section, we use the following empirical method: we look at the distribution of inter-loss times and try to find a point, which clearly separates the two distributions. We zoom in Fig. 6, the distribution of inter-loss times (Fig. 4) around the origin. It is clear that two bursts are separated by approximately $\delta = 0.4$ s. We use this δ for the identi-

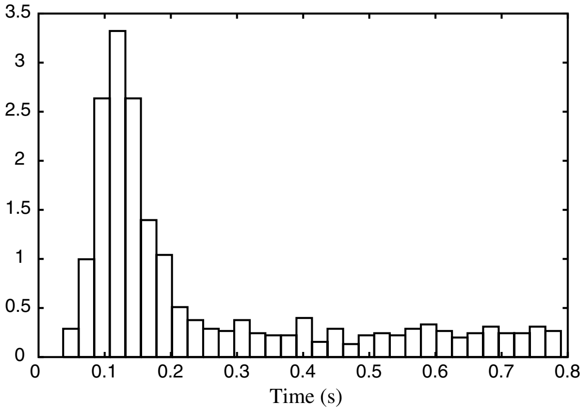

Fig. 6. Histogram of inter-loss times within bursts.

fication of bursts. In the following, we present two different ways to describe the behavior of the random reduction factor. The first way is to assume that it is iid. The second way is to model it with a Markov chain.

First, let us consider the case of iid. reduction factor. The evolution of the transmission rate in this case is given by

$$X_{n+1} = A_n X_n + \alpha D_n,$$

where the reduction factor A_n has a distribution function $F(a)$. D_n is the time between bursts which can be approximated by a Poisson process. Of course, this can be viewed as a particular case of our general model where the path of the connection has only one state. The general results of Section 2 can be specified for the present case as follows:

$$E[X_n] = \frac{\alpha d}{1-\bar{a}},$$

$$\bar{x} = E[X(t)] = \frac{\alpha d\bar{a}}{1-\bar{a}} + \tfrac{1}{2}\alpha\frac{d^{(2)}}{d}, \tag{8}$$

where $\bar{a} = \int_0^1 a\,\mathrm{d}F(a)$. Here, the reduction factor A_n is a discrete random variable which takes the values multiple of 1/2. Thus, we calculated $\bar{a}$ as

$$\bar{a} = \sum_{i=1}^{m} \frac{1}{2^i} p_i, i,$$

where the probabilities p_i are estimated from the TCP connection trace. Let n be the total number of aggregate losses in the trace. We can write

$$p_i = \sum_{k=1}^{n} 1\{a_k = 1/2^i\}/n.$$

Note here that the main gain from aggregation, is that the second moment of D_n can now be taken as $2d^2$ (exponential random variable). Furthermore, from Fig. 5, one can see that the distribution of D_n is a shifted exponential distribution given that the time between two aggregate losses is always larger than δ. Thus, a more correct estimation for the second moment is given by

$$d^{(2)} = \delta^2 - 2\delta d + 2d^2.$$

Next we consider the case where the reduction factor is modeled using a Markov chain. We associate a multi-state Markov chain to the path. The transitions of the chain occur upon aggregate loss arrivals. The state of the chain when an aggregate loss arrives is equal to the number of losses within the burst. The Markov chain determines then how many times the transmission rate is divided by two. Fig. 7 explains how the transmission rate and the Markov chain change together. Observe that, dividing the y-axis values in Fig. 7 by 1000 we get the values for the y-scale corresponding to the Markov chain. A interval of 0.4 s is used to identify the losses belonging to the same burst. The evolution of the transmission rate in this case can be described as follows:

$$X_{n+1} = \sum_{j=1}^{N} a_j 1\{Y_n = j\} X_n + \alpha D_n, \tag{9}$$

where a_j is constant equal to $1/2^j$ and where Y_n is the state of the Markov chain. D_n again represents the time between bursts which can be approximated by a Poisson process. As a corollary of Theorem 3, the throughput can be written as

$$\bar{x} = E[X(t)] = \sum_{j=1}^{N} a_j x_j + \frac{\alpha d^{(2)}}{2d}. \tag{10}$$

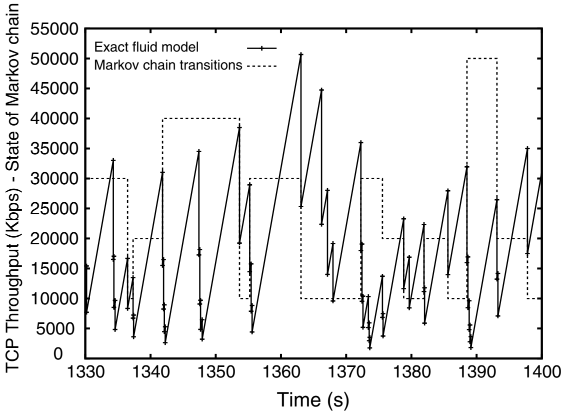

Fig. 7. Transitions of the multi-state Markov chain.

The estimations of transition probabilities p_{ij}, $i,j = 1,\ldots,N$, of the Markov chain $\{Y_k\}$ are identified from the trace of the TCP connection as follows:

$$p_{ij} = \sum_{k=1}^{n-1} 1\{Y_{k+1} = j, Y_k = i\} \bigg/ \sum_{k=1}^{n-1} 1\{Y_k = i\},$$

where the Markov chain state Y_k corresponds to the number of transmission rate reduction at the event of the kth aggregate loss and n is the total number of aggregate loss events. If the number of transmission rate reductions at the aggregate loss moment is greater than N, we assume that the Markov chain is in the state N. Since N is chosen so that it is unlikely to have the rate reduced more than N times during a burst, this assumption should not cause any problem. In the following we take $N = 4$.

Using the maximum distance of 0.4 s between losses within a burst (Fig. 6), we aggregate in bursts the moments at which the transmission is divided by two. As before, we assume that the resulting aggregate loss process is Poisson. We approximate the throughput of the exact fluid model using Eqs. (8) and (10). Fig. 8 shows the results. The iid. batch model denotes the first case where the number of losses in a burst is

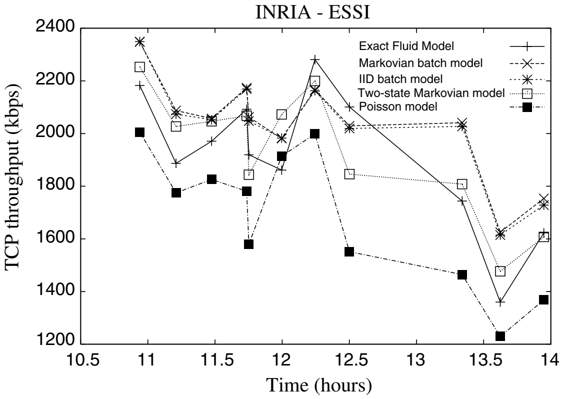

Fig. 8. Comparison among the different methods.

described by an iid. random variable. The Markovian batch model denotes the second case where this number is described by a Markov chain. We notice that the two methods give approximately the same result, which means that the number of losses within a burst is really iid. distributed. The result is closer to that of the exact fluid model than the throughput calculated for the Poisson model. However, it is not as good as we expected. The main reason is that we are ignoring the length of a burst, which is here comparable to the time between bursts. Possibly, for other connections where losses are more clustered together, this batch method will have a better performance. One may expect that the Markov version of the batch model will perform better than the iid. version on connections where strong correlation exists among burst sizes. In the next subsection, we will present a model that accounts for the time the connection spends during a burst.

5.3. The two-state model

Consider a particular case of our general model where the path switches between two different states. Namely, let $N = 2$ and let the state 1 corresponds to the *Good* state of the path and the state 2 to the *Bad* state. We also denote the transition probabilities of the Markov chain as follows: $p_{11} = g$, $p_{12} = \bar{g} = 1 - g$, $p_{21} = \bar{b} = 1 - b$ and $p_{22} = b$. The stationary distribution of this chain are equal to,

$$\pi_1 = \frac{\bar{b}}{\bar{b} + \bar{g}}, \quad \pi_2 = \frac{\bar{g}}{\bar{b} + \bar{g}}.$$

The following results can be easily obtained as straightforward corollaries of the theorems for the general N state model.

Corollary 1. *The Laplace–Stieltjes Transforms $W(s, i)$, $i = 1, 2$, are the solutions of the following implicit equations*:

$$W(s,1) = D^*(\alpha s)\left[g\int_0^1 W(as,1)\,\mathrm{d}F^1(a)\right] + D^*(\alpha s)\left[\bar{b}\int_0^1 W(as,2)\,\mathrm{d}F^2(a)\right],$$

$$W(s,2) = D^*(\alpha s)\left[\bar{g}\int_0^1 W(as,1)\,\mathrm{d}F^1(a)\right] + D^*(\alpha s)\left[b\int_0^1 W(as,2)\,\mathrm{d}F^2(a)\right].$$

We shall now proceed to obtain explicit expressions for the first and second moments of the transmission rate at potential loss instants.

Corollary 2. *The first moment of the transmission rate at a potential loss event is given by*

$$E[X_n] = x_1 + x_2,$$

where

$$x_1 = \alpha d \frac{a_2(\pi_2 - b) + \pi_1}{1 - a_2 b - a_1 g + a_1 a_2 (g + b - 1)}, \tag{11}$$

$$x_2 = \alpha d \frac{a_1(\pi_1 - g) + \pi_2}{1 - a_2 b - a_1 g + a_1 a_2 (g + b - 1)}. \tag{12}$$

Corollary 3. *The second moment of the transmission rate at a potential loss event is given by*

$$E[X_n] = x_1^{(2)} + x_1^{(2)},$$

where

$$x_1^{(2)} = \frac{2\alpha d a_1 a_2^{(2)} x_1 (1-g-b) + 2\alpha d(a_2 x_2 \bar{g} + a_1 x_1 g) + \alpha^2 d^{(2)} \left(a_2^{(2)} \pi_2 + \pi_1 - b a_2^{(2)}\right)}{\left(1 - g a_1^{(2)} - b a_2^{(2)} - a_1^{(2)} a_2^{(2)} (1-g-b)\right)}, \tag{13}$$

$$x_2^{(2)} = \frac{2\alpha d a_1^{(2)} a_2 x_2 (1-g-b) + 2\alpha d(a_1 x_1 \bar{g} + a_2 x_2 b) + \alpha^2 d^{(2)} \left(a_1^{(2)} \pi_1 + \pi_2 - g a_1^{(2)}\right)}{\left(1 - g a_1^{(2)} - b a_2^{(2)} - a_1^{(2)} a_2^{(2)} (1-g-b)\right)}. \tag{14}$$

Corollary 4. *The throughput, or the time-average of the transmission rate, is given by*

$$E[X(t)] = a_1 x_1 + a_2 x_2 + \tfrac{1}{2}\alpha \frac{d^{(2)}}{d},$$

where x_1 and x_2 are given in Eqs. (11) *and* (12).

Corollary 5. *The second moment of the transmission rate at an arbitrary time instant is given by*

$$E[X^2(t)] = a_1^{(2)} x_1^{(2)} + a_2^{(2)} x_2^{(2)} + \frac{\alpha d^{(2)} (a_1 x_1 + a_2 x_2)}{d} + \tfrac{1}{3}\alpha^2 \frac{d^{(3)}}{d},$$

where x_1 and x_2 are given in Eqs. (11) *and* (12) *and $x_1^{(2)}$ and $x_2^{(2)}$ in Eqs.* (13) *and* (14), *respectively.*

Next we specialize the model further by taking A_n^j, for $j \in \{1,2\}$ and $\forall n \geqslant 0$, to be discrete random variables with values in $\{0.5, 1\}$. Note that $A_n^j = 0.5$ represents the case when a potential loss is transformed into a real loss, namely when it causes a reduction in the transmission rate, whereas $A_n^j = 1$ represents the case when the transmission rate is not reduced at the potential loss event. We get here the same model as that described in [1]. Note that in [1] we validate via simulation a particular case of this two-state model that corresponds to $p_G = 0$, $p_B = 1$. In the present work, we show how to set the different parameters of the two-state model in its general case. $\{D_n\}$ is the sequence of the times between potential losses. We also denote $p_G := P\{A_n^1 = 0.5\} = 1 - P\{A_n^1 = 1\}$, as the probability of the event when a potential loss is transformed into a real loss in the Good state. Analogously, we define the probability of a potential loss becoming a real loss in the Bad state as $p_B := P\{A_n^2 = 0.5\} = 1 - P\{A_n^2 = 1\}$. We assume that $p_G \leqslant p_B$. Clearly,

$$a_1 = 1 - \tfrac{1}{2} p_G \quad \text{and} \quad a_2 = 1 - \tfrac{1}{2} p_B.$$

Next we demonstrate how the above introduced parameters as well as d and the transition matrix P can be determined from the data in real TCP traces. First, we obtain an estimation of the transition matrix for the Markov chain $\{Y_n\}$. Recall that this is the Markov chain obtained when looking at the state of the channel at potential loss events. Let $\{S_n\}$ be a sequence of inter-loss times measured from a TCP trace. We need to determine when the path is in the "Good" state and when it is in the "Bad" state. We use the following simple method. Choose some time interval τ. We will explain later how to make this choice. If the inter-loss time S_n is less than τ then the path is in the Bad state, otherwise the path is considered to be in the Good state. If two or more inter-loss times correspond to the same state, we will merge these intervals together and call the new interval L_k^G or L_k^B depending on the state. Note that these new intervals represent the time during which the path of the connection is either in the Good or in the Bad state. Denote n_G (resp. n_B) the number of the time intervals S_k^G (resp. S_k^B) during the time interval that we use for measurement. Then, the evolution of the path of the TCP connection can be described by a two-state continuous time Markov process with the following infinitesimal generator matrix:

$$Q = \begin{bmatrix} -\sigma_{\mathrm{G}} & \sigma_{\mathrm{G}} \\ \sigma_{\mathrm{B}} & -\sigma_{\mathrm{B}} \end{bmatrix}, \tag{15}$$

where the rates σ_{G} and σ_{B} are calculated as follows:

$$\sigma_{\mathrm{G}} = \frac{1}{E[S_k^{\mathrm{G}}]} \simeq \frac{n_{\mathrm{G}}}{\sum_{k=1}^{n_{\mathrm{G}}} S_k^{\mathrm{G}}}, \quad \sigma_{\mathrm{B}} = \frac{1}{E[S_k^{\mathrm{B}}]} \simeq \frac{n_{\mathrm{B}}}{\sum_{k=1}^{n_{\mathrm{B}}} S_k^{\mathrm{B}}}.$$

Note that on some paths, say a wireless link, this Markov chain is apriori known and can be directly used without the need to look at the trace of the TCP connection. In case it is not known, we need to define it using the parameter τ as described above. We present now two approaches for the determination of τ. The first one is more empirical. We look at the histogram of the inter-loss times (Fig. 4) and we choose τ as the time separating the two distributions it encloses (0.4 s in the figure). The second method is less empirical and was used in the context of Markov-modulated Poisson processes [19]. In this second approach, we define parameter τ as the expectation of the inter-loss times, that is

$$\tau = E[S_k] \simeq \frac{1}{n} \sum_{k=1}^{n} S_k,$$

where n is the total number of inter-loss intervals we get from the trace. Given the continuous time Markov chain associated to the channel, we can now extract the parameters of the discrete time Markov chain embedded at the potential loss events. We use for this purpose the uniformization technique [29]. Let us choose the potential loss process $\{D_n\}$ as a Poisson process with intensity $1/d$ higher than both σ_{G} and σ_{B}. For example, a reasonable choice of d is the estimation of the average Round Trip Time of the connection. According to the uniformization technique [29], the state of the path described by the Markov process (15) and sampled at the moments of potential losses can be equivalently given by a discrete time Markov chain with the following transition matrix:

$$P = \begin{bmatrix} 1 - d\sigma_{\mathrm{G}} & d\sigma_{\mathrm{G}} \\ d\sigma_{\mathrm{B}} & 1 - d\sigma_{\mathrm{B}} \end{bmatrix}.$$

Having chosen d and calculated σ_{G} and σ_{B} from the trace, we can easily deduce the parameters b and g of the loss model. Namely, $\bar{g} = d\sigma_{\mathrm{G}}$ and $\bar{b} = d\sigma_{\mathrm{B}}$. Now we determine p_{G} and p_{B}. Let ω_k^{G} (ω_k^{B}) be the number of real losses in the time interval S_k^{G} (resp. in S_k^{B}). Then the probabilities p_{G} and p_{B} are given by

$$p_{\mathrm{G}} = \frac{\sum_{k=1}^{n_{\mathrm{G}}} \omega_k^{\mathrm{G}}}{\sum_{k=1}^{n_{\mathrm{G}}} S_k^{\mathrm{G}}/d} = \frac{d\sum_{k=1}^{n_{\mathrm{G}}} \omega_k^{\mathrm{G}}}{\sum_{k=1}^{n_{\mathrm{G}}} S_k^{\mathrm{G}}} = d\lambda_{\mathrm{G}}, \quad p_{\mathrm{B}} = \frac{\sum_{k=1}^{n_{\mathrm{B}}} \omega_k^{\mathrm{B}}}{\sum_{k=1}^{n_{\mathrm{B}}} S_k^{\mathrm{B}}/d} = \frac{d\sum_{k=1}^{n_{\mathrm{B}}} \omega_k^{\mathrm{B}}}{\sum_{k=1}^{n_{\mathrm{B}}} S_k^{\mathrm{B}}} = d\lambda_{\mathrm{B}}.$$

$1/\lambda_{\mathrm{G}}$ and $1/\lambda_{\mathrm{B}}$ represent the average time between window reductions in the Good and in the Bad state respectively. For the same eleven traces obtained in our experiments, we calculated the parameters of the model. We use $\tau = \delta = 0.4$ s to separate the Bad state from the Good state. In Fig. 8, we compare the result with that of the exact fluid model. A close match is noticed. In addition to the good results and the closed form expression it provides, this model has the advantage of having simple parameters. All what we need to approximate the throughput of TCP are the parameters of the two-state Markov chain associated to the path and the intensity of losses in both states. Concerning the parameter d, it is enough to choose in a way that the intensity of potential losses $1/d$ is higher than the intensity of losses in the Bad state λ_{B}.

The maximum receiver window in the 11 runs of our TCP connection is equal to 64 Kbytes (which gives approximately 44 packets of 1460 bytes). Our connections show an average round-trip time equal to 100 ms, which is relatively large due to queuing delays in intermediate routers.

6. Concluding remarks

We considered in this paper a multi-state Markov model to describe the loss process experienced by a flow control mechanism that has a linear window increase between losses, and a multiplicative window decrease upon a loss event. The modeling of some channels using a Markov chain with more than two states have long been advocated, see e.g. [25,28].

Using an approach based on Laplace–Stieltjes Transform, we derived explicit expressions for the two first moments of the transmission rate of the mechanism just prior to losses, as well as the two first moments of the transmission rate at arbitrary time. The first moment of the transmission rate of the flow-control mechanism at an arbitrary time is often the measure of its throughput. We note that the expression for the second moment of the transmission rate at arbitrary time (call it the second moment of the throughput) could be useful in designing TCP friendly protocols for real time applications [13]. In these applications, other parameters of the linear increase and multiplicative decrease are chosen so as to maintain the same throughput (as a function of the loss process and of the round-trip time) as the original TCP protocol. (The latter requirement on the throughput stems from fairness arguments.) Such applications (e.g., interactive voice or video transmissions) typically require a smaller variance of the transmission rate than the one of the original TCP in order to ensure a reasonable quality of service.

In [2] we have succeeded in analyzing non Markovian models for loss events [2], and obtained similar performance measures using a completely different approach (that relies on some covariance functions of the inter-loss times). The approach proposed here, in contrast, leads to formulae that involve only a finite and small number of easily computable parameters. In addition, we proposed here methods for the identification of such parameters.

Acknowledgments

The authors thank anonymous referees for helpful comments on the earlier version of this paper which are, in particular, helpful in better focusing the contributions of this work.

References

[1] E. Altman, K.E. Avrachenkov, C. Barakat, TCP in presence of bursty losses, Performance Evaluation 42 (2–3) (2000) 129–147.

[2] E. Altman, K.E. Avrachenkov, C. Barakat, A stochastic model of TCP-IP with stationary ergodic random losses, in: ACM SIGCOMM, August 28–September 1, Stockholm, Sweden, 2000.

[3] E. Altman, K.E. Avrachenkov, C. Barakat, TCP network calculus: The case of large delay-bandwidth product, in: Proceedings of IEEE INFOCOM, NY, USA, June 2002.

[4] E. Altman, K. Avratchenkov, C. Barakat, R. Núñez-Queija, State-dependent M/G/1 type queueing analysis for congestion control in data networks, in: IEEE INFOCOM, Anchorage, Alaska, April 2001.

[5] E. Altman, K. Avrachenkov, C. Barakat, R. Núñez-Queija, TCP modeling in the presence of nonlinear window growth, in: Proceedings of ITC-17, Salvador da Bahia, Brazil, December 2001.

[6] E. Altman, C. Barakat, V.M. Ramos R., Analysis of AIMD protocols over paths with variable delay, in: IEEE INFOCOM, Hong Kong, March 2004.

[7] F. Baccelli, P. Bremaud, Elements of Queueing Theory: Palm–Martingale Calculus and Stochastic Recurrences, Springer, Berlin, 1994.

[8] C. Barakat, TCP modeling and validation, IEEE Network 15 (3) (2001) 38–47.

[9] A. Brandt, The stochastic equation $Y_{n+1} = A_n Y_n + B_n$ with stationary coefficients, Adv. Appl. Prob. 18 (1986) 211–220.

[10] T. Bu, D. Towsley, Fixed point approximation for TCP behavior in an AQM Network, in: ACM Sigmetrics, Cambridge, MA, USA, June 18–20, 2001.

[11] V. Dumas, F. Guillemin, P. Robert, A Markovian analysis of AIMD algorithms, Adv. Appl. Prob. 34 (1) (2002) 85–111.

[12] K. Fall, S. Floyd, Simulation-based comparisons of Tahoe, Reno, and SACK TCP, in: ACM Computer Communication Review, July 1996.

[13] S. Floyd, M. Handley, J. Padhye, Equation-based congestion control for unicast applications, in: ACM SIGCOMM, Stockholm, Sweden, August 28–September 1, 2000.

[14] P. Glasserman, D.D. Yao, Stochastic vector difference equations with stationary coefficients, J. Appl. Prob. 32 (1995) 851–866.

[15] F. Kelly, Mathematical modeling of the Internet, in: B. Engquist, W. Schmid (Eds.), Mathematics Unlimited—2001 and Beyond, Springer, Berlin, 2001, pp. 685–702.

[16] S. Kunniyur, R. Srikant, End-to-end congestion control: utility functions, random losses and ECN marks, IEEE/ACM Trans. Network. 11 (5) (2003) 689–702.

[17] LBNL's tcpdump tool, available at <http://www-nrg.ee.lbl.gov/>.

[18] S.H. Low, D.E. Lapsley, Optimization flow control, I: Basic algorithm and convergence, IEEE/ACM Trans. Network. 7 (6) (1999) 861–875.

[19] K.S. Meier-Hellstern, A fitting algorithm for Markov-modulated Poisson processes having two arrival rates, Euro. J. Oper. Res. 29 (1987) 370–377.

[20] V. Misra, W. Gong, D. Towsley, Stochastic differential equation modeling and analysis of TCP windowsize behavior, in: Performance '99, Istanbul, Turkey, October 1999.

[21] V. Misra, W. Gong, D. Towsley, Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED, in: Proceedings of ACM SIGCOMM '00, Stockholm, Sweden, 2000.

[22] T.J. Ott, J.H.B. Kemprerman, M. Mathis, The stationary behavior of ideal TCP congestion avoidance, Unpublished manuscript, August 1996.

[23] J. Padhye, V. Firoiu, D. Towsley, J. Kurose, Modeling TCP throughput: a simple model and its empirical validation, in: Proceedings of ACM SIGCOMM, 1998.

[24] J. Padhye, V. Firoiu, D. Towsley, Modeling TCP Reno performance: a simple model and its empirical validation, IEEE/ACM Trans. Network. 8 (2) (2000) 133–145.

[25] M. Rahman, M. Bulmer, M. Wilkinson, Error models for land mobile satellite channels, Aust. Telecommun. Res. 25 (2) (1991) 61–68.

[26] W. Stevens, TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms, RFC 2001, Jan 1997.

[27] M. Vojnovic, J.Y. Le Boudec, C. Boutremans, Global fairness of additive-increase and multiplicative-decrease with heterogeneous round-trip times, in: IEEE INFOCOM, Tel-Aviv, Israel, March 2000.

[28] B. Vucetic, J. Du, Channel modeling and simulation in satellite mobile communication systems, IEEE J. Select. Areas Commun. 10 (8) (1992) 1209–1218.

[29] J. Walrand, An Introduction to Queueing Networks, Prentice Hall, Englewood Cliffs, NJ, 1988.

Eitan Altman received the B.Sc. degree in electrical engineering (1984), the B.A. degree in physics (1984) and the Ph.D. degree in electrical engineering (1990), all from the Technion-Israel Institute, Haifa. In 1990, he further received his B.Mus. degree in music composition in Tel-Aviv University. Since 1990, he has been with INRIA (National Research Institute in Informatics and Control) in Sophia-Antipolis, France. His current research interests include performance evaluation and control of telecommunication networks and in particular congestion control, wireless communications and networking games. He is in the editorial board of several scientific journals: Stochastic Models, JEDC, COMNET, SIAM SICON and WINET. He has been the (co)chairman of the program committee of several international conferences and workshops (on game theory, networking games and mobile networks). More information can be found at http://www.inria.fr/mistral/personnel/Eitan.Altman/me.html.

Konstantin Avrachenkov received the Master degree in control theory (1996) from St Petersburg State Technical University and Ph.D. degree (2000) in Mathematics from University of South Australia. Currently he is with INRIA Sophia Antipolis, France. His main research interests include Markov chains and Markov decision processes, Mathematical Programming, Singular Perturbations. One of his most recent research directions is the performance evaluation of data networks.

Chadi Barakat is a permanent research scientist in the Planete research group at INRIA—Sophia Antipolis since March 2002. He got his Electrical and Electronics engineering degree from the Lebanese University of Beirut in 1997, and his master and Ph.D. degrees in Networking from the University of Nice—Sophia Antipolis in 1998 and 2001. His Ph.D. has been done in the Mistral group at INRIA—Sophia Antipolis. From April 2001 to March 2002, he was with the LCA department at EPFL-Lausanne for a post-doctoral position, and from March to August 2004, he was a visiting faculty member at Intel Research Cambridge. Chadi Barakat was the general chair of PAM 2004 and serves in the program committees of many international conferences as Infocom, PAM, WONS, ASWN and Globecom. His main research interests are congestion and error control in computer networks, the TCP protocol, voice over IP, wireless LANs, Internet measurement and traffic analysis, and performance evaluation of communication protocols.

Parijat Dube received his M.S. in Electrical Communication Engg. from Indian Institute of Science, Bangalore in 2001 and his Ph.D. in Computer Science from University of Nice-Sophia Antipolis in 2002 where he was affiliated to INRIA, Sophia Antipolis, France. He joined IBM T.J. Watson Research Center, Hawthorne, New York in 2002. His research interests include queueing theory, performance evaluation and control of communications networks, sensor networks, revenue management and pricing.