Chapter 11

FLUID MODEL OF AN INTERNET ROUTER UNDER THE MIMD CONTROL SCHEME

Urtzi Ayesta
LAAS-CNRS, Toulouse, France
urtzi@laas.fr

Alexei B. Piunovskiy
Department of Mathematical Sciences, University of Liverpool, Liverpool, UK
piunov@liverpool.ac.uk

Y. Zhang
Department of Mathematical Sciences, University of Liverpool, Liverpool, UK
zy1985@liv.ac.uk

Abstract We study the interaction between an MIMD (Multiplicative Increase Multiplicative Decrease) congestion control algorithm and a Drop Tail buffer. We consider the problem in the framework of deterministic hybrid models. We first show that the hybrid model of the interaction between the MIMD congestion control and bottleneck router always converges to a cyclic behavior that can be of only two different types. Second, we characterize the minimum buffer size required such that the bottleneck link is fully utilized. In particular, our analysis indicates that an MIMD algorithm requires smaller buffer sizes than AIMD in order to fully utilize the link capacity.

Keywords: Internet router; MIMD control; beamforming; fluid model.

1. Introduction

Most traffic in the Internet is governed by TCP/IP (Transmission Control Protocol and Internet Protocol) [Allman et al 1999; Jacobson 1988]. Data packets of an Internet connection travel from a source node to a destination

node via a series of routers. Some routers, particularly edge routers, experience periods of congestion when packets spend a non-negligible time waiting in the router buffers to be transmitted over the next hop. The TCP protocol tries to adjust the sending rate of a source to match the available bandwidth along the path. During the principle Congestion Avoidance phase, standard TCP uses an AIMD (Additive Increase Multiplicative Decrease) binary feedback congestion control scheme. In the absence of congestion signals from the network, TCP increases the congestion window linearly in time, and upon the reception of a congestion signal TCP reduces the congestion window by a multiplicative factor.

It has been shown that the AIMD algorithm underutilizes the resources in High-Speed networks [Floyd 2003]. Thus, in recent years researchers have proposed and studied different versions of congestion control for the Internet such as HS-TCP [Floyd 2003] and Scalable TCP [Kelly 2003]. These algorithms have in common that in the absence of congestion, the sources increase the congestion window in a much more aggressive fashion than does standard TCP. We note that Scalable TCP is a particular instance of an MIMD (Multiplicative Increase Multiplicative Decrease) congestion control algorithm, and thus, in the absence of congestion signals, the algorithm increases the congestion window exponentially.

On the other hand, most of the routers in the Internet are of Drop-Tail type. In basic Drop Tail routers, apart from the router capacity, the buffer size is the only parameter to tune. In fact, the buffer size is one of the few parameters of the TCP/IP network that can be managed by network operators. This makes the choice of the router buffer size a very important problem in the TCP/IP network design. This choice has recently received considerable attention [Appenzeller et al 2004; Avrachenkov et al 2002; Dhamdhere et al 2005; Gorinsky et al 2005; Vu-Brugier et al 2007; Avrachenkov et al 2005; Avrachenkov et al 2007]. All these works assume an AIMD congestion control protocol.

In this paper we study the interaction of MIMD congestion control algorithms with Drop-Tail buffers. We consider the problem in the framework of deterministic hybrid models. Dynamical systems that combine both discrete and continuous behavior are known as Hybrid Systems. Recently, hybrid models have been successfully applied to the modeling of communication networks [Hespanha et al 2001; Bohacek et al 2003]. We show that the hybrid model of the interaction between the MIMD congestion control and bottleneck router always converges to a cyclic behavior that can be of only two different types. Then we characterize the minimum buffer size required such that the bottleneck link is fully utilized. In particular, our analysis indicates that an MIMD algorithm requires smaller buffer sizes than AIMD in order not to waste resources.

The remainder of the paper is organized as follows. In Section 2 we develop the hybrid model. In Section 3 we present the main results. Section 4 contains sketches of the various proofs, and Section 5 contains some numerical experiments.

2. Mathematical Model

Let n long-lived TCP connections share a bottlenecked Internet router with buffer size B and transmission capacity μ. Denote by $\lambda_i(t)$ the instantaneous sending rate of connection $i = 1, ..., n$ at time $t \in [0, \infty)$. We consider a fluid model. Namely, data is represented by a fluid that flows into the buffer with rate $\lambda(t) = \sum_{i=1}^{n} \lambda_i(t)$, and it leaves the buffer with constant rate μ, if there is a backlog in the buffer. Denote by $x(t)$ the amount of data in the buffer at time $t \in [0, \infty)$. Then, the evolution of x is described by the following differential equation

$$\dot{x}(t) = \begin{cases} \lambda(t) - \mu, & \text{if } x(t) > 0, \text{or if } x(t) = 0 \text{ and } \lambda(t) > \mu, \\ 0, & \text{if } x(t) = 0 \text{ and } \lambda(t) \leq \mu. \end{cases} \tag{1}$$

If $x(t) < B$, the sending rate of connection i increases exponentially in time with rate r_i. We assume that $r_i \equiv r$. Thus, if $x(t) < B$,

$$\dot{\lambda}(t) = \sum_{i=1}^{n} \dot{\lambda}_i(t) = r\lambda(t). \tag{2}$$

When $x(t)$ reaches B, a congestion signal is sent to one or several TCP connections. Upon the reception of the congestion signal at time t, the TCP connection reduces its sending rate by a multiplicative factor $\beta_0 \in (0, 1)$, that is, $\lambda_i(t + 0) = \beta_0 \lambda_i(t - 0) = \beta_0 \lambda_i(t)$: we assume that all the functions are left-continuous. For example, in Scalable TCP, $r_i = 0.01/$'**RTT**' and $\beta_0 = 0.875$ [Kelly 2003], where '**RTT**' denotes the round-trip-time of the corresponding TCP connection. We assume that the sending rates of connections at the congestion moment are distributed uniformly and that congestion signals are sent to $\{1, ..., \tilde{n}\} \subset \{1, ..., n\}$ connections. Then we have

$$\begin{aligned} \sum_{i=1}^{n} \lambda_i(t+0) &= \sum_{i=1}^{\tilde{n}} \lambda_i(t+0) + \sum_{i=\tilde{n}+1}^{n} \lambda_i(t+0) \\ &= \beta_0 \sum_{i=1}^{\tilde{n}} \lambda_i(t-0) + \sum_{i=\tilde{n}+1}^{n} \lambda_i(t-0) \\ &= (\beta_0 - 1) \sum_{i=1}^{\tilde{n}} \lambda_i(t-0) + \sum_{i=1}^{n} \lambda_i(t-0) \end{aligned}$$

$$
\begin{aligned}
&= \sum_{i=1}^{n} \lambda_i(t-0)\left((\beta_0-1)\frac{\sum_{i=1}^{\tilde{n}}\lambda_i(t-0)}{\sum_{i=1}^{n}\lambda_i(t-0)}+1\right)\\
&\approx \sum_{i=1}^{n} \lambda_i(t-0)\left((\beta_0-1)\frac{\tilde{n}}{n}+1\right),
\end{aligned}
$$

and the total sending rate is reduced on average by the factor

$$\beta = 1-(1-\beta_0)\frac{\tilde{n}}{n}. \tag{3}$$

Since in the fluid model all variables stand for average values, we can write that $\lambda(t+0) = \beta\lambda(t-0)$, where t is a moment of congestion. We call such moments 'jump moments' (of component λ).

Let us now formulate performance criteria. On one hand, we are interested in obtaining as large throughput as possible. That is, we are interested to maximize the average sending rate

$$\bar{\lambda} = \lim_{t\to\infty}\frac{1}{t}\int_0^t \lambda(s)ds.$$

On the other hand, we are interested to make the delay of data in the buffer as small as possible. That is, we are also interested in minimizing the average amount of data in the buffer

$$\bar{x} = \lim_{t\to\infty}\frac{1}{t}\int_0^t x(s)ds.$$

Clearly, these two objectives are contradictory, and the goal of the current research is to construct the Pareto set for this multicriteria problem. We accept that all the parameters of the system are fixed, and only the buffer size B is under control. Clearly, if B is big enough the average sending rate $\bar{\lambda}$ is close to μ, but in this case $\bar{x}$ can become too big. If we reduce the value of B, the average amount of data $\bar{x}$ decreases, but $\bar{\lambda}$ also decreases because the (small) buffer can be empty during long time intervals. Remind that Pareto set is the collection of all non-dominated solutions to a multiple-objective problem. Its graphical representation is known as a trade-off curve on the plane $(\bar{\lambda}, \bar{x})$.

For the sake of simplicity we put $r = 1,\ \mu = 1$. In the general case, one should make change of variables: $\tilde{t} = rt$, $\tilde{x} = (r/\mu)x$, $\tilde{\lambda} = \lambda/\mu$. Clearly, $\bar{\lambda}$ and $\bar{x}$ must be multiplied by μ and by μ/r respectively.

In what follows, we investigate the trajectories of dynamical system (1), (2) which turn to converge to (stable) cycles. Remember that $\lambda(t+0) = \beta\lambda(t-0)$, where t is such that $x(t-0) = B$, $\lambda(t-0) > 1$, and in principle there can be several instant jumps meaning that β above should be replaced by β^k.

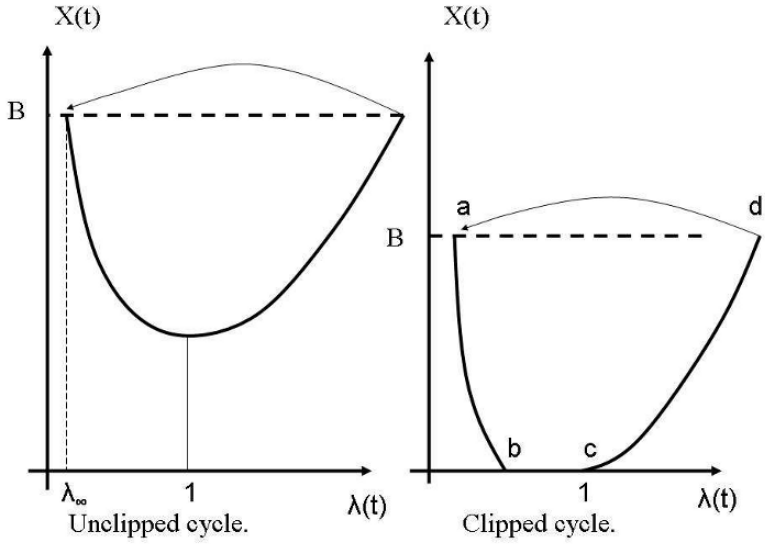

Figure 11.1. Possible cycles.

Definition 1 A trajectory of (1), (2) on a finite interval $t \in [0, T]$ *is called a cycle if* $x(0) = B$, $\lambda(0) < 1$, $x(T) = B$, $\lambda(T) \geq 1$ *and* $\lambda(0) = \beta^k \lambda(T)$, *where* $k \geq 1$ *is such that* $\beta^{k-1}\lambda(T) > 1$. *A cycle is called simple if* $k = 1$ *and* $\forall t \in (0, T)\ x(t) < B$. *(One cannot exclude in advance the situation when a cycle has several different loops resulting from the jumps of* λ *to several different points.) Cycles with component* x *being zero during a positive time interval are called clipped (Figure 11.1).*

Actually, a cycle is a one period of the vector-valued periodic function $(x(t), \lambda(t))$, but it is better to represent it graphically as a phase portrait: $x(t)$ against $\lambda(t)$. For clipped cycles, $\bar{\lambda}$ and $\bar{x}$ both increase as B increases. The trade-off of $\bar{x}$ versus $\bar{\lambda}$ is described in Theorem 3.

As it will be shown in Theorem 2, only a simple cycle can exist which is clipped/unclipped depending on the values of parameters B and β.

If $x(0) = B$, $\lambda(0) = \lambda_\infty$ is the starting point of a cycle then obviously $x(kT + 0) = B$, $\lambda(kT + 0) = \lambda_\infty$ for all integer $k \geq 0$.

It will be also shown that for fixed values of B and β, only one (simple) cycle exists which is stable in the following sense. Suppose $x(0)$ and $\lambda(0)$ are arbitrary and let λ_i be the value of $\lambda(t)$ immediately after the i-th jump. Then $\lim_{i\to\infty} \lambda_i = \lambda_\infty$. To put it different, any trajectory converges to the cycle which will be sometimes called 'limiting cycle'.

3. Main Results

Theorem 1 Let

$$B_0 \triangleq \ln \frac{\beta - 1}{\beta \ln \beta} - \frac{\beta \ln \beta}{1 - \beta} - 1. \tag{4}$$

Then unclipped (clipped) cycle exists iff $B \geq B_0$ ($B < B_0$). The duration of the cycle equals $T = -\ln \beta$, for all $B \geq 0$.

B_0 denotes the minimal buffer size such that the queue is never empty, see Figure 11.1. Thus it is natural to call B_0 'critical' buffer size. In the desynchronized case, i.e. $\tilde{n} = 1$, from equation (3) we get $\beta = 1 - \frac{0.125}{n}$ for the case of Scalable TCP ($\beta_0 = 0.875$).

Remark 1 If $r \neq 1$, $\mu \neq 1$ then the 'critical' buffer size will become $\tilde{B}_0 = (\mu/r)B_0$.

Lemma 1 In the desynchronized case with $\beta = 1 - \frac{1}{2n}$, $\lim_{n\to\infty} n^2 B_0 = \frac{1}{32}$ meaning that B_0 decreases to zero as $\frac{1}{n^2}$.

Theorem 2 Let λ_k be the value of $\lambda(t)$ immediately after the k-th jump. Then, starting from any initial value λ_0, $\lim_{k\to\infty} \lambda_k = \lambda_\infty$.

(a) If $B \geq B_0$ then $\lambda_\infty = \frac{-\beta \ln \beta}{1-\beta}$.

(b) If $B < B_0$ then $\lambda_\infty = \beta e^\theta$, where θ is the single non-negative solution to $e^\theta - 1 - \theta = B$. In this case $\lambda_2 = \lambda_3 = \ldots = \lambda_\infty$.

(c) There exist only simple limiting cycles shown in figure 11.1, i.e. instant series of more than one jump are never realized, and all the values of $\lambda(t)$ immediately after a jump coincide with λ_∞ for the trajectory starting from $x_0 = B$, $\lambda_0 = \lambda_\infty$. The limiting cycle is stable.

Remark 2 The meaning of variables can be understood from Fig.11.1: θ is the time interval corresponding to part c-d, δ is the time interval corresponding to the part a-b.

Theorem 3 If $B \geq B_0$ then

$$\bar{\lambda} = 1; \quad \bar{x} = B + \frac{\ln \beta}{2} + \frac{\beta \ln \beta}{1 - \beta} + 1.$$

If $B < B_0$ then

$$\bar{\lambda} = \frac{e^\theta(\beta - 1)}{\ln \beta}; \quad \bar{x} = -\frac{\delta e^\theta (1 - \beta) - \frac{1}{2}(\delta + \theta)^2}{\ln \beta},$$

where θ is defined in Theorem 2 and δ is the minimal positive solution to $\beta e^\theta e^\delta - \beta e^\theta - \delta + B = 0$.

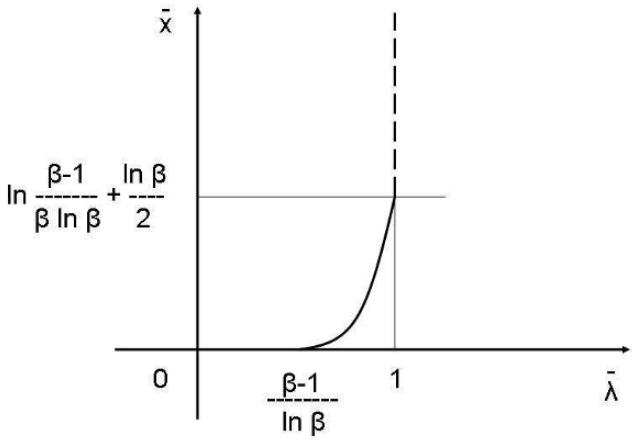

Figure 11.2. Tradeoff curve.

Corollary 1 Let $B < B_0$. Then θ, δ, $\bar{\lambda}$ and $\bar{x}$ monotonously increase with B. If B approaches zero then

$$\bar{x} \to 0; \qquad \bar{\lambda} \to \frac{\beta - 1}{\ln \beta}.$$

If B approaches B_0 then

$$\bar{x} \to \ln \frac{\beta - 1}{\beta \ln \beta} + \frac{\ln \beta}{2}; \qquad \bar{\lambda} \to 1.$$

Now it is clear that the Pareto set for objectives $(\bar{x}, \bar{\lambda})$ is realized for $0 \leq B \leq B_0$: the minimal value of $\bar{x}$ equals zero and corresponds to $B = 0$; the maximal value of $\bar{\lambda}$ equals 1 and corresponds to $B = B_0$. If $B > B_0$ then $\bar{x}$ increases with B and $\bar{\lambda} = 1$ remains the same. Thus, solutions on the vertical dashed line (Fig.11.2), when $B > B_0$, are obviously dominated.

3.1 Comments on the Results

As the term "scalable" suggests, the cycle duration T for the MIMD scheme does not depend on the capacity of the link (see Theorem 1). We recall that the cycle duration for the AIMD scheme does depend on the link capacity [Avrachenkov et al 2005].

In the case of a general link capacity $\mu \neq 1$, from equation (4) and Remark 1 we see that the 'critical' buffer size (4) depends linearly on the value of μ. In contrast, the expression for B_0 in the AIMD case has the factor μ^2. This shows that MIMD not only scales well with the capacity of the link but that also requires smaller buffer size in high speed networks. In particular, for the Scalable TCP flavor, when there is only one connection ($n = 1$ $\beta = \beta_0 = 0.875$ and $r = 0.01/$'**RTT**'), from (4) and Remark 1 we get

$$\tilde{B}_0 = 0.22\mu \cdot \text{'}\mathbf{RTT}\text{'}.$$

The most important rule-of-thumb for dimensioning buffers on the Internet suggests that the size of the queues should be equal to the bandwidth-delay product, that is, $\mu \cdot$ '**RTT**' [Villamizar and Song 1994]. Thus, the fluid model suggests that a single Scalable TCP connection requires a 78% smaller buffer than what the previous rule-of-thumb indicated. The simulation results of Section 4 suggest that in practice the value of $\tilde{B}_0$ could even be smaller.

The performance of several synchronized MIMD connections with the same '**RTT**' are equivalent to a single MIMD connection [Altman et al 2005]. If the MIMD connections are not synchronized, from Lemma 1 we know that B_0 decreases to zero as $1/n^2$. As in the AIMD congestion control scheme (see [Avrachenkov et al 2005]), the multiplexing of non-synchronized MIMD connections helps to reduce significantly the minimal required buffer space for the full utilization of the link capacity.

When $B \to 0$, from Corollary 1 we know that $\bar{\lambda} \to \mu(\beta - 1)/\ln(\beta)$. When $n = 1$ $\beta = \beta_0 = 0.875$, and we obtain that $\bar{\lambda} = 0.94\mu$, that is, with Scalable TCP, the minimal guaranteed link utilization will be equal to 94%.

4. Simulation Results

We perform network simulations with the help of NS-2, the widely used open-source network simulator [Network Simulator]. We consider the following benchmark example of a TCP/IP network with a single bottleneck link. The topology may for instance represent an access network. The capacity of the bottleneck link is denoted by μ and its propagation delay is denoted by d. The capacities of N links leading to the bottleneck link are supposed to be large enough (or the load on each access link is small enough) so that they do not hinder the traffic. Each of these N links has a propagation delay d_i. We assume that in each access link there is one persistent TCP connection.

In the NS simulations we use the following values for the network parameters: bottleneck capacity is $\mu = 100$Mbps, bottleneck link propagation delay $d = 1$ms, the access link capacity and delay are 100Mbps and 1ms, respectively. The packet size is 500bytes. As examples of AIMD and MIMD congestion control schemes, we consider New Reno [Allman et al 1999] and Scalable TCP [Kelly 2003] flavors, respectively. The number of access links is equal to

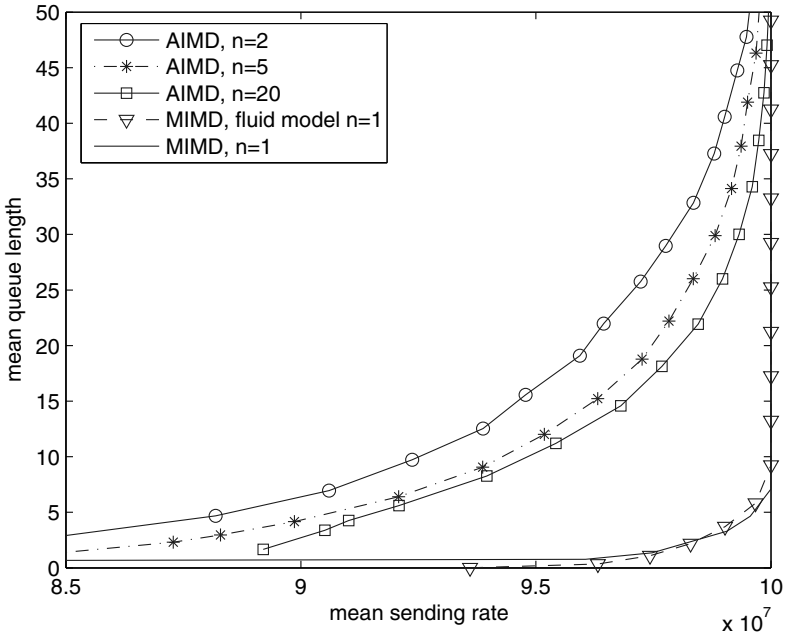

Figure 11.3. The trade-off curves for AIMD ($N = 2$, $N = 5$, $N = 20$) and MIMD $N = 1$.

the number of connections. The fact that the delays in the access links are the same implies that the TCP connections will be synchronized.

In Figure 11.3 we depict the Pareto set for the cases of AIMD with $N = 2$, $N = 5$ and $N = 20$ connections, and MIMD with just one connection. We recall that several symmetric synchronized MIMD connections are equivalent to a single MIMD connection. The qualitative shape of the curves agrees with what our model predicts (Figure 11.2). In particular, MIMD achieves the full link utilization with a much smaller buffer size than in the case of AIMD.

We also display the theoretical trade-off curve for the mathematical model with parameters $\beta = 0.875$, **'RTT'**$= 4$ ms, $r = 0.01/$**'RTT'**$= 2.5$ ms^{-1}, $\mu = 10^8$ b/s. It turns to be close to the curve coming from simulations. Anyway, when comparing the results obtained from the analytical model and from simulations we have observed differences. For example, when the buffer size is zero, the simulated average sending rate is smaller than the one obtained with the fluid model. Similarly, in the simulated scenario the minimal buffer size that guarantees full utilization of the link is larger than the one the fluid model predicts. These differences can be explained by the fact that the aggregated traffic in the simulations is not as smooth as the fluid model that we have used.

5. Conclusions

In this paper we have formulated the problem of choosing the buffer size of routers in the Internet as a multi-criteria optimization problem. A rigorous mathematical analysis of the interaction between the DropTail buffer and an MIMD congestion control algorithms have been provided. In agreement with previous works for AIMD congestion control, our model suggests that as the number of long-lived MIMD connections sharing the common link increases, the minimal buffer size required to achieve full link utilization reduces. Furthermore, our model suggests that MIMD congestion control scheme requires much less buffer space than AIMD. The Pareto set obtained with the help of our model could allow to dimension the IP router buffer size to accommodate real time traffic as well as data traffic.

Appendix: Sketch of the Proofs of the Main Statements

Proof of Theorem 1. Clearly, the first jump (or the first instant series of jumps) of the trajectory starting from $\lambda(0) = \lambda_0$, $x(0) = x_0$ results in the value $\lambda_1 \in [\beta, 1)$. Assuming the trajectory is not clipped and has no jumps on $[0, t]$, equations (1),(2) imply

$$\lambda(t) = \lambda_0 e^t, \quad x(t) = \lambda_0 e^t - t + x_0 - \lambda_0. \tag{11.A.1}$$

Equations $x_0 = x(T) = B$, $\beta\lambda(T) = \lambda_0$ result in formulae

$$T = \ln \frac{1}{\beta} \quad \text{(the duration of the cycle)} ; \qquad \lambda_0 = \frac{-\beta \ln \beta}{1-\beta}.$$

The minimal value x_{min} of trajectory (11.A.1) starting from $(\lambda_0 = \frac{-\beta \ln \beta}{1-\beta}, x_0 = B)$ corresponds to $t^* = \ln \frac{1-\beta}{-\beta \ln \beta}$. Therefore an unclipped cycle exists iff

$$x_{min} = x(t^*) = B - B_0 \geq 0.$$

In what follows, we use denotation $\lambda_\infty = \frac{-\beta \ln \beta}{1-\beta}$ since the initial value λ_0 may be arbitrary.

In case $B < B_0$, starting from $(\lambda_0 = 1, x_0 = 0)$, the trajectory reaches level $x(\theta) = B$ at moment θ satisfying equation $e^\theta - 1 - \theta = B$, and initial values $(\lambda_0 = \beta e^\theta, x_0 = B)$ generate the trajectory with $x_{min} < 0$ meaning that we have constructed the clipped cycle. Conversely, if the clipped cycle exists then the above reasoning must lead to $x_{min} < 0$ which is equivalent to $B < B_0$. Note that the duration of the cycle equals $T = -\ln \beta$ for any $B \geq 0$. ■

Proof of Lemma 1. Calculations are straightforward, based on the L'Hopital's rule. ■

Proof of Theorem 2. (a) Consider case $B \geq B_0$. For an arbitrary $\lambda_0 \in [\beta, 1)$ we define

$$\varphi(\lambda_0) = \beta \lambda_0 e^{T(\lambda_0)}, \tag{11.A.2}$$

where $T(\lambda_0)$ is the single positive solution to equation

$$F(\lambda_0, T) = \lambda_0 e^T - \lambda_0 - T = 0. \tag{11.A.3}$$

Here $T(\lambda_0)$ is the time interval up to the next jump assuming this part of trajectory is not clipped. This function $T(\lambda_0)$ is decreasing as well as $\varphi(\lambda_0)$. One can show that $T(\beta) <$

$-2\ln\beta$. Indeed, function $F(\beta,\cdot)$ has a single minimum and the value $F(\beta,-2\ln\beta)$ is already positive. Therefore $\varphi(\beta)=\beta^2e^{T(\beta)}<1$ meaning that in fact $\varphi:\ [\beta,\varphi(\beta)]\to[\beta,\varphi(\beta)]$. Another important consequence: starting from λ_1, instant series of more than one jump are never realized. (See Item (c).)

In case the trajectory is clipped, component $x(t)$ reaches the value of B earlier, the new value $\lambda_2\le\varphi(\lambda_1)$ and again $\lambda_2\in[\beta,\varphi(\beta)]$. One can show that the trajectory starting from $(\lambda_2,x_0=B)$ cannot be clipped.

Since φ is decreasing, the double iteration $\psi(\lambda_0)\stackrel{\triangle}{=}\varphi(\varphi(\lambda_0))$ is an increasing function meaning that the sequence $\lambda_2,\lambda_4,\lambda_6,\ldots$ is monotonous and hence converges to λ_∞ such that $\psi(\lambda_\infty)=\lambda_\infty$. We intend to prove that $\varphi(\lambda_\infty)=\lambda_\infty$. Suppose $\varphi(\lambda_\infty)=\lambda'_\infty\ne\lambda_\infty$ and let T_1 and T_2 be non-negative solutions to equations

$$\lambda_\infty e^{T_1}-\lambda_\infty-T_1=0,\quad \lambda'_\infty e^{T_2}-\lambda'_\infty-T_2=0. \tag{11.A.4}$$

We know that $\beta^2\lambda_\infty e^{T_1+T_2}=\lambda_\infty$, hence $\beta=e^{-(T_1+T_2)/2}$. Since

$$\lambda'_\infty=\varphi(\lambda_\infty)=\beta\lambda_\infty e^{T_1}=\beta e^{T_1}\frac{T_1}{e^{T_1}-1},$$

we have from (11.A.4): $\lambda'_\infty=\frac{T_2}{e^{T_2}-1}=\frac{\beta T_1e^{T_1}}{e^{T_1}-1}$. Finally, using the formula for β we conclude that

$$\frac{T_1e^{T_1/2}}{e^{T_1}-1}=\frac{T_2e^{T_2/2}}{e^{T_2}-1}.$$

But function $g(\tau)=\frac{\tau e^{\tau/2}}{e^{\tau}-1}$ is monotonous, hence $T_1=T_2$ and $\lambda_\infty=\lambda'_\infty$.

Equation $\varphi(\lambda_\infty)=\lambda_\infty$ results in formulae

$$T=\ln\frac{1}{\beta}\quad\text{(the duration of the cycle)}\ ;\qquad \lambda_\infty=\frac{-\beta\ln\beta}{1-\beta},$$

and no other unclipped cycle exists. Since

$$\frac{d\varphi}{d\lambda_0}=\frac{\beta e^T(T-e^T+1)}{Te^T-e^T+1},$$

where T is given by (11.A.3), we conclude that at the stable point

$$\left.\frac{d\varphi}{d\lambda_0}\right|_{\lambda_\infty}=\frac{\beta-1-\beta\ln\beta}{\beta-1-\ln\beta}>-1$$

meaning that $\left|\frac{d\varphi}{d\lambda_0}\right|<1$ in the neighborhood of λ_∞, so that the limiting cycle is stable.

(b) If $B<B_0$ then, maximum after one jump (starting from λ_1), the trajectory is clipped, because otherwise we would have faced a sequence $\lambda_1,\lambda_2,\ldots$ resulting in unclipped trajectories and, according to (a), there would have existed an unclipped limiting cycle. Therefore, $\lambda_2=\lambda_3=\ldots=\lambda_\infty$, and the value $\lambda_\infty=\beta e^\theta$ follows from equation $x(\theta)=e^\theta-1-\theta=B$ describing the trajectory starting from ($\lambda_0=1$, $x_0=0$).

(c) The last statement follows from the previous reasoning. ■

Proof of Theorem 3. According to Theorems 1,2, the trajectories converge to a single cycle with initial values ($\lambda_0=\lambda_\infty$, $x_0=B$). Hence

$$\bar\lambda=\frac{1}{T}\int_0^T\lambda(t)dt,\qquad \bar x=\frac{1}{T}\int_0^T x(t)dt.$$

In case $B\ge B_0$, the further calculations are straightforward having in mind formulae (11.A.1) and $T=-\ln\beta$.

Case $B < B_0$. Here one has to calculate the integrals along branches a-b, b-c and c-d (Fig.11.1). Equation for δ comes from condition $x(\delta) = 0$ for trajectory starting from ($\lambda_0 = \lambda_\infty = \beta e^\theta$, $x_0 = B$). One can show that it has exactly two positive solutions. The further calculations are routine. ■

Proof of Corollary 1. The monotonicity of θ follows directly from its equation. δ is the first moment when coordinate $x(t)$ becomes zero. As B and $\lambda_\infty = \beta e^\theta$ increase, the value $x(t)$ at an arbitrary t corresponding to the part a-b (Fig.11.1) increases with B meaning that δ increases. The reasoning presented implies that the value of the integral $\int_0^T x(t)dt$ increases, and hence $\bar{x}$ increases. (Remember $T = -\ln\beta$ is constant.) The value of $\bar{\lambda}$ increases because θ increases. If $B \to 0$ then $\theta \to 0$ and $\delta \to 0$. If $B \to B_0$ then

$$\theta \to \ln\left(\frac{-\ln\beta}{1-\beta}\right); \quad \delta \to T - \theta = -\ln\beta - \ln\left(\frac{-\ln\beta}{1-\beta}\right) = -\ln\left(\frac{-\beta\ln\beta}{1-\beta}\right).$$

Calculations of the limits are straightforward. ■

References

M. Allman, V. Paxson and W. Stevens, TCP congestion control, *RFC 2581*, April 1999, available at `http://www.ietf.org/rfc/[Allman et al 1999].txt`.

E. Altman, K. Avrachenkov, and B. Prabhu, "Fairness in MIMD Congestion Control Algorithms", in Proceedings of IEEE Infocom, March, 2005.

G. Appenzeller, I. Keslassy and N. McKeown, "Sizing Router Buffers", ACM SIGCOMM '04, Portland, Oregon, September 2004. Also in Computer Communication Review, Vol. 34, No. 4, pp. 281-292, October 2004.

K. Avrachenkov, U. Ayesta, E. Altman, P. Nain, and C. Barakat, "The effect of router buffer size on the TCP performance", in the Proceedings of LONIIS workshop, St. Petersburg, Jan. 29 - Feb. 1, 2002.

K. Avrachenkov, U. Ayesta, A. Piunovskiy, "Optimal choice of the buffer size in the Internet Routers", in Proceedings of IEEE CDC-ECC 2005.

K. Avrachenkov, U. Ayesta and A. Piunovskiy, "Convergence and Optimal Buffer Sizing for Window Based AIMD Congestion Control", INRIA Research Report RR-6142, Available at: http://hal.inria.fr/inria-00136205.

S. Bohacek, J.P. Hespanha, J. Lee, and K. Obraczka, "A hybrid systems modeling framework for fast and accurate simulation of data communication networks", in Proceedings of ACM SIGMETRICS 2003, pp.58-69.

A. Dhamdhere, H. Jiang, and C. Dovrolis, "Buffer Sizing for Congested Internet Links", in the Proceedings of IEEE Infocom, Miami FL, March 2005.

S. Floyd, "HighSpeed TCP for Large Congestion Windows", *RFC 3649*, December 2003, available at `http://www.ietf.org/rfc/rfc3649.txt`.

S. Gorinsky, A. Kantawala, and J. Turner, "Link buffer sizing: a new look at the old problem", in Proceedings of IEEE Symposium on Computers and Communications (ISCC 2005), June 2005.

J.P. Hespanha, S. Bohacek, K. Obraczka, and J. Lee, "Hybrid modeling of TCP congestion control", In *Hybrid Systems: Computation and Control*, LNCS v.2034, pp.291-304, 2001.

V. Jacobson, Congestion avoidance and control, *ACM SIGCOMM'88*, August 1988.

T. Kelly, "Scalable TCP: Improving performance in highspeed wide area networks", *Computer Comm. Review*, v.33, no.2, pp.83-91, 2003.

"Network Simulator, Ver.2, (NS-2) Release 2.18a", Available at: http://www.isi.edu/nsnam/ns/index.html.

C. Villamizar and C. Song, "High Performance TCP in the ANSNET", *ACM SIGCOMM Computer Communication Review*, v.24, no.5, pp.45–60, November 1994.

G. Vu-Brugier, R.S. Stanojevic, J. Leith, and R.N. Shorten, "A critique of recently proposed buffer-sizing strategies", *ACM SIGCOMM Computer Communication Review*, v.37(1), pp.43-48, 2007.