Available online at www.sciencedirect.com

SCIENCE DIRECT®

Performance Evaluation 58 (2004) 89–108

PERFORMANCE EVALUATION
An International Journal

www.elsevier.com/locate/peva

A Markov model of TCP throughput, goodput and slow start

Sophie Fortin-Parisi, Bruno Sericola*

IRISA-INRIA, Campus de Beaulieu, 35042 Rennes Cedex, France

Abstract

This paper presents a discrete-time Markov chain model for TCP, the Transmission Control Protocol for reliable transport on the Internet. The purpose is the evaluation of stationary TCP flows behavior using performance measures such as the mean throughput. The model is based on previous works which are generalized by taking into account the slow start phases that appear after each time-out recovery, whose importance is discussed.

Keywords: TCP; Congestion control; Throughput; Markov chain; Slow start

1. Introduction

The great expansion of the Internet has triggered a lot of work on its efficiency and on possible improvements. The apparently simple mechanism of the *Transport Control Protocol* (TCP) used by HTTP transfer, file transfer, email and remote access has thus been modeled with various stochastic tools.

Assuming a periodic window evolution marked by random loss events of probability p, separating successive congestion avoidance phases, the authors of have shown that the mean throughput ρ was $O(1/\sqrt{p})$.

Many studies are based on a fluid approach and are usually and mainly focussing on getting an analytical expression for the mean throughput of a single steady-state TCP connection. It is the case of [6,12,13,15], but also [1,2,4,5] which focus on the window size W_n just before the n-th loss. The case of multiple TCP connections is the subject of [3,8,11] for instance. Among all other tools explored, the max-plus algebra provides in [7] expressions for the mean throughput in the case of several routers in series.

Our paper is based on previous works presented in [9,16,17] which consider a discrete-time model and a discrete evolution of the window size. We propose here a discrete-time Markov chain model which aims

* Corresponding author.
E-mail addresses: sfortin@irisa.fr (S. Fortin-Parisi), sericola@irisa.fr (B. Sericola).

0166-5316/$ – see front matter © 2004 Published by Elsevier B.V.
doi:10.1016/j.peva.2004.07.016

at giving analytical expressions for measures such as the mean throughput of one bulk transfer TCP-Reno flow among exogenous traffic. A flow may represent the transfer of a large data file as well as the global TCP traffic from one ftp server to another for instance. This model also provides various results for the successive TCP phases.

The paper is organized as follows. The TCP-Reno mechanisms are reviewed in Section 2 and modeled in Section 3 with a discrete-time Markov chain based on the notion of *rounds*. Expressions for the send rate and for the goodput are obtained in Section 4 and Section 5, in which our numerical results are discussed. Section 6 shows the importance of slow start phases in terms of duration and of number of segments sent. Section 7 eventually draws a conclusion.

2. Description of TCP

TCP is a reliable flow control protocol for connection oriented links, see [10,19]. In this protocol, network congestion, identified by packets loss, is detected by the lack of packet acknowledgments, leading the protocol to a modification of the transmission throughput.

Indeed, each successfully transmitted packet is validated and confirmed to the source by a small packet called ACK (*ACKnowledge*) which contains the sequence number of the next expected byte and a receiver's maximum window size giving information about its buffer occupancy. So as not to unnecessarily load the network, the receiver sometimes waits for more data to acknowledge before sending an ACK. Those ACKs are thus called *delayed* ACKs. The number b of segments validated per ACK is typically equal to 1 or 2. A timer T_s will set the departure of an ACK if no new data is to be ACKed.

There are two kinds of loss detection:

- detection by *time-out* (TO): if no ACK is received for byte number n before the expiry of a timer T_0, then a *time-out* occurs. The segment starting with byte n is considered lost and is thus retransmitted, and no more data is sent until byte n is ACKed;
- detection by the arrival of *three duplicate* ACKs (TD): if a segment beginning with byte n is lost but some following segments are received, each of these will generate an ACK requesting byte n, that is one ACK requesting byte n and successive duplicate ACKs. The reception of the third duplicate ACK will halve the window and generate the segment retransmission. In fact, duplicate ACKs can occur when disordered segments are received, and the arrival of one or two duplicate ACKs is not considered as a proof of loss.

TCP is based on a sliding window dynamic. The window, initialized to 1, gives the number of bytes that can be sent before receiving any ACK. Each time an ACK arrives, the window slides to the right to release as many bytes as the ACK validates into the network. The function of TCP is to modify the window size W^c (in segments) according to the algorithm presented below and described in the RFC2001 [20].

First, TCP-Reno consists in three phases depending on loss events and on the comparison of the congestion window size W^c to the *slow start threshold* W^{th}. If a TD loss occurs, then $W^{th} := \max(\lfloor W^c/2 \rfloor, 2)$ and $W^c := \max(\lfloor W^c/2 \rfloor, 1)$, then starts a congestion avoidance phase. If a TO loss occurs, then $W^{th} := \max(\lfloor W^c/2 \rfloor, 2)$, $W^c := 1$, and a time-out phase starts.

Time-out, slow start and congestion avoidance operate as follows:

- *time-out* (to): just after a TO loss detection, the apparently lost segment is retransmitted. After each retransmission failure, the timer value doubles (from T_0 to $2T_0$, $4T_0$, $8T_0$, ...) until $64T_0$, and then remains constant (and gets back to T_0 at the end of this time-out period, that is when the corresponding ACK arrives).
- *slow start* (ss): starts after a time-out recovery and lasts as long as $W^c < W^{th}$. During slow start, $W^c := W^c + 1$ each time an ACK is received (b segments ACKed). If the whole window is successfully transmitted, then it generates $\lceil W^c/b \rceil$ ACKs, where $\lceil x \rceil$ denotes the smallest integer $\geq x$. For $b = 1$, a window of size W^c will thus generate W^c ACKs, so it will grow from W^c to $2W^c$. Consequently, the congestion window grows exponentially during the slow start phase;
- *congestion avoidance* (ca): each ACK reception adds $1/W^c$ segments to the window size, so that the ACKment of the whole window increases W^c by $1/b$. Consequently, the congestion window grows linearly (of one segment every b rounds) during the congestion avoidance phase.

3. The model

If the dispatch duration of all the segments and of all the ACKs held in a given window is negligible compared to the *round trip time* (RTT), then we can justify the following definition of *round* given in [9,16,17]: a *round* is the period of time between the departure of the first segment of the current window and the arrival of its ACK. The duration of a round is close to the round trip time when the delayed ACK timer T_s is small compared to the RTT.

3.1. Definition

We aim at modeling the window behavior using a homogeneous discrete-time Markov chain $X = (X_n)_{n\geq1}$ with two components $X_n = (W_n^c, W_n^{th})$. The first component W_n^c denotes, when positive, the window size during the n-th round. The null value for W_n^c is used to represent the time-out period. The second component W_n^{th} denotes the value of the slow start threshold during the n-th round. We denote by W_{max} the maximum window size, which is the receiver's buffer capacity indicated in the ACKs (when W_n^c reaches W_{max} it remains constant until the next loss). The description the state space of this Markov chain is given, more formally, by:

- $X_n = (i, j)$ with $i \in \{1, \ldots, W_{max}\}$ and $j \in \{2, \ldots, \lfloor W_{max}/2 \rfloor\}$ when the current window size is i and the slow start threshold is j,
- $X_n = (0, j)$ with $j \in \{2, \ldots, \lfloor W_{max}/2 \rfloor\}$ when the connection is in a time-out period with threshold j.

As long as $W_n^c = i \geq 1$, a transition of the Markov chain represents one round and thus lasts RTT seconds. In order to make the mean duration (in seconds) of a time-out period $E[T_{to}]$ equal to RTT times the mean number of successive visits to state $(0, j)$, we define the two following transitions from each state $(0, j)$, $j = 2, \ldots, \lfloor W_{max}/2 \rfloor$:

- from $(0, j)$ to $(1, j)$ with probability p_0 at the end of a time-out period,
- from $(0, j)$ to $(0, j)$ with probability $1 - p_0$ otherwise,

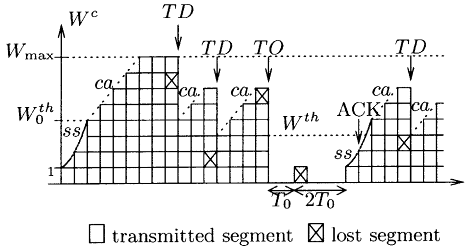

Fig. 1. Example of congestion window evolution.

with p_0 = RTT/$E[T_{\text{to}}]$. In Section 4.5, we give the expression of $E[T_{\text{to}}]$ as a function of RTT, p and T_0.

The state space E of this Markov chain is a subset of the set E' defined by $E' = \{0, \ldots, W_{\max}\} \times \{2, \ldots, \lfloor W_{\max}/2 \rfloor\}$. We can notice that for $W_{\max}$ = 10, 50, 100, 200, the set E' contains, respectively, 44, 1224, 4949 and 19,899 states.

A simple example of the beginning of a connection is given in Figs. 1 and 2 where we take $W_0^{\text{th}} = 4$ segments, $W_{\max} = 8$ and $b = 1$. It can be noted in Fig. 1, that, for instance, state (3, 4) will never be reached. This is due to the fact that the window sizes reached in the slow start phase are for:

- $b = 1 : 1, 1 + \lceil 1/b \rceil = 2, 2 + \lceil 2/b \rceil = 4, 8, 16, 32, \ldots$
- $b = 2 : 1, 1 + \lceil 1/b \rceil = 2, 2 + \lceil 2/b \rceil = 3, 5, 8, 12, \ldots$

This example leads to the following partitioning for the state space of the Markov chain, which is represented in Fig. 2. The state space E is written as $E = E^0 \cup A \cup B$ where

- $E^0 = \{(0, j) | 2 \leq j \leq \lfloor W_{\max}/2 \rfloor\}$,
- $B = \{(i, j) | 2 \leq j \leq i \leq W_{\max}$ and $j \leq \lfloor W_{\max}/2 \rfloor\}$,
- $A = \{(i, j) | 1 \leq i < j \leq \lfloor W_{\max}/2 \rfloor$ and $\exists n \geq 0\}$ such that $i = f^{[n]}(1)\}$, where $f(w) = w + \lceil w/b \rceil$, $f^{[0]}(w) = w$, and $f^{[n]} = f^{[n-1]} \circ f$, for $n \geq 1$.

The partition shown in Fig. 2 is in fact a partition of the state space E' and the set A contains the reachable states of A' during the slow start phase.

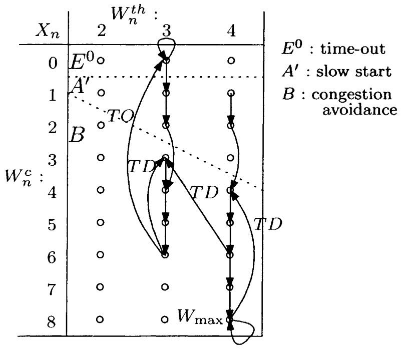

Fig. 2. Markov chain transitions and partitioning.

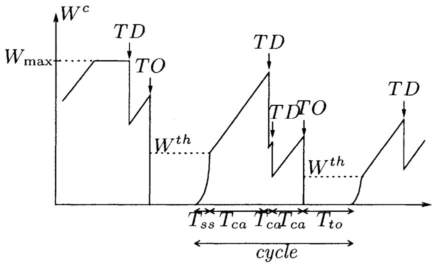

Fig. 3. Description of a cycle.

This discrete-time Markov chain is irreducible and aperiodic. It is thus ergodic and its stationary distribution π is the unique distribution verifying $\pi P = \pi$ where P is the transition probability matrix, which is given in Section 3.3.

3.2. Cyclic behavior

In what follows, we consider the Markov chain in stationary regime and we assume that the source behaves as a saturated one, which means that there are always packets waiting for transmission.

In such a context, an observation of the congestion window size shows a cyclic evolution, consisting in one slow start phase followed by several congestion avoidance phases separated by TD losses, and then a TO loss starting a time-out period at the end of which a new cycle begins (see Fig. 3). We denote, respectively, by

- T_{to}, T_{ss} and T_{ca} the duration of a time-out period, a slow start phase and a congestion avoidance phase,
- d_{to}, d_{ss} and d_{ca} the number of segments sent during the periods T_{to}, T_{ss} and T_{ca},
- $T_{E^0}^{\text{back}}$ the time between a time-out recovery and the next TO loss,
- $d_{E^0}^{\text{back}}$ the number of segments sent during $T_{E^0}^{\text{back}}$,
- N_{loss} the mean number of loss detections per cycle,
- ρ the connection throughput, more precisely the mean transmission rate or *send rate*, which takes into account all segments that have left the source, including lost segments and retransmissions (ρ is the input rate seen by the network).

Observing Fig. 3 and because of the cyclic window evolution, we would write the throughput ρ as

$$\frac{E[d_{E^0}^{\text{back}}] + E[d_{\text{to}}]}{E[T_{E^0}^{\text{back}}] + E[T_{\text{to}}]}. \tag{1}$$

However, this formula does not take into account the residual rounds that appear after each loss and which are presented in Section 4.1 together with the expression of ρ.

3.3. The transition probabilities

We assume that losses only occur in the direction from the sender to the receiver (no loss of ACKs) and that any segment has a fixed probability p to get lost. More precisely, the random variable defined by the number of consecutive segments that are transmitted before loss has a geometric distribution with parameter $1 - p$.

Let us first suppose that the connection is in slow start, i.e. $W_n^{\mathrm{c}} = i < j = W_n^{\mathrm{th}}$. As long as the Markov chain remains in slow start, the congestion window increases by one segment each time an ACK is received. And because $\lceil W_n^{\mathrm{c}}/b \rceil$ segments are acknowledged for the whole round, $W_{n+1}^{\mathrm{c}} = W_n^{\mathrm{c}} + \lceil W_n^{\mathrm{c}}/b \rceil = \lceil \gamma W_n^{\mathrm{c}} \rceil$ with $\gamma = 1 + 1/b$. In the following propositions, we give expressions for the non-zero transition probabilities of the Markov chain. These expressions being easy to obtain, we omit the proofs.

Proposition 1. *For* $1 \leq i < j \leq \lfloor W_{\max}/2 \rfloor$ *, we get*

- $P_{(i,j)(\lceil \gamma i \rceil, j)} = (1 - p)^i$: *no loss occurs,*
- $P_{(i,j)(0,\max(\lfloor i/2 \rfloor,2))} = (1 - (1 - p)^i)q_i$: *a TO loss occurs,*
- $P_{(i,j)(\max(\lfloor i/2 \rfloor,1),\max(\lfloor i/2 \rfloor,2))} = (1 - (1 - p)^i)(1 - q_i)$: *a TD loss occurs,*

where q_i (*computed in* Section 4.2) *denotes the probability that a loss is due to time-out when* $W^{\mathrm{c}} = i$.

Suppose now that the transmission is in congestion avoidance in state (i, j), i.e. $W_n^{\mathrm{c}} = i \geq j = W_n^{\mathrm{th}}$.

Proposition 2. *Observing that congestion avoidance globally raises the window size by* $1/b$, *i.e. by one segment every b rounds, then for* $1 \leq j \leq i < W_{\max}$,

- $P_{(i,j)(i,j)} = (1 - p)^i(1 - 1/b)$: *no loss occurs,*
- $P_{(i,j)(i+1,j)} = (1 - p)^i(1/b)$: *no loss occurs,*
- $P_{(i,j)(0,\max(\lfloor i/2 \rfloor,2))} = (1 - (1 - p)^i)q_i$: *a TO loss occurs,*
- $P_{(i,j)(\max(\lfloor i/2 \rfloor,1),\max(\lfloor i/2 \rfloor,2))} = (1 - (1 - p)^i)(1 - q_i)$: *a TD loss occurs.*

Note that in order to get the model more accurate about the raise of one segment every b rounds, we should decompose the Markov chain state (i, j) into b new states, say $(i, j, 1), (i, j, 2), \ldots, (i, j, b)$, but, first that would of course significantly increase the Markov chain size (even for $b = 2$) and secondly, that would not change the measures of interest since the stationary distribution on the state space E remains the same after such a transformation.

Proposition 3. *Similarly, for each j we have*

- $P_{(W_{\max},j)(W_{\max},j)} = (1 - p)^{W_{\max}}$: *no loss occurs,*
- $P_{(W_{\max},j)(0,\max(\lfloor W_{\max}/2 \rfloor,2))} = (1 - (1 - p)^{W_{\max}})q_{W_{\max}}$: *a TO loss occurs,*
- $P_{(W_{\max},j)(\max(\lfloor W_{\max}/2 \rfloor,1),\max(\lfloor W_{\max}/2 \rfloor,2))} = (1 - (1 - p)^{W_{\max}})(1 - q_{W_{\max}})$: *a TD loss occurs.*

As explained in Section 3, we define the transition probabilities in time-out.

Proposition 4. *For each j, we have*

- $P_{(0,j)(0,j)} = 1 - (\mathrm{RTT}/E[T_{\mathrm{to}}])$: *no acknowledgment yet,*
- $P_{(0,j)(1,j)} = \mathrm{RTT}/E[T_{\mathrm{to}}]$: *the acknowledgment has arrived.*

The expression of $E[T_{\mathrm{to}}]$ as a function of the timer T_0 and the loss probability p is computed in Section 4.5.

The shape of the transition probability matrix P and the regions corresponding to the different types of losses are shown in Fig. 4.

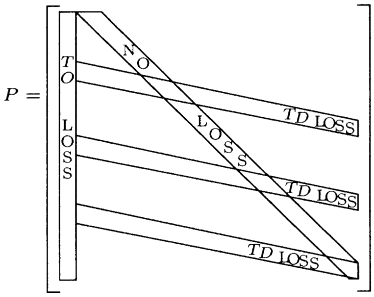

Fig. 4. Link between the transition matrix *P* and TCP.

4. Computation of the throughput

4.1. Residual rounds

First, we make the assumption that in a given round, the loss of one segment leads to the loss of the following segments (correlated losses). This should be the case in a high-speed network for instance. Moreover, in the round where the loss takes place, if *k* segments are however transmitted before congestion, then those segments will generate ACKs and the window will slide. This means that *k* new segments are transmitted in the next round, which is called the *residual round*.

This behavior is shown in Fig. 5, which depicts the case where the last segment sent during the residual round is lost. We consequently introduce the following notations:

- d_{rr}: number of segments sent in a residual round,
- p_{rr}: probability that a loss is followed by a residual round, that is probability that a residual round is not empty.

We can now give the expression of the send rate ρ.

Proposition 5. *The send rate* ρ *is given by*

$$\rho = \frac{E[d_{\text{to}}] + E[d_{E^0}^{\text{back}}] + N_{\text{loss}}E[d_{\text{rr}}]}{E[T_{\text{to}}] + E[T_{E^0}^{\text{back}}] + \text{RTT}(N_{\text{loss}} - 1)p_{\text{rr}}}. \tag{2}$$

Proof. The first terms of expression (2) correspond to Eq. (1). The last terms, where N_{loss} appears, are due to the residual rounds. In counting the mean number of segments transmitted during a cycle, we also

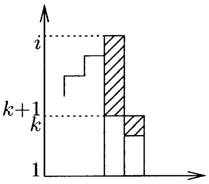

Fig. 5. The residual round.

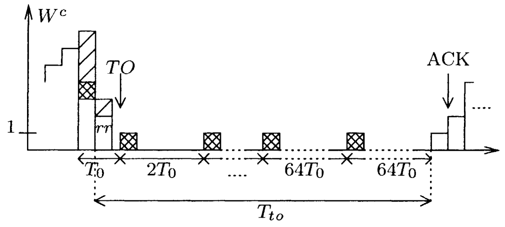

Fig. 6. Detail of a time-out period.

need to take into account the mean number of segments constituting the residual rounds generated by the N_{loss} loss detections. This mean number of segments is equal to $N_{\text{loss}}E[d_{\text{rr}}]$.
For what concerns the mean cycle duration, it is increased by $p_{\text{rr}}\text{RTT}$ for each of the $(N_{\text{loss}} - 1)$ TD losses, because the TO loss residual round is taken into account in the next time-out period, as shown in Fig. 6. □

The expressions of N_{loss}, p_{rr}, $E[d_{\text{rr}}]$, $E[d_{\text{to}}]$, $E[T_{\text{to}}]$, $E[T_{E^0}^{\text{back}}]$, $E[d_{E^0}^{\text{back}}]$ are given in Eqs. (4)–(8) and (11)–(12).

Remark 6. Let us denote by n_{ca} the number of ca phases in a cycle and $N_{\text{ca}} = E[n_{\text{ca}}]$. Whereas, it is clear that $E[T_{E^0}^{\text{back}}] = E[T_{\text{ss}}] + E[n_{\text{ca}}T_{\text{ca}}]$, our numerical results have shown that $E[T_{E_0}^{\text{back}}]$ is very closed to $E[T_{\text{ss}}] + N_{\text{ca}}E[T_{\text{ca}}]$, which means that n_{ca} and T_{ca} can be considered as independent. The same results hold for variables n_{ca} and d_{ca}.

4.2. TO-type losses proportion

Now that we introduced residual rounds, we are able to understand how a loss might be a TO loss and not a TD loss, and thus to compute probabilities q_t that a loss is due to TO when $W^c = i$, which are necessary for the evaluation of transition probabilities.

Proposition 7. *The probability q_i that a loss is due to TO when $W^c = i$ is given by*: $q_i = 1$ *if* $i \leq 2b + 1$ *and*

$$q_i = \frac{(1-(1-p)^{2b+1})(1+(1-p)^{2b+1}-(1-p)^i)}{1-(1-p)^i}, \quad \textit{if } i \leq 2b+1. \tag{3}$$

Proof. Using the notation in Fig. 5, we have

- If $i \leq 2b + 1$ then $k \leq 2b$ hence no TD loss can happen (three duplicate ACKs need $b + b + 1 = 2b + 1$ segments to be received). In this case, the loss is necessarily due to TO, i.e. $q_i = 1$.
- If $i \geq 2b + 2$ then
  - ○ if $k \leq 2b$: similarly, only a TO loss can occur;
  - ○ if $k \geq 2b + 1$: there is a TO loss only when less than $2b + 1$ segments from the residual round arrive at destination (the $2b + 1$ first segments are not all received), i.e. the lth segment from the residual round gets lost, with $1 \leq l \leq 2b + 1$.

Thus, if we denote by L_{k+1} the event corresponding to the loss of the (k + 1)th segment, we get

$$q_i = P(\mathrm{TO}|W^{\mathrm{c}} = i\,\&\,\mathrm{loss}) = \sum_{k=0}^{i-1} q_{i,k} P(L_{k+1}|W^{\mathrm{c}} = i\,\&\,\mathrm{loss})$$

where

$$q_{i,k} = P(\mathrm{TO}|W^{\mathrm{c}} = i\,\&\,L_{k+1}) = \begin{cases} 1, & \text{if } k \le 2b, \\ 1-(1-p)^{2b+1}, & \text{if } k \ge 2b+1, \end{cases}$$

and $P(L_{k+1}|W^{\mathrm{c}} = i\,\&\,\mathrm{loss}) = ((1-p)^k p)/(1-(1-p)^i)$. Eq. (3) then follows after some algebra. □

4.3. Mean number of losses per cycle

Proposition 8. *The mean number N_{loss} of loss detections per cycle is given by*

$$N_{\mathrm{loss}} = \frac{1-\sum_{(i,j)\in E}(1-p)^i \pi(i,j)}{\sum_{(i,j)\in E} q_i(1-(1-p)^i)\pi(i,j)}. \tag{4}$$

Proof. Each cycle (see Fig. 3) is composed of several TD losses and only one TO loss. Thus, we have

$$\frac{1}{N_{\mathrm{loss}}} = P(\mathrm{TO}|\mathrm{loss}\,\&\,W^{\mathrm{c}} \ge 1) = \sum_{i=1}^{W_{\max}} q_i p_{i|\mathrm{loss}}$$

where

$$p_{i|\mathrm{loss}} = P(W^{\mathrm{c}} = i|\mathrm{loss}\,\&\,W^{\mathrm{c}} \ge 1) = \frac{P(\mathrm{loss}|W^{\mathrm{c}} = i)P(W^{\mathrm{c}} = i|W^{\mathrm{c}} \ge 1)}{P(\mathrm{loss}|W^{\mathrm{c}} \ge 1)}$$

$$= \frac{(1-(1-p)^i(P(W^{\mathrm{c}} = i)/P(W^{\mathrm{c}} \ge 1)))}{\sum_{i=1}^{W_{\max}}(1-(1-p)^i)(P(W^{\mathrm{c}} = i)/P(W^{\mathrm{c}} \ge 1))} = \frac{(1-(1-p)^i)\sum_{j=2}^{\lfloor W_{\max}/2\rfloor}\pi(i,j)}{\sum_{(i,j)\in E}(1-(1-p)^i)\pi(i,j)}.$$

□

4.4. The weight of residual rounds

Proposition 9. *The probability p_{rr} that a residual round appears after loss is given by*

$$p_{\mathrm{rr}} = 1 - p\frac{1-\sum_{j=2}^{\lfloor W_{\max}/2\rfloor}\pi(0,j)}{1-\sum_{(i,j)\in E}(1-p)^i\pi(i,j)}. \tag{5}$$

Proof. Let K be the random variable equal to the number of segments sent before loss in the round in which that loss occurred (see Fig. 5, in which we have drawn the case $K = k$). We thus have

$$p_{\mathrm{rr}} = P(K \neq 0 | \mathrm{loss}\,\&\, W^{\mathrm{c}} \geq 1) = \sum_{i=1}^{W_{\max}} P(K \neq 0 | W^{\mathrm{c}} = i\,\&\,\mathrm{loss}) p_{i|\mathrm{loss}}$$

$$= \sum_{i=1}^{W_{\max}} \left(1 - \frac{p}{1-(1-p)^i}\right) p_{i|\mathrm{loss}},$$

which leads to Eq. (5) using the expression of $P_{i|\mathrm{loss}}$ given in the proof of Proposition 8. □

Proposition 10. *The mean number of segments $E[d_{\mathrm{rr}}]$ that are sent in a residual round is given by*

$$E[d_{\mathrm{rr}}] = \frac{1-p}{p} - \frac{\sum_{(i,j)\in E} i(1-p)^i \pi(i,j)}{1 - \sum_{(i,j)\in E} (1-p)^i \pi(i,j)}. \tag{6}$$

Proof. As above, we denote by K the random variable equal to the number of segments sent before loss in the round in which that loss occurred (see Fig. 5). We have

$$E[d_{\mathrm{rr}}] = E[K | \mathrm{loss}\,\&\, W^{\mathrm{c}} \geq 1] = \sum_{i=1}^{W_{\max}} E[d_{\mathrm{rr}} | i] p_{i|\mathrm{loss}}$$

where

$$E[d_{\mathrm{rr}} | i] = E[K | W^{\mathrm{c}} = i\,\&\,\mathrm{loss}] = \sum_{k=0}^{i=1} k \frac{(1-p)^k p}{1-(1-p)^i} = \left(\frac{1-p}{p}\right) \frac{1-(1-p)^i - ip(1-p)^{i-1}}{1-(1-p)^i}.$$

Eq. (6) is then obtained using the expression of $P_{i|\mathrm{loss}}$ given in the proof of Proposition 8. □

4.5. Time-out study

The behavior of TCP during a time-out period is illustrated in Fig. 6, where rr denotes the residual round (see also Fig. 5). The following result can be found in [17].

Proposition 11. *The mean number of segments sent during a time-out period and the mean duration of a time-out period are given by*

$$E[d_{\mathrm{to}}] = \frac{p}{1-p} \quad (\textit{geometric distribution of segments loss}), \tag{7}$$

and

$$E[T_{\mathrm{to}}] = T_0 \frac{1 + p + 2p^2 + 4p^3 + 8p^4 + 16P^5 + 32p^6}{1-p} - \mathrm{RTT}. \tag{8}$$

4.6. Between two time-out periods

In the following remark, we briefly recall some results on sojourn times in Markov chains. These results have been obtained in [18].

Remark 12. Consider an irreducible discrete-time Markov chain with finite state space E, transition probability matrix P and stationary probability distribution π. We denote by $\mathbb{1}$ the column vector with all the entries equal to 1. Let F be a proper subset of E and F' the complementary subset $E - F$. The partition F, F' of E induces the following decomposition of P, π and $\mathbb{1}$:

$$P = \begin{pmatrix} P_F & P_{F,F'} \\ P_{F',F} & P_{F'} \end{pmatrix}, \qquad \pi = (\pi_F \quad \pi_{F'}) \qquad \text{and} \qquad \mathbb{1} = \begin{pmatrix} \mathbb{1}_F \\ \mathbb{1}_{F'} \end{pmatrix}.$$

If v_i denotes the stationary probability that a sojourn in F initiates in state i ($i \in F$) and v the row vector composed of the v_i, then

$$v = \frac{\pi_F(I - P_F)}{\pi_F(I - P_F)\mathbb{1}_F} = \frac{\pi_{F'} P_{F',F}}{\pi_{F'} P_{F',F}\mathbb{1}_F}, \tag{9}$$

where I is the identity matrix of dimension given by the context. Moreover, for every $i \in F$, let $N_{i,F}$ be the number of visits to state i during a sojourn in F and let r_i be any real number. If we denote by r_F the column vector composed of the r_i and, by C_F the random, variable $C_F = \sum_{i \in F} r_i N_{i,F}$, we easily get

$$E[C_F] = v(I - P_F)^{-1} r_F = \frac{\pi_F r_F}{\pi_{F'} P_{F',F}\mathbb{1}_F}. \tag{10}$$

Using these results, we have the following proposition.

Proposition 13. *The mean time $E[T_{E^0}^{\text{back}}]$ between the end of a time-out period* (*the beginning of slow start*) *and the next TO loss is given by*

$$E[T_{E^0}^{\text{back}}] = \frac{\text{RTT}}{p_0}\left(\frac{1}{\sum_{j=2}^{\lfloor W_{\max/2}\rfloor} \pi(0, j)} - 1\right). \tag{11}$$

Proof. $E[T_{E^0}^{\text{back}}]$ is RTT times the mean time spent by the Markov chain in subset $A \cup B$. Following Remark 12, Eq. (10), we have

$$E[T_{E^0}^{\text{back}}] = \text{RTT} \times E[C_{A\cup B}] = \text{RTT}\frac{\pi_{A\cup B} r_{A\cup B}}{\pi_{E^0} P_{E^0, A\cup B}\mathbb{1}_{A\cup B}}$$

where $r_{A\cup B} = \mathbb{1}_{A\cup B}$. We thus have

$$E[T_{E^0}^{\text{back}}] = \text{RTT}\frac{\sum_{(i,j)\in A\cup B} \pi(i, j)}{\sum_{j=2}^{\lfloor W_{\max}/2\rfloor} p_0 \pi(0, j)} = \frac{\text{RTT}}{p_0}\frac{1 - \sum_{(i,j)\in E^0} \pi(i, j)}{\sum_{j=2}^{\lfloor W_{\max}/2\rfloor} \pi(0, j)} = \frac{\text{RTT}}{p_0}\frac{1 - \sum_{j=2}^{\lfloor W_{\max}/2\rfloor} \pi(0, j)}{\sum_{j=2}^{\lfloor W_{\max}/2\rfloor} \pi(0, j)},$$

where the last equality derives from the fact that E^0 is the subset of states $(0, j)$, $j = 2, \ldots, \lfloor W_{\max}/2\rfloor$. □

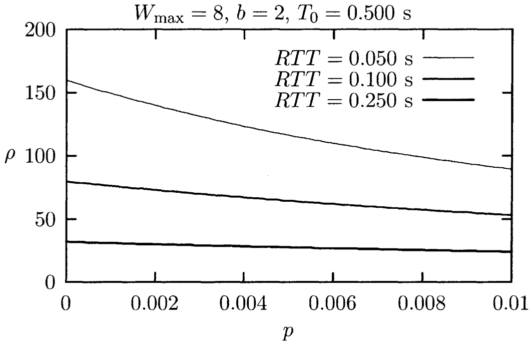

Fig. 7. Send rate ρ for different values of RTT.

Proposition 14. *The mean number* $E[d_{E^0}^{\text{back}}]$ *of segments sent between the end of a time-out period and the next TO loss is given by*

$$E[d_{E^0}^{\text{back}}] = \frac{\sum_{(i,j)\in A\cup B} i\pi(i,j)}{p_0\sum_{j=2}^{\lfloor W_{\max}/2\rfloor}\pi(0,j)}. \tag{12}$$

Proof. $E[d_{E^0}^{\text{back}}]$ is the mean number of segments sent during a sojourn in $A \cup B$. We thus have $E[d_{E^0}^{\text{back}}] = E[C_{A\cup B}]$ where the entry (i, j) of vector $r_{A\cup B}$ is now the number of segments sent when the Markov chain is in state $(i, j) \in A \cup B$, that is $r_{(i,j)} = i$, for every $(i, j) \in A \cup B$. Following Remark 12, Eq. (10), the rest of the proof is similar to that of Proposition 13. □

4.7. Numerical results

In Fig. 7, the send rate ρ gets equal to $W_{\max}$ segments per RTT ($W_{\max}$/RTT segments per second) when loss probability p is close to zero, and converges to zero when p increases. Moreover, the shorter the RTT, the more segments per second (quick acknowledgments) are sent. In Fig. 8, when $W_{\max}$ increases, the

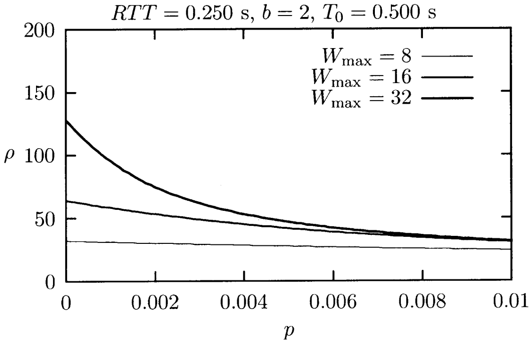

Fig. 8. Send rate ρ for different values of $W_{\max}$.

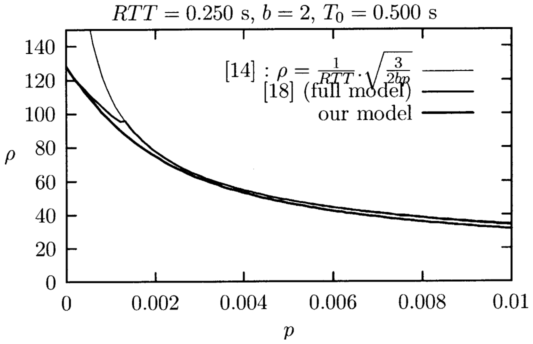

Fig. 9. Comparison to other models for RTT = 0.250 s.

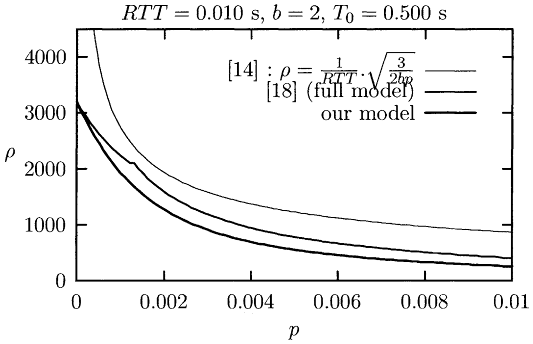

Fig. 10. Comparison to other models for RTT = 0.010 s.

window size can reach higher values and the mean throughput naturally increases too. Note that for small values of the loss probability p, ρ reaches $W_{\max}$/RTT segments per second, and for large values of p, ρ seems to be less dependent on $W_{\max}$. Indeed, for $p = 0.01$, ρ gets close to 20 or 30 segments per second, that is around 6 segments per RTT for $W_{\max} = 8, 16, 32$. Figs. 9 and 10 provide a comparison to simpler models [14,17] which have been validated from both simulations and real traffic measurements which we do not report here but which can be found in [9,14,17]. Note that the throughput of our model, evaluated with less simplifications, is lower than the one obtained by the authors of [14,17]. But the higher the RTT, the closer the different models are.

5. Computation of the goodput

In this section, we study the *goodput* (or *output rate*) of the connection, defined as the mean number of segments successfully transmitted per second. The goodput thus represents the throughput seen by the receiver.

If we denote by $d_{E^0}^{\text{back},0}$ the number of segments successfully transmitted during $T_{E^0}^{\text{back}}$, and by d_{rr}^0 the number of segments successfully transmitted in a residual round, then the connection goodput, denoted by ρ_0, is given by the following proposition.

Proposition 15. *The goodput ρ_0 can be expressed as*

$$\rho_0 = \frac{E[d_{E^0}^{\text{back},0}] + N_{\text{loss}} E[d_{\text{rr}}^0]}{E[T_{\text{to}}] + E[T_{E^0}^{\text{back}}] + \text{RTT}(N_{\text{loss}} - 1) p_{\text{rr}}}. \tag{13}$$

Proof. The goodput is computed as the mean number of segments successfully transmitted during a cycle over the mean duration of a cycle. The difference with ρ is thus confined to the numerator, and during a time-out period, no segment is received (as shown in Fig. 6, the reception of the retransmitted segment is included in the slow start phase that thus begins). □

In the next subsection, we give the expressions of $E[d_{E^0}^{\text{back},0}]$ and $E[d_{\text{rr}}^0]$.

5.1. Successfully transmitted segments

Proposition 16. *The mean number $E[d_{\text{rr}}^0]$ of segments transmitted in a residual round is given by*

$$E[d_{\text{rr}}^0] = \frac{1-p}{p} - \frac{1-p}{p(2-p)} \frac{1 - \sum_{(i,j) \in E} (1-p)^{2i} \pi(i, j)}{\sum_{(i,j) \in E} (1-p)^i \pi(i, j)}.$$

Proof. If the random variable K is still defined as the number of segments sent before loss in the round in which the loss occured (see Fig. 5), and if L denotes the random variable such that $L = l$ when, in the residual round, l segments are transmitted and the $(l + 1)$th gets lost, we have

$$E[d_{\text{rr}}^0] = E[L|\text{loss}\,\&\, W^{\text{c}} \geq 1] = \sum_{k=0}^{W_{\max}-1} E[L|K = k \,\&\, \text{loss}] P(K = k|\text{loss}\,\&\, W^{\text{c}} \geq 1)$$

with

$$E[L|K = k \,\&\, \text{loss}] = \sum_{l=0}^{k-1} l(1-p)^l p + k(1-p)^k = \frac{(1-p)(1-(1-p)^k)}{p}$$

and

$$P(K = k|\text{loss}\,\&\, W^{\text{c}} \geq 1) = \sum_{i=k+1}^{W_{\max}} P(K = k|W^{\text{c}} = i \,\&\, \text{loss}) p_{i|\text{loss}} = \sum_{i=k+1}^{W_{\max}} \frac{(1-p)^k p}{1-(1-p)^i} p_{i|\text{loss}}.$$

Thus, using the expression of $P_{i|\text{loss}}$ given in the proof of Proposition 8

$$E[d_{\text{rr}}^0] = \frac{\sum_{(i,j)\in E}(1-p)\left(\sum_{k=0}^{i-1}(1-p)^k - \sum_{k=0}^{i-1}((1-p)^2)^k\right)\pi(i,j)}{\sum_{(i,j)\in E}(1-(1-p)^i)\pi(i,j)}$$

$$= \frac{1-p}{p}\frac{\sum_{(i,j)\in E}(1-(1-p)^i)\pi(i,j)}{\sum_{(i,j)\in E}(1-(1-p)^i)\pi(i,j)} - \frac{1-p}{p(2-p)}\frac{\sum_{(i,j)\in E}(1-(1-p)^{2i})\pi(i,j)}{\sum_{(i,j\in E)}(1-(1-p)^i)\pi(i,j)}$$

$$= \frac{1-p}{p} - \frac{1-p}{p(2-p)}\frac{1-\sum_{(i,j)\in E}(1-p)^{2i}\pi(i,j)}{1-\sum_{(i,j)\in E}(1-p)^i\pi(i,j)}.$$

□

Proposition 17. *The mean number of segments $E[d_{E^0}^{\text{back},0}]$ successfully transmitted between the end of a time-out period and the next TO loss is given by*

$$E[d_{E^0}^{\text{back},0}] = \frac{1-p}{pp_0}\frac{\sum_{(i,j)\in A\cup B}(1-(1-p)i)\pi(i,j)}{\sum_{j=2}^{\lfloor W_{\max}/2\rfloor}\pi(0,j)}.$$

Proof. $E[d_{E^0}^{\text{back},0}]$ is the mean number of segments successfully transmitted during a sojourn in $A\cup B$. We thus have $E[d_{E^0}^{\text{back},0}] = E[C_{A\cup B}]$ where the entry (i,j) of vector $r_{A\cup B}$ is the number of segments successfully transmitted when the Markov chain is in state $(i,j)\in A\cup B$, that is

$$r_{(i,j)} = \sum_{k=0}^{i-1} k(1-p)^k p + i(1-p)^i = \frac{1-p}{p}(1-(1-p)^i).$$

Following Eq. (10) in Remark 12, the rest of the proof is similar to that of Proposition 13. □

5.2. Numerical results

Comparing Figs. 8 and 11, we notice that ρ and ρ_0 seem to take very close values. It is thus interesting to evaluate the ratio $e=\rho_0/\rho$. This ratio represents the proportion of received segments among the transmitted ones, that is the percentage of "useful data". For this reason, we call e the *efficiency* of the connection.

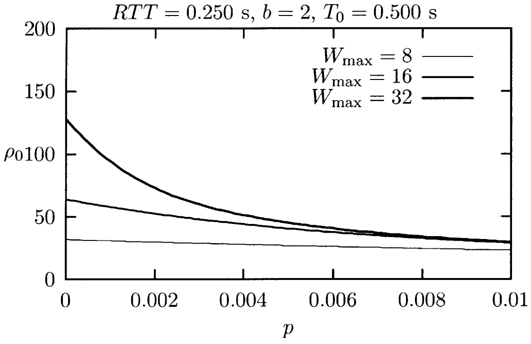

Fig. 11. Goodput ρ_0 for different values of $W_{\max}$.

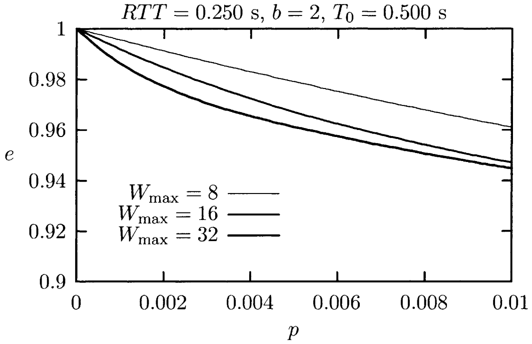

Fig. 12. Efficiency $e = \rho_0/\rho$ for different values of $W_{\max}$.

Our numerical results have shown that e is weakly sensitive to RTT, so the curves representing e for different values of RTT with a given value of $W_{\max}$ will merge.

However, e depends on $W_{\max}$. Indeed, as shown in Fig. 12, the values of $e = \rho_0/\rho$ decrease when $W_{\max}$ increases. It is explained by the correlated losses assumption made at the beginning of Section 4.1. In fact, the higher $W_{\max}$, the more segments are lost in each round where a loss occurs, and each lost segment will generate retransmissions. In other words, the higher the bandwidth (large value of $W_{\max}$), the faster you may transmit data (ρ_0 increases), however this also entails a higher number of retransmissions, which means overloading the network.

6. The importance of slow start

The strength of our model is that it allows us to give a detailed description of the window evolution. In particular, we obtain the expression of $E[T_{\text{ss}}]$, the mean duration of a slow start phase, and of $E[d_{\text{ss}}]$, the mean number of segments sent in a slow start phase.

Proposition 18. *The mean duration $E[T_{\text{ss}}]$ of a slow start phase is*

$$E[T_{\text{ss}}] = \frac{\text{RTT}}{p_0} \frac{\sum_{j=2}^{\lfloor W_{\max}/2 \rfloor} \sum_{i=1}^{j-1} \pi(i, j)}{\sum_{j=2}^{\lfloor W_{\max}/2 \rfloor} \pi(0, j)}.$$

Proof. $E[T_{\text{ss}}]$ is RTT times the mean time spent by the Markov chain in subset A. Following Remark 12, Eq. (10), we have

$$E[T_{\text{ss}}] = \text{RTT} \times E[C_A] = \text{RTT} \frac{\pi_A r_A}{\pi_{E^0 \cup B} P_{E^0 \cup B, A} \mathbb{1}_A}$$

where $r_A = \mathbb{1}_A$. We thus have

$$E[T_{\text{ss}}] = \text{RTT} \frac{\sum_{(i,j) \in A} \pi(i, j)}{\sum_{j=2}^{\lfloor W_{\max}/2 \rfloor} p_0 \pi(0, j)} = \text{RTT} \frac{\sum_{j=2}^{\lfloor W_{\max}/2 \rfloor} \sum_{i-1}^{j-1} \pi(i, j)}{p_0 \sum_{j=2}^{\lfloor W_{\max}/2 \rfloor} \pi(0, j)},$$

Proposition 19. *The mean number $E[d_{ss}]$ of segments sent during a slow start phase is* □

$$E[d_{ss}] = \frac{\sum_{j=2}^{\lfloor W_{\max}/2\rfloor}\sum_{n=1}^{n_j} w_n(1-p)^{d_{n-1}}\pi(0,j)}{\sum_{j=2}^{\lfloor W_{\max}/2\rfloor}\pi(0,j)},$$

where in any slow start phase, w_n is the size of nth round, $d_n = \sum_{k=1}^{n} w_k$ (with $d_0 = 0$) is the number of segments sent during the first n rounds and n_j is the number of rounds needed to reach the slow start threshold j.

Proof. Let us denote by Z_A the state of subset A by which a sojourn in A begins. These states are necessarily the states $(1, j)$ for $j = 1, \ldots, \lfloor W_{\max}/2\rfloor$. From Remark 12, Eq. (9), $P(Z_A = (l, j))$ is equal to the entry $(1, j)$ of the vector $\pi_A(I - P_A)/[\pi_A(I - P_A)\mathbb{1}_A]$, that is

$$P(Z_A = (1,j)) = \frac{[\pi_A(I-P_A)](1,j)}{\pi_A(I-P_A)\mathbb{1}_A} = \frac{[\pi_{E^0}P_{E^0,A}](1,j)}{\pi_{E^0}P_{E^0,A}\mathbb{1}_A} = \frac{p_0\pi(0,j)}{\sum_{j=2}^{\lfloor W_{\max}/2\rfloor} p_0\pi(i,j)}.$$

Now, if the slow start phase initiates by state $(1, j)$ then the maximum number of rounds in that phase is equal to n_j. For $n < n_j$, the w_n segments of the n-th round are sent if no loss has occurred during the $n - 1$ first rounds, that is among the d_{n-1} first segments. Thus

$$E[d_{ss}\,|Z_A = (1,j)] = \sum_{n=1}^{n_j} w_n(1-p)^{d_{n-1}}.$$

The result follows by writing

$$E[d_{ss}] = \sum_{j=2}^{\lfloor W_{\max}/2\rfloor} E[d_{ss}|Z_A = (1,j)]P(Z_A = (1,j)).$$

□

We can notice in Fig. 13 that the proportion of time spent in slow start per cycle depends on $W_{\max}$ since when $W_{\max}$ gets higher, slow start phases can reach higher thresholds and thus last longer (whereas in congestion avoidance, the bigger the window size, the higher the probability of a loss is, thus stopping the congestion avoidance phase). But the main remark is that the duration of a slow start phase may reach 10

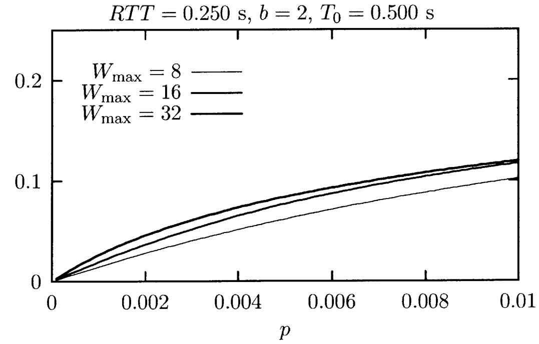

Fig. 13. Proportion of time in each cycle: $E[T_{ss}]/E[T_{E^0}^{\text{back}}]$.

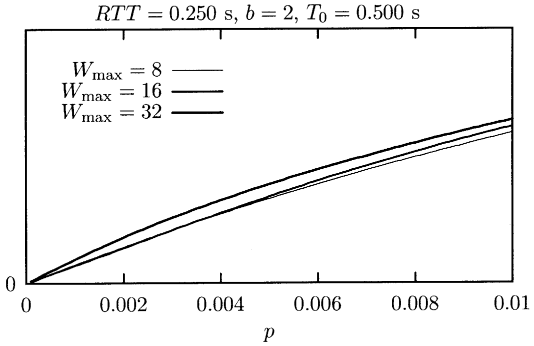

Fig. 14. Number of segments in each cycle: $E[d_{\text{ss}}]/E[d_{E^0}^{\text{back}}]$.

or 15% of $E[T_{E^0}^{\text{back}}]$. Contrary to slow start duration, Fig. 14 shows that the number of segments $E[d_{\text{ss}}]$ sent in slow start remains less than 5% of $E[d_{E^0}^{\text{back}}]$, even for a high $W_{\max}$. This implies that in the expression of ρ given in Relation (2), the numerator will not change a lot if slow start is not taken into account, but the denominator will be significantly reduced, and thus ρ may significantly grow.

The best way of neglecting slow start phase is to consider that this phase is instantaneous. So if we denote by ρ' the throughput obtained without integrating slow start phases, we have

$$\rho' = \frac{E[d_{\text{to}}] + (E[d_{E^0}^{\text{back}}] - E[d_{\text{ss}}]) + N_{\text{loss}} E[d_{\text{rr}}]}{E[T_{\text{to}}] + (E[T_{E^0}^{\text{back}}] - E[T_{\text{ss}}]) + \text{RTT}(N_{\text{loss}} - 1) p_{\text{rr}}}.$$

Fig. 15, shows that ρ' can be up to 12% higher than ρ. The lower the loss probability p, the closer ρ' is to p. But traffic management and bandwidth allocation for instance need a good estimation of ρ, and even a 5% overestimation can lead to severe problems in performance evaluation of other measures of interest.

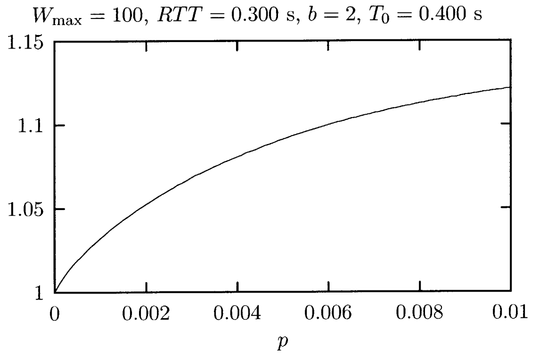

Fig. 15. ρ'/ρ vs. the loss probability p.

7. Conclusion

The main assumption we made is that the connection is established in a high speed and wide area (large RTT) network. Indeed, the time needed to send all segments in congestion window and the time interval between ACKs must be significantly low compared to the round trip time for the identification of separated bursts, called and defined as *rounds*.

Moreover, we supposed that the loss probability p was independent of the window size, because in high capacity networks, the load of a single connection is not responsible for congestion. Concerning loss correlation (when a segment gets lost, all the following ones in the same round also get lost), we apply our model to high capacity and high speed networks with drop-tail routers, in which the connection is not the cause of congestion and packets of a given round arrive in burst in the overflowed router. And despite multiplexing, a router remains full as long as packets of the same window arrive and thus rejects all of them.

With these assumptions, we have been able to obtain an analytical expression for the send rate and for the goodput of a long-term steady-state connection (stationary regime). But our model gives a more precise description of TCP, which allows an accurate study of its performance. Other performance measures can be discussed such as, for instance, the proportion of TO-type losses, the average time interval between two consecutive losses, and the proportion of time during which the window size is at its maximum.

References

[1] A.A. Abou-zeid, M. Azizoglu, S. Roy, Stochastic modeling of a single TCP/IP session over a random loss channel, in: Proceedings of the DIMACS Workshop on Mobile Networks and Computing, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 52, American Mathematical Society, Providence, RI, 1999.

[2] A.A. Abou-zeid, S. Roy, M. Azizoglu, Stochastic modeling of TCP over lossy links, in: Proceedings of INFOCOM 2000, Tel-Aviv, Israel, March, 2000.

[3] O. Ait-Hellal, E. Altman, D. Elouadghiri, M. Erramdani, N. Mikou, Performance of TCP/IP: the case of two controlled sources, in: Proceedings of the ICCC'97, Cannes, France, 1997.

[4] E. Altman, C. Barakat, K. Avrachenkov, A stochastic model of TCP/IP with stationary ergodic random losses, in: Proceedings of ACM-SIGCOMM 2000, Stockholm, Sweden, August 2000.

[5] E. Altman, C. Barakat, K. Avrachenkov, P. Dube, CP in presence of bursty losses, International Conference on Performance and QoS of Next Generation Networks, P&Q Net 2000. Nagoya, Japan, November 2000.

[6] E. Altman, J. Bolot, P. Nain, D. Elouadghiri, M. Erramdani, P. Brown, D. Collange, Performance modeling of TCP/IP in wide-area network, Technical Report RR-3142, INRIA, 1997.

[7] F. Baccelli, D. Hong, TCP is max-plus linear, ACM-SIGCOMM Computer Communication Review 30 (4) (2000) 219–230.

[8] P. Brown, Resource sharing of TCP connections with different round trip times, in: Proceedings of INFOCOM 2000, Tel-Aviv, Israel, March, 2000.

[9] N. Cardwell, S. Savage, T. Anderson, Modeling TCP latency, in: Proceedings of INFOCOM 2000, Tel-Aviv, Israel, March, 2000.

[10] D. Comer, Internetworking with TCP/IP, Volume 1: Principles, Protocols and Architecture, third ed., Prentice-Hall, Englewood Cliffs, NJ, 1995.

[11] P. Hurley, J.Y. Le Boudec, P. Thiran, A note on the fairness of additive increase and multiplicative decrease, in: Proceedings of ITC-16, Edinburgh, Scotland, June, 1999.

[12] A. Kumar, Comparative performance analysis of versions of TCP in a local networks with a lossy link, IEEE/ACM Trans. Networking 6 (4) (1998) 485–498.

[13] T.V. Lakshman, U. Madhow, The performance of TCP/IP for networks with high bandwidth-delay products and random loss, IEEE/ACM Trans. Networking 5 (3) (1997) 336–350.

[14] M. Mathis, J. Semke, J. Mahdavi, T. Ott, The macroscopic behavior of the TCP congestion avoidance algorithm, Comput. Commun. Rev. 27 (3) (1997) 67–82.

[15] V. Misra, W.B. Gong, D. Towsley, Stochastic differential equation modeling and analysis of TCP-window size behavior, in: Proceedings of the Performance'99, Istanbul, Turkey, 1999.

[16] J. Padhye, V. Firoiu, D. Towsley, A stochastic model of TCP Reno congestion avoidance and control, Technical Report 99-02, University of Massachussets, 1999.

[17] J. Padhye, V. Firoiu, D. Towsley, J. Kurose, Modeling TCP throughput: a simple model and its empirical validation, in: Proceedings of the ACM-SIGCOMM'98, Vancouver, Canada, September, 1998.

[18] G. Rubino, B. Sericola, Sojourn times in Markov processes, J. Appl. Probability 26 (1989) 744–756.

[19] W.R. Stevens, TCP/IP Illustrated: Vol. 1 The Protocols, Addison-Wesley, Reading, MA, 1994.

[20] W.R. Stevens, TCP slow start congestion avoidance fast retransmit and fast recovery algorithms, RFC 2001, January 1997.