Complementary analysis and numerical results for an elementary fluid flow model of TCP

Esa Hyytiä[†], Erling Austreim[†,‡] and Peder J. Emstad[†,‡]
Centre for Quantifiable Quality of Service in Communication Systems, Centre of Excellence,[†]
Department of Telematics,[‡] Norwegian University of Science and Technology,
Trondheim, Norway

Abstract

The congestion control algorithm used in Transmission Control Protocol (TCP) is a so-called additive increase multiplicative decrease (AIMD) algorithm. The algorithm and its performance have been studied extensively in the literature. Today, several versions of the TCP exist, which behave slightly differently when congestion occurs. However, for the analytical work it is often necessary to consider idealized models. One such model based on fluid flow approach is proposed in [1], on which we will also focus in this paper. In particular, we study this simplified fluid flow approximation and its generalization by numerical simulations in order to evaluate relationships between different parameters of the model and the resulting performance quantities. Moreover, we also compare the model against an actual implementation of TCP congestion control mechanism. Additionally, we give some complementary analytical results for the model.

Keywords: TCP, AIMD, fluid flow model, embedded Markov chain

I. Introduction

The additive increase multiplicative decrease (AIMD) rate control scheme employed by the transmission control protocol (TCP), has turned out to be very important due to the enormous success of Internet. In its simplicity, an AIMD controlled source increases the sending rate linearly in time until a negative feedback is received. In response to this, the source reduces its sending rate by multiplying the current rate with some constant less than one. In an ideal situation the AIMD scheme is capable of sharing the given bandwidth fairly among the competing flows [2]. The AIMD scheme has been studied actively, see, e.g., [3] and [4].

One particularly important aspect of bandwidth allocation protocols is the fairness, i.e., how the available bandwidth is allocated between the flows sharing the same resources. For example, in TCP/IP networks flows with higher round trip times (RTT) tend to get smaller shares. The most famous results on TCP are on throughput analysis, e.g., the famous square root formula by Floyd and Fall [5], and the more accurate expression by Padhye et al. [6]. These results give the average throughput per flow as a function of the packet loss probability and RTT. The common assumption in TCP throughput analysis has been the independence between congestion periods, i.e., it is assumed that the packet losses occur in fixed time intervals ([5], [6], [7], [8]), or originate from a (non-uniform) Poisson process [9]. A more general approach can be found from [10], where the losses are generated by an arbitrary exogenous random process.

In contrast to the above work, in [1] we have considered an elementary model for the TCP rate control mechanism, where the loss process is explicitly defined by the sending rates of all TCP sources. The aim was to characterize the behavior of the concurrent TCP flows at the microscopic level, e.g., in order to study the interactions between TCP traffic and other traffic flows (e.g., real time voice or video streams). To this end, we have studied a single bottleneck link and made several simplifying assumptions about the rate control mechanism. Firstly, we consider a continuous (fluid flow) model where the sending rates are some non-negative real numbers. Secondly, the decision to send a negative feedback is based on the current arrival rate of the packets into the bottleneck link, not directly on the occupancy level of the buffer (unfinished work). This can be interpreted as a bufferless model for router. Thirdly, we assume a constant delay before sources react on the negative feedback signals (i.e., a constant RTT).

This model can be analyzed in the framework of Markov processes. In particular, considering the time instances when the total sending rate achieves the capacity limit yields an embedded Markov chain (MC),

[†] "Centre for Quantifiable Quality of Service in Communication Systems, Centre of Excellence" is appointed by The Research Council of Norway, and funded by the Research Council, NTNU and UNINETT. (http://www.ntnu.no/Q2S/)

for which we were able to derive steady state distribution for the special case of two TCP flows. For more than two flows we proposed flow aggregation approach, where one flow is chosen as a targeted flow and the rest are assumed to share the remaining capacity equally. Expressions are given for several important performance measures, e.g., the mean and the variance of the total sending rate, the distribution of the window size upon a negative feedback, and a full characterization of intervals between congestion events.

In this paper we will provide new analytical and numerical results for this model. In particular, we argue that the state space of the MC is dense and that any two (macro) states communicate. For the numerical results we use simulations and consider two scenarios. In the first scenario we study the original model with $n > 2$ flows, and also its modified version where we have allowed random variations in the delay of negative feedback. In the second scenario we compare the model to the results obtained by a J-Sim simulator implementing dutifully all the small details of the TCP congestion control mechanism.

The rest of the paper is organized as follows. In Section II we briefly describe the model and restate the results obtained in [1]. In Section III we give some new analytical results for the model. Section IV contains the numerical results and comparisons to a real TCP, and Section V concludes the paper.

II. Model

Next we present the notation and give a brief introduction to the fluid flow model described in detail in [1]. Let n denote the number of TCP sources sharing the same bottleneck link. Each source increases its sending rate linearly in time until the total rate would exceed the capacity of the bottleneck link. At this point of time, a randomly chosen source receives a negative acknowledgement (NAK) and reduces its sending rate according to the multiplicative decrease. Let $r_i(t)$ denote the source i sending rate at time t, and $R(t)$ the total sending rate,

$$R(t) = \sum_i r_i(t).$$

Let α denote the linear increase rate and ν the multiplicative factor, i.e., normally the sources increase their sending rate according to

$$r_i(t + dt) = r_i(t) + \alpha \cdot dt,$$

but upon a negative feedback sent to flow i it will reduce its sending rate according to

$$r_i(t + 0) = \nu \cdot r_i(t).$$

We assume a proportional marking, i.e., upon reaching the capacity limit c the flow to be downsized is chosen randomly with the probabilities proportional to the sending rates r_i,

$$\mathrm{P}\left\{\text{flow } i \text{ is chosen}\right\} = \frac{r_i(t)}{c}.$$

Let Δ_R denote the drop in total sending rate $R(t)$. Thus, $\Delta_R = (1-\nu) \cdot r_i(t)$ with probability of $r_i(t)/c$. Consequently, the mean drop in total rate (conditional to the current state $\mathbf{r}$) is given by

$$\mathrm{E}\left[\Delta_R \,|\, \mathbf{r}\right] = \frac{1-\nu}{c} \cdot \sum_i r_i^2.$$

Finally, let Δ_T denote the time between two consecutive NAK signals (congestion events, packet drops).

A. Normalized Model

Without loss of generality we can consider a so-called *normalized model* defined as follows. First, we assume that the total capacity of the bottleneck link is scaled to one, $c = 1$, so that we have

$$R(t) = \sum_i r_i(t) \leq 1, \qquad \forall \ t,$$

and the probability of choosing flow i upon reaching the capacity limit is simply $r_i(t)$. Moreover, we assume such a time scale that the (total) linear increase rate is equal to 1, i.e., $\alpha = 1/n$, and

$$R(t + dt) = R(t) + dt,$$

(excluding the discontinuities upon reaching the capacity limit). With this choice of the time scale we have

$$\Delta_R = \Delta_T = \Delta.$$

In the rest of the paper we will discuss the normalized model unless otherwise stated.

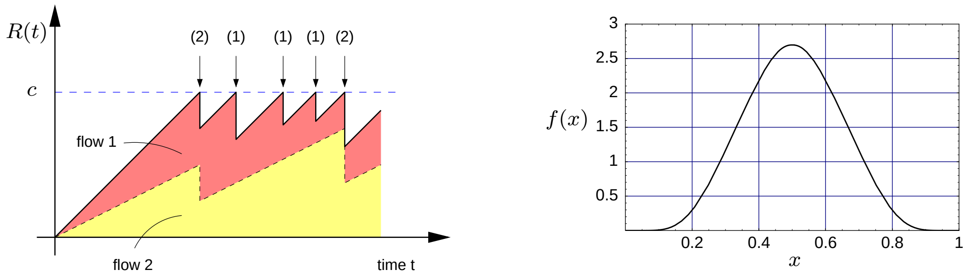


Fig. 1. Sample realization of two competing flows (left), and the corresponding steady state distribution $f(x)$ of the embedded chain (right).

B. Analytical Results

Consider first the total sending rate $R(t)$ at an arbitrary point in time. An arbitrary period starts from a packet drop and the corresponding reduction in the total sending rate is denoted by Δ. Note that the number of flows and the multiplicative factor can be arbitrary, we only assume that as the total sending rate reaches the capacity limit $c = 1$ one or more flows reduce their rate by the amount of Δ. Thus, the sending rate during this period is given by

$$R(t) = t + 1 - \Delta, \qquad 0 \le t \le \Delta.$$

Hence, we immediately obtain for the kth moment of $R(t)$,

$$\mathrm{E}\left[(R(t)^k\right] = \frac{\mathrm{E}\left[\int_0^\Delta (t+1-\Delta)^k \, dt\right]}{\mathrm{E}\left[\Delta\right]} = \frac{1 - \mathrm{E}\left[(1-\Delta)^{k+1}\right]}{(k+1)\cdot \mathrm{E}\left[\Delta\right]}.$$

For example, with $k = 1, 2$ the above reduces into

$$\mathrm{E}\left[R(t)\right] = 1 - \frac{1}{2}\cdot\frac{\mathrm{E}\left[\Delta^2\right]}{\mathrm{E}\left[\Delta\right]} \quad \text{and} \quad \mathrm{E}\left[R(t)^2\right] = 1 - \frac{3\cdot \mathrm{E}\left[\Delta^2\right] - \mathrm{E}\left[\Delta^3\right]}{3\cdot \mathrm{E}\left[\Delta\right]}, \tag{1}$$

and, consequently, we have,

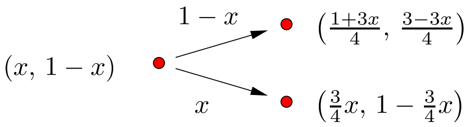
$$\mathrm{V}\left[R(t)\right] = \mathrm{E}\left[R(t)^2\right] - \mathrm{E}\left[R(t)\right]^2 = \frac{4\cdot \mathrm{E}\left[\Delta\right]\mathrm{E}\left[\Delta^3\right] - 3\cdot \mathrm{E}\left[\Delta^2\right]^2}{12\cdot \mathrm{E}\left[\Delta\right]^2}. \tag{2}$$

C. Embedded Markov chain

One can associate an embedded Markov chain to the (normalized) sending rate process described earlier by considering the time instances when the total rate $R(t)$ attains the capacity of the bottleneck link. Let vector $\mathbf{X}^{(k)}$ denote the sending rates at the k^{th} point, $k = 1, 2, \ldots$. Then, with the probability of $X_i^{(k)}$ the next state $\mathbf{X}^{(k+1)}$ of the embedded Markov chain is

$$(\, X_1^{(k)} + \Delta^*, \, \ldots, \, \nu X_i^{(k)} + \Delta^*, \, \ldots, \, X_n^{(k)} + \Delta^*),$$

where $\Delta^* = (1-\nu)X_i^{(k)}/n$. We note that the above Markov chain has a state space in $\mathbb{R}^n$ dimensional hyperplane, $\sum_{i=1}^n X_i = 1$ with $X_i \in (0,1)$ $\forall i$. With two flows and $X_1 = x$ the state of the embedded Markov chain is $(x, \, 1-x)$. The transitions to the next embedded point (with $\nu = 1/2$) are then as follows.


$1-x$
$\left(\frac{1+3x}{4}, \, \frac{3-3x}{4}\right)$
$(x, \, 1-x)$
x
$\left(\frac{3}{4}x, \, 1-\frac{3}{4}x\right)$

where, e.g., at state $(x, 1-x)$ the reduction occurs for flow 1 with probability of x and the embedded Markov chain moves to state $(3x/4, 1-3x/4)$. Let $f(x)$ denote the pdf for the state of the flow 1 at the embedded points, $\mathrm{P}\left\{x < X_1 \le x + dx\right\} = f(x)\, dx$. For this system one can write the global balance equations (see [1] and Section III-A), which yield the steady state solution for the interval $x \in (0, 1/4]$,

$$f(x) = \frac{a}{4}\cdot\sqrt{\frac{3^{n(n-3)}}{4^{n(n-1)}}} = \frac{a}{8}\left(\frac{2}{3}\right)^n x^{(n-1)/2}, \qquad \text{with } n = \frac{\log 4x}{\log 3/4}.$$

For the other values of x we have the identities,

$$f(x) = 0, \qquad \forall\, x < 0 \text{ and } x > 1, \tag{3}$$

$$f(x) = f(1-x), \tag{4}$$

$$f(x) = \left(\frac{4}{3}\right)^2 \left[(1{-}x) f\left(\frac{4x-1}{3}\right) {+} x f\left(\frac{4x}{3}\right)\right]. \tag{5}$$

The constant a follows from the normalization condition. The resulting pdf is illustrated in Fig. 1. The distribution has a mean 0.5 and variance $\sigma^2 \approx 0.0192$.

D. Flow aggregation approach

For more than two flows it was proposed in [1] to aggregate the flows. In the flow aggregation approach we have m flows and choose one flow as the "targeted flow" while the rest of the $m-1$ flows are aggregated. Both the additive increase rate and the multiplicative decrease factor of the aggregate flow are adjusted accordingly. Writing the global balance equations for the embedded MC yields a similar steady state solution, giving the pdf explicitly for a certain interval while the value elsewhere can be computed recursively using a set of recursive equations. For details we refer to [1].

III. Analytical Results

Next we give some new complementary analytical results to those given in [1] and restated in Section II. In particular, first we give some arguments about the state space of the embedded MC, which show that the states are dense in interval $[0,1]$, and that any two macro states, i.e., subintervals of $[0,1]$, communicate. These support the used approach of describing the steady state distribution of the MC by a continuous pdf. Then we also derive explicit expressions for the joint pdf of the sending rates at an arbitrary point of time.

A. Remarks On the State Space

In the analysis it has been implicitly assumed that the state space is dense, i.e., that feasible states exist inside every open interval. Our aim here is to argue that this indeed is the case. For simplicity, let us consider the 2-flow case, for which the next state in embedded MC is defined by,

$$X_{k+1} = \begin{cases} \dfrac{3}{4} X_k, & \text{with probability of } X_k, \\ \dfrac{1+3X_k}{4}, & \text{otherwise.} \end{cases}$$

In general, the multiplicative factor can be different than $1/2$. Thus, consider recursive equations of form,

$$X_{k+1} = \begin{cases} \alpha X_k, & \text{with probability of } p(X_k), \\ 1-\alpha(1-X_k), & \text{otherwise,} \end{cases}$$

where $p(x)$ denotes the probability of choosing the "targeted flow" for the rate reduction, and α depends on the multiplicative factor ν according to,

$$\alpha = \frac{1+\nu}{2}.$$

We can safely assume that in any state both transitions are possible, i.e., $\forall\, x \in (0,1)$, $\exists\, \epsilon > 0$ such that $\epsilon < p(x) < 1-\epsilon$. Let x_0 denote the initial state, $X_0 = x_0$. Then for the next state X_1 we have

$$X_1 = (1-\alpha)\, d_1 + \alpha\, x_0,$$

where d_1 is a binary random variable equal to 0 if in the first step the targeted flow was chosen to reduce its rate, and otherwise 1. Similarly, after two steps the state X_2 is given by

$$X_2 = (1-\alpha)\, d_2 + \alpha\, X_1 = (1-\alpha)\, d_2 + \alpha\,(1-\alpha)\, d_1 + \alpha^2\, x_0.$$

Consequently, it turns out that after k steps the state of the general process is given by

$$X_k = (1-\alpha) \sum_{i=0}^{k-1} \alpha^i \cdot d_i + \alpha^k x_0.$$

Note the change in numbering, i.e., here d_i denotes the event i steps ago. For $0 < \alpha < 1$ the last term clearly vanishes as $k \to \infty$, i.e., when stationarity is reached the system state is given by an infinite series,

$$X(\mathbf{d}) = (1-\alpha) \sum_{i=0}^{\infty} \alpha^i \cdot d_i, \tag{6}$$

where the d_i are binary variables having value 0 or 1. In other words, using (6) we can associate a certain state $X(\mathbf{d})$ to each infinite sequence $d_0, d_1, \ldots$ of binary numbers. Note that the random variables d_i are not independent. As $0 < \alpha < 1$ the series clearly converges always (a strictly increasing series bounded by a geometric series). With $d_i = 0,\ \forall\, i$, we obtain the minimum, $X(\mathbf{d}) = 0$, and for the maximum $d_i = 1,\ \forall\, i$, which yields $X(\mathbf{d}) = 1$. The important question is if the state space is dense in $[0,1]$, i.e., whether a feasible state exists inside every open interval $(a,b) \subset [0,1]$. The answer is yes as long as $1/2 \le \alpha < 1$, which is the case here as $\alpha = 1/2 + \nu/2$.

Lemma 1 *For any given $r \in [0,1]$ there exists a sequence of binary numbers d_i, $d_i \in \{0,1\}$, for which the series* (6) *converges to r as long as $1/2 \le \alpha < 1$.*

Proof: Case $\alpha = 1/2$ is trivial (binary number representation). Also for $r = 1$ a series with $d_i = 1$, $\forall\, i$, clearly converges. Thus, we can assume that $1/2 < \alpha < 1$ and $r < 1$. Define the partial sums,

$$S_k = \sum_{i=0}^{k-1} d_i \cdot a_i, \qquad \text{where } a_i = (1-\alpha)\alpha^i.$$

Then, for each $k = 0, 1, 2, \ldots$ define a series recursively as follows

$$d_k = \begin{cases} 1, & \text{if } S_k + a_k < r, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, $S_k < r,\ \forall\, k$. Let $k_0 = k$ denote the smallest k with $d_k = 0$, which clearly exists as otherwise $S_k > r$ for k large enough. In other words, $S_{k_0} < r < S_{k_0} + a_{k_0}$. As for all $k = 0, 1, \ldots$ the tail is heavier than a_k,

$$\sum_{i=k+1}^{\infty} a_i = \alpha^{k+1} > (1-\alpha)\alpha^k = a_k, \qquad \text{(when } 1/2 < \alpha < 1)$$

there exists $k_1 > k_0$ with $d_{k_1} = 0$. By induction, there exists a sequence of integers, k_i, $i = 0, 1, \ldots$, with $k_i \ge i$ and $d_{k_i} = 0$ for which it holds that

$$S_{k_i} < r < S_{k_i} + a_{k_i}.$$

Furthermore, as $a_{k_i} \le a_i = (1-\alpha)\alpha^i$, it follows that for any $\epsilon > 0$,

$$a_{k_i} < \epsilon, \quad \text{when} \quad i > \frac{\log \epsilon - \log(1-\alpha)}{\log \alpha},$$

and hence $S_k \to r$ as $k \to \infty$. ■

Remark 1 *For $0 < \alpha < 1/2$ there are intervals of positive length in $[0,1]$ that no series of form* (6) *converges to. For example, it is easy to see that no series converges to any number in interval $(\alpha, 1-\alpha)$.*

Remark 2 *Similarly, it can be shown that any two macro states consisting of states within (non-empty) intervals $I_1 = (x_1, x_1 + dx_1) \subset [0,1]$ and $I_2 = (x_2, x_2 + dx_2) \subset [0,1]$ communicate, i.e., there is a positive probability that the system initially in any state in I_1 ends up to a some state in I_2. This can be shown, e.g., by noticing that any macro state $I = (x, x+dx)$ communicates with macro state $(0, \epsilon/2)$ for any $\epsilon < dx$.*

B. Joint Distribution of Rates

Let us next pose the question what is the joint distribution of the sending rates $r_1(t)$ and $r_2(t)$ at an arbitrary point of time. For simplicity we consider the two flow case. Note that the embedded points correspond to the conditional (joint) distribution on the capacity limit, $r_1(t) + r_2(t) = c$. The knowledge of the joint distribution, however, is in some sense a more general and answers to the question what proportion of time source 1 and source 2 are sending at rate u and rate v, respectively, where $u + v \le c$. In particular, our aim is to determine expression for the cumulative joint distribution function,

$$F(u,v) := \mathrm{P}\{r_1(t) < u,\ r_2(t) < v\}.$$

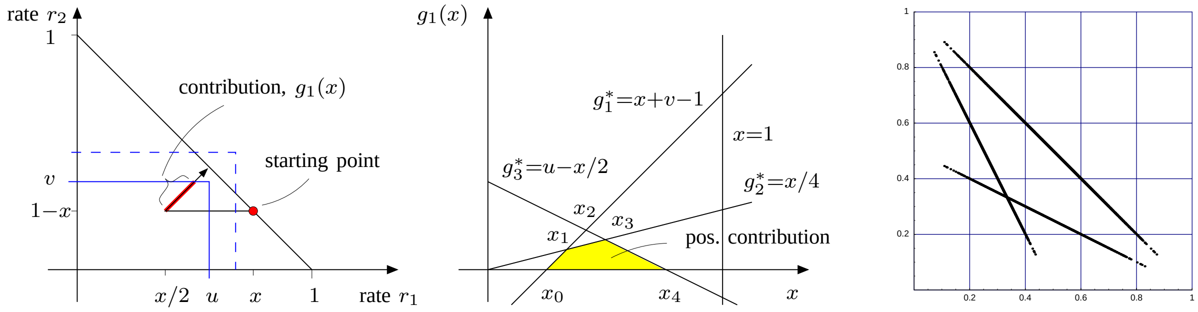


Fig. 2. (Left): one period from the capacity limit back to the capacity limit. (Middle): the time interval inside the box (u, v) when starting from point $(x/2,\ 1-x)$ is the minimum of linear functions. (Right): Visited states before and after rate reducement in a sample of 10000 packet drops. On x- and y-axes are the rates r_1 and r_2, respectively.

Conditioning on the period starting from state $(x, 1-x)$ yields two possible paths (see Fig. 2 left):

1) With probability of x, $\quad \big(x, 1-x\big) \rightarrow \big(\frac{x}{2}, 1-x\big) \rightarrow \big(\frac{3x}{4}, 1-\frac{3x}{4}\big)$.

2) Otherwise, $\quad \big(x, 1-x\big) \rightarrow \big(x, \frac{1-x}{2}\big) \rightarrow \big(\frac{1+3x}{4}, \frac{3-3x}{4}\big)$.

In above, the first state is the initial state before rate reduction at time t, the second state is the state immediately after that, and the third state is the state at the point of time when the total sending rate again attains the capacity limit, at time $t+\Delta$. The time averages are generally obtained by evaluating the integral,

$$\frac{1}{\mathrm{E}\left[\Delta\right]}\int_0^1 x\cdot f(x)\cdot g_1(x) + (1-x)\cdot f(x)\cdot g_2(x)\,dx = \frac{\mathrm{E}\left[X\cdot g_1(X)\right]+\mathrm{E}\left[X\cdot g_2(1-X)\right]}{\mathrm{E}\left[X^2\right]},$$

where $g_1(x)$ denotes the contribution when the first source is reduced at state $(x, 1-x)$, and $g_2(x)$, similarly, the contribution when the second source is reduced at state $(x, 1-x)$. In the latter form we have utilised the symmetry, $f(x) = f(1-x)$, which holds for two flow case. It is easy to convince oneself about the correctness of the above, e.g., by considering a long time interval consisting of M periods when M tends to infinity. As we are interested in the cumulative joint distribution $F(u, v)$, the functions $g_1(x)$ and $g_2(x)$ correspond to time intervals during which $r_1(t) < u$ and $r_2(t) < v$ in a given period. The situation is illustrated in Fig. 2 (left). In order to have a positive contribution to the integral the initial state $(x, 1-x)$ must be suitable, i.e.,

$$1-x < v \quad\Rightarrow\quad x > 1-v \qquad \text{and} \qquad x/2 < u \quad\Rightarrow\quad x < 2u.$$

Moreover, we have $0 < x < 1$, which yields

$$1-v < x < \min\{1,\ 2u\}.$$

Similarly, the time interval inside the box $(0, 0) - (u, v)$ when starting from point $(x/2, 1-x)$ is the time to the first linear constraint as illustrated in Fig. 2 (middle). Thus, we have,

$$\begin{aligned}
g_1^*(x) &= x + (v-1), & x_0 &= 1-v, & x_3 &= (4/3)u,\\
g_2^*(x) &= x/4, & x_1 &= (4/3)(1-v), & x_4 &= \min\{1,\ 2u\},\\
g_3^*(x) &= -x/2 + u, & x_2 &= (2/3)(u-v+1), & &
\end{aligned}$$

where $g_1^*(x)$ and $g_3^*(x)$ correspond to the time-intervals until $r_2(t) = v$ and $r_1(t) = u$, respectively, and $g_2^*(x)$ corresponds to the time-interval until $r_1(t) + r_2(t) = 1$ in case $u + v > 1$ (cf. dotted box in Fig. 2 (left)). Hence, denoting $f_i^* = x\cdot f(x)\cdot g_i^*(x)$, the first integral $I_1 = I_1(x, u, v)$ can be written as

$$I_1 = \begin{cases}
\displaystyle\int_{x_0}^{x_4} f_1^*, & \text{when } x_4 = \min\{x_1, x_2, x_4\},\\[2ex]
\displaystyle\int_{x_0}^{x_2} f_1^* + \int_{x_2}^{x_4} f_3^*, & \text{when } x_2 = \min\{x_1, x_2, x_4\},\\[2ex]
\displaystyle\int_{x_0}^{x_1} f_1^* + \int_{x_1}^{x_4} f_2^*, & \text{when } x_1 = \min\{x_1, x_2, x_3, x_4\},\\[2ex]
\displaystyle\int_{x_0}^{x_1} f_1^* + \int_{x_1}^{x_3} f_2^* + \int_{x_3}^{x_4} f_3^*, & \text{otherwise.}
\end{cases}$$

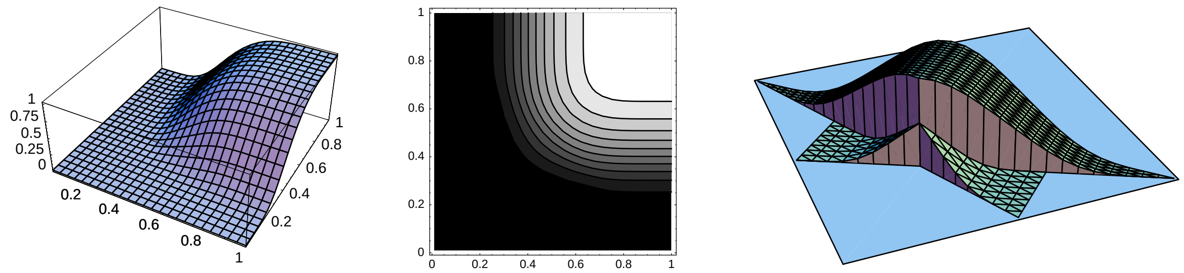


Fig. 3. Joint distribution of sending rates $r_1(t)$ and $r_2(t)$ at an arbitrary point of time. Figures on left and middle correspond to the cumulative distribution $\mathrm{P}\left\{r_1(t) < u,\, r_2(t) < v\right\}$, and the figure on right to the pdf.

Due to the symmetry, we finally have

$$F(u, v) = \frac{I_1(u, v) + I_1(v, u)}{\mathrm{E}\left[X^2\right]},$$

and the corresponding pdf,

$$f(u, v) = \frac{\partial^2}{\partial u\, \partial v} F(u, v).$$

The resulting cdf and pdf are illustrated in Fig. 3. Especially the pdf looks interesting, we have obtained the “fingerprint” of the TCP rate control mechanism in the case of two flows.

IV. Numerical Simulations

A. Idealized Model

Let us first consider the idealized model with $n > 2$ flows, and also a modified version with random delays before a source receives a NAK. In summary, we consider the ideal model

1) without delays,
2) with random delays upon negative feedback,
3) with flow aggregation and no delays,
4) with flow aggregation and random delays.

Note that 1, 2 and 4 we deal with numerical simulations, while the case 3 we can obtain from the analytical results. In particular, the idea is to evaluate how much aggregation approach and/or random delays deviate from the idealized situation. The random delays are set so that the maximum delay is 10% higher than the minimum while the mean rate $\mathrm{E}\left[R(t)\right]$ is kept the same. The important statistics we consider are:

1) decrease in total rate Δ, i.e., the time between two packet drops: a) the mean, $\mathrm{E}\left[\Delta\right]$, b) the variance, $\mathrm{V}\left[\Delta\right]$, and c) the covariance between the consecutive times between packet drops, $\mathrm{Cov}\left[\Delta_t,\, \Delta_{t+1}\right]$.
2) total sending rate at random point of time: a) the mean rate, $\mathrm{E}\left[R(t)\right]$, and b) the variance, $\mathrm{V}\left[R(t)\right]$

The numerical results are presented in Table I. The analytical results for mean rate $\mathrm{E}\left[R(t)\right]$ and its standard deviation are obtained using (1) and (2), respectively. From the numerical results we can make several observations. Firstly, the random delay does not affect $\mathrm{E}\left[\Delta\right]$, which is rather obvious. However, the random delay does increase $\mathrm{E}\left[\Delta^2\right]$ and $\sigma(\Delta)$, which again is expected. In particular, adding the random delay in the model does not have much effect on $\mathrm{E}\left[R(t)\right]$, which mean that the mean transmission rate can be approximated by

$$1 - \frac{1}{2} \cdot \frac{\mathrm{E}\left[\Delta^2\right]}{\mathrm{E}\left[\Delta\right]},$$

even in the cases when there is a random delay before a source receives a NAK. As adding random delays increases $\mathrm{E}\left[\Delta^2\right]$ it can be expected that also the variance of random variable $R(t)$ becomes higher. And this indeed is the case as can be seen from Table I.

Finally, comparing the results of the ideal model and the flow aggregation for $n > 2$ flows (for $n = 2$ flows the flow aggregation is accurate) suggests that the assumption made in the aggregation (i.e., flows inside the aggregate share the bandwidth equally) is rather optimistic. The difference in the mean values of Δ and $R(t)$ is not large, but for their respective standard deviations the difference becomes noticeable.

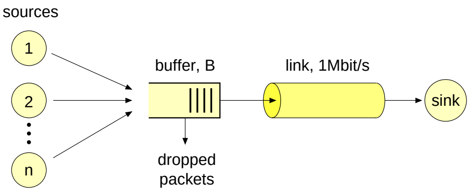
TABLE I: Simulation results compared with the analytical (aggregation without delays).

		analytical				simulated			
Model	# of flows	$\mathrm{E}[\Delta]$	$\sigma(\Delta)$	$\mathrm{E}[R(t)]$	$\sigma(R(t))$	$\mathrm{E}[\Delta]$	$\sigma(\Delta)$	$\mathrm{E}[R(t)]$	$\sigma(R(t))$
ideal without delays	2	0.269	0.0666	0.857	0.0903	0.269	0.0667	0.857	0.0903
ideal with random delays	2					0.269	0.0763	0.858	0.0940
ideal without delays	3	0.175	0.0360	0.909	0.0566	0.184	0.0539	0.900	0.0654
ideal with random delays	3					0.184	0.0657	0.901	0.0707
aggregation without delays	3					0.175	0.0363	0.909	0.0567
aggregation with random delays	3					0.175	0.0523	0.910	0.0627
ideal without delays	4	0.129	0.0219	0.933	0.0398	0.140	0.0441	0.923	0.0512
ideal with random delays	4					0.140	0.0579	0.924	0.0582
aggregation without delays	4					0.130	0.0238	0.933	0.0412
aggregation with random delays	4					0.130	0.0450	0.934	0.0496
ideal without delays	5	0.103	0.0195	0.947	0.0347	0.113	0.0370	0.938	0.0421
ideal with random delays	5					0.113	0.0518	0.940	0.0506
aggregation without delays	5					0.103	0.0172	0.947	0.0324
aggregation with random delays	5					0.103	0.0420	0.948	0.0427

B. *Comparison Against "real TCP"*

Finally, we will present the numerical results obtained by a J-Sim simulator. J-Sim is an open source, component-based compositional simulation environment developed in Java [11], which, among other things, provides a detailed implementation of the different flavors of the TCP protocol. In this work we have chosen to use TCP Reno. The simulation setup is illustrated below. The feedback channel is lightly loaded and is not thus depicted in figure. Otherwise, the parameters are as follows:

- two TCP flows
- TCP MSS of 512 B
- single bottleneck link, 1 Mb/s
- router buffer size, B, is a variable parameter
- buffer management scheme: RED with minTresh $0.25 \cdot B$, maxTresh $0.75 \cdot B$, maxProb 0.02.



1) Packet inter-loss times: From the simulations we have recorded the time instances when a packet is dropped and the aim is to compare the statistics of Δ (after normalization). The covariance and correlation coefficient between two consecutive inter-loss times are given in Table II. The model predicts negative correlation, i.e., after a short inter-loss time the next is typically longer, and vice versa. This is sensible, as a short inter-loss time means that a flow with a rather small transmission rate was chosen to reduce its rate, and consequently, the other flow has rather high transmission rate and is likely to be chosen to reduce its rate the next. However, in the simulations we have obtained both negatively and positively correlated structure of inter-loss times. The inconsistency is due to the RED algorithm, which tries to "smoothen" the loss process, and to the fact that buffer size is strictly positive (as it is in reality).

The empirical inter-loss distribution is depicted in Fig. 4 for three different buffer sizes: 3000, 8000 and 15000 bytes. In the first two cases the distribution seems to have two peaks, where the first one corresponds to the burst of losses, and the second peak to the next loss after the rate reduction. As soon as the buffer size is large enough, about 12000 bytes in our case, the first peak in practice disappears. The mean inter-loss time increases as the buffer size is increased. However, our model is "bufferless" and thus is not capable to grasp this. Instead, let us define a so-called normalized inter-loss time,

$$\Delta_* = \frac{\Delta}{\mathrm{E}[\Delta]}.$$

Then we can compare the second and third moment of Δ_* based on the simulations and the model. According to the model, the first three moments of the (non-normalized) inter-loss time are

$$(0.269,\ 0.0769,\ 0.0231)\,.$$

The respective normalized moments for the simulation results and the model are given in Table II. As can be seen from the table, the results with buffer size $B = 15000$ bytes matches best the shape of the

TABLE II: STATISTICS OF PACKET INTER-LOSS TIMES.

case	$\mathrm{E}\left[\Delta_*\right]$	$\mathrm{E}\left[\Delta_*^2\right]$	$\mathrm{E}\left[\Delta_*^3\right]$	$\mathrm{Cov}\left[\Delta_{k-1},\,\Delta_k\right]$	ρ
B=3000	1.0	1.492	3.017	−0.00148	−0.239
B=8000	1.0	1.613	2.944	−0.00462	−0.244
B=15000	1.0	1.062	1.195	0.00096	0.108
model	1.0	1.061	1.182	−0.00134	−0.303

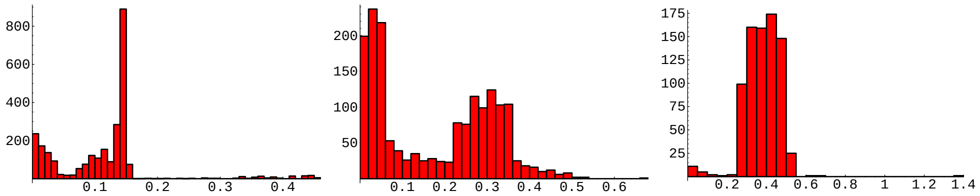


Fig. 4. Histogram of inter-loss times for buffer sizes of 3000, 8000 and 15000 bytes.

inter-loss distribution predicted by the model. For the shorter buffer sizes the burst losses, obviously, lead to mismatch. A fact that could have been seen already from the figures. But more importantly, it is clear that the inter-loss process (and the behaviour of the system in general) depends strongly on the buffer size and developing an elementary and general model for it is an extremely difficult task.

2) Times between cwnd *adjustemts:* With a small buffer size (as assumed implicitly in the model) the packet losses can be bursty, i.e., a TCP source may experience multiple losses during a single period. In order to take this into account we have also studied the times between congestion window (*cwnd*) adjustments. The simulation results for the key statistics are shown in Table III, and the corresponding empirical distribution is depicted in Fig. 5. From the figure it can be seen that when the buffer size B is large enough ($B = 15000$), the situation is rather good and the resulting distribution is smooth. Also the correlation between two consequent intervals is negative and thus matches the model better than when considering the times between packet losses (cf. Table II). For a smaller buffer size ($B = 8000$) there are two peaks in the distrubution. This corresponds to the events where during a congestion period both flows experience packet losses and consequently reduce their sending rate. We note that this kind of behaviour does not exist in the model, where it is explicitly assumed that NAK is sent to exactly one of the sources. When buffer size is very small ($B = 3000$) the system is more or less synchronized.

Finally, in Fig. 6 we have depicted the estimates of $\mathrm{E}\left[\Delta_*^2\right]$ and $\mathrm{E}\left[\Delta_*^3\right]$ as function of buffer size B according to the times between packet losses (diamond symbol), and the times between *cwnd* adjustments (star symbol). From the figure we can make some observations. Firstly, the process “settles” once the buffer size B is 12kB or larger, and secondly, both time-intervals exhibit similar behaviour while the *cwnd* based statistics are generally lower corresponding, e.g., to smaller variance.

TABLE III: STATISTICS OF TIME INTERVALS BETWEEN *cwnd* ADJUSTMENTS.

case	$\mathrm{E}\left[\Delta_*\right]$	$\mathrm{E}\left[\Delta_*^2\right]$	$\mathrm{E}\left[\Delta_*^3\right]$	$\mathrm{Cov}\left[\Delta_{k-1},\,\Delta_k\right]$	ρ
B=3000	1.0	1.220	1.997	0.00017	0.035
B=8000	1.0	1.517	2.627	−0.00645	−0.317
B=15000	1.0	1.063	1.205	−0.00117	−0.127
model	1.0	1.061	1.182	−0.00134	−0.303

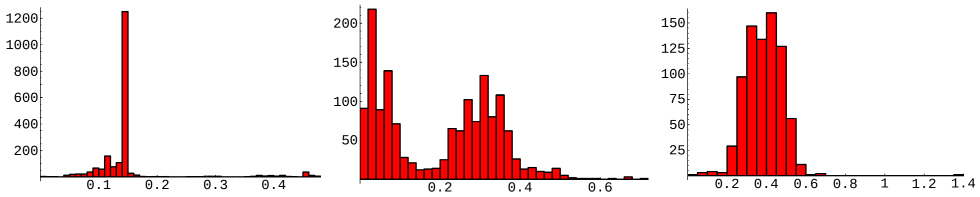


Fig. 5. Histogram of time intervals between *cwnd* adjustments for buffer sizes of 3000, 8000 and 15000 bytes.

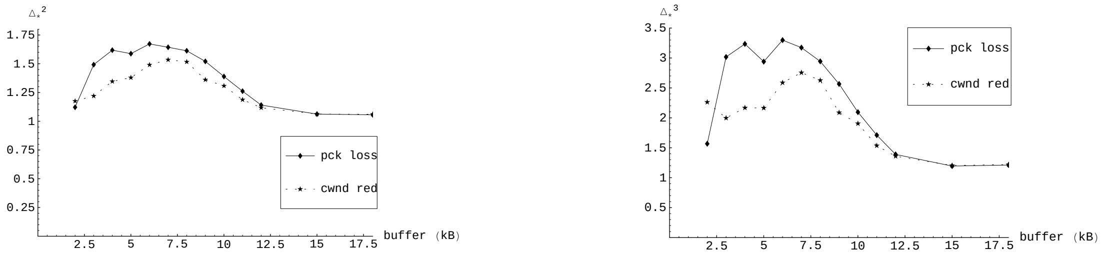


Fig. 6. Estimates of Δ_*^2 and Δ_*^3 as a function of buffer size B.

V. CONCLUSIONS

In this paper we have studied an idealized fluid flow model for TCP rate control. First we have given some new analytical results concerning the state space of the embedded Markov chain, and derived the joint distribution of the sending rates. In particular, for the state space it was shown that the states are dense in interval $[0, 1]$, and that any two macro states, i.e., subintervals of $[0, 1]$, communicate. These arguments support the approach used in [1], where the steady state distribution was described by a continuous pdf.

Then, by means of numerical simulations, we compared the elementary model to a generalized version by introducing random fluctuations in the one-way delay. It was shown that the performance quantities obtained from the model are not sensitive to small variations of RTT, which is a desirable property and justifies the original assumption of a constant delay in negative feedbacks. Especially the mean values remained almost the same, while, e.g., the variance of inter-loss times naturally increases. The model was also compared with simulation results obtained from an actual TCP implementation. This, however, turned out to be extremely difficult task due to the TCP itself. There exist several versions of TCP (e.g., Tahoe, Reno, Vegas) which all exhibit slightly different characteristics. Also, e.g., the parameters of a (single) bottleneck link may have a great impact on the behaviour of the TCP sources, while in the model the bottleneck router is assumed to be ideal. Thus, it comes as no surprise that the model cannot cover this extremely broad spectrum of different variations. A somewhat better match is obtained if the model is compared to the times between congestion window adjustments (instead of simply to the times between packet losses).

However, we can still say that the considered model of TCP serves as a good candidate for analytical work when there is a need to model TCP-like traffic in simple terms. It catches the capacity probing nature of the TCP and its sawtooth behaviour while, at the same time, can be described in simple terms (three parameters: number of sources, linear increase rate and multiplicative decrease factor). The future work includes extending the model to accomodate the possibility of several TCP sources experiencing a packet loss during a single congestion period.

REFERENCES

[1] Esa Hyytiä and Peder J. Emstad, “A model for TCP congestion control capturing the correlations in times between the congestion events,” in *Proceedings of NGI 2006*, València, Spain, Apr. 2006.

[2] Jim Kurose and Keith Ross, *Computer Networking: a top-down approach featuring the Internet*, Addison-Wesley, 2001.

[3] Dah-Ming Chiu and Raj Jain, “Analysis of the increase and decrease algorithms for congestion avoidance in computer networks,” *Computer Networks and ISDN Systems*, vol. 17, no. 1, pp. 1–14, 1989.

[4] Paul Hurley, Jean-Yves Le Boudec, and Patrick Thiran, “A note on the fairness of additive increase and multiplicative decrease,” in *Proceedings of ITC-16*, Edinburgh, UK, June 1999.

[5] Sally Floyd and Kevin Fall, “Promoting the use of end-to-end congestion control in the Internet,” *IEEE/ACM Trans. Networking*, vol. 7, no. 4, pp. 458–472, 1999.

[6] Jitendra Padhye, Victor Firoiu, Donald F. Towsley, and James F. Kurose, “Modeling TCP Reno performance: a simple model and its empirical validation,” *IEEE/ACM Trans. Networking*, vol. 8, no. 2, pp. 133–145, Apr. 2000.

[7] T. V. Lakshman and Upamanyu Madhow, “The performance of TCP/IP for networks with high bandwidth-delay products and random loss,” *IEEE/ACM Trans. Networking*, vol. 5, no. 3, pp. 336–350, 1997.

[8] Matthew Mathis, Jeffrey Semke, and Jamshid Mahdavi, “The macroscopic behavior of the TCP congestion avoidance algorithm,” *SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 3, pp. 67–82, 1997.

[9] S. Savari and E.Telatar, “The behavior of certain stochastic processes arising in window protocols,” in *Global Telecommunications Conference (GLOBECOM’99)*, 1999, vol. 1B, pp. 791–795.

[10] Eitan Altman, Konstantin Avrachenkov, and Chadi Barakat, “A stochastic model of TCP/IP with stationary random losses,” *IEEE/ACM Trans. Networking*, vol. 13, no. 2, pp. 356–369, 2005.

[11] J-Sim, `http://www.j-sim.org/`.