Heavy Traffic Analysis of AIMD Models

Eitan Altman, J. Harold Kushner

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Heavy Traffic Analysis of AIMD Models

Eitan Altman — Harold J. Kushner

N° 5088

January 2004

THÈME 1

Rapport
de recherche

ISSN 0249-6399 ISRN INRIA/RR--5088--FR+ENG

Heavy Traffic Analysis of AIMD Models

Eitan Altman* , Harold J. Kushner[†]

Thème 1 — Réseaux et systèmes
Projets Maestro

Rapport de recherche n° 5088 — January 2004 — 21 pages

Abstract: The goal of this paper is to study heavy traffic asymptotics of many Additive Increase Multiplicative Decrease (AIMD) connections sharing a common router in the presence of other uncontrolled traffic, called "mice". The system is scaled by speed and average number of sources. With appropriate scalings of the packet rate and buffer content, an approximating delayed diffusion model is derived. By heavy traffic we mean that there is relatively little spare capacity in the operating regime. In contrast to previous scaled models, the randomness due to the mice or number of connections is not averaged, and plays its natural and dominant role. The asymptotic heavy traffic model allows us to analyze buffer management policies of early discarding as a function of the queue size and/or of the total input rate and to choose its parameters by posing an appropriate limiting optimal control problem. This model is intuitively reasonable, captures the essential features of the physical problem, and can guide us to good operating policies. After studying the asymptotics of a large number of persistent AIMD connections we also handle the asymptotics of finite AIMD connections whose number varies as connections arrive and leave. The data illustrate the advantages of the approach.

Key-words: Stochastic processes/Queueing theory, control theory, AIMD, TCP.

[*] The work of this author was supported by the European Network of Excellence EURO NGI

[†] The work of this author was supported by contracts DAAD-19-02-1-0425 from the Army Research Office and National Science Foundation Grant ECS 0097447. The author is with the Division of Applied Mathematics, Brown University, Providence RI 02912, USA

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Analyse de modèles AIMD en charge lourde

Résumé : L'objectif de cet article est d'étudier des régimes asymptotiques de charge lourde d'un grand nombre de connexions à débit contrôlé de type croissance additive et décroissance multiplicative (AIMD) partageant un routeur commun, en présence d'autres connexions non-contrôlées appelées "souris". Avec un passage à l'échelle appropriée des débits des connexions et de la taille du tampon, nous obtenons un modèle approché de diffusion à retard. Par "charge lourde" nous entendons qu'il reste peu de capacité disponible dans ce régime. Contrairement aux approches précédentes de modélisation du passage à l'échelle, l'aléa du aux souris et au nombre de connexions ne disparaît pas dans le régime de charge lourde, mais joue un rôle prépondérant. Le régime de charge lourde nous permet d'analyser des mécanismes de gestion des tampons et de rejet préventif des paquets en fonction de la taille du tampon et/ou du débit total de transmission, et nous permet de choisir les paramètres de ces mécanismes à travers une formulation de problèmes asymptotiques de contrôle optimal. Nous étudions des connexions AIMD permanentes mais aussi des connexions AIMD finies, dont le nombre varie dû aux arrivées et des départs de connexions. Des calculs numériques illustrent l'avantage de notre approche.

Mots-clés : Processus stochastiques, files d'attentes, théorie de contrôle, AIMD, TCP

1 Introduction

Background and motivation One of the most active research areas in networking in recent years has been the modeling and analysis of AIMD traffic see e.g. [1, 2, 3, 4, 6, 9, 14, 15, 16, 17, 19] and references therein. When considering a single connection and modeling all other connections through an idealized loss process that they create, simple mathematical formulas for the connection's throughput can be obtained, see e.g. [1, 6, 17]. However, it may be important in practice to understand the interaction of competing random numbers of connections and the associated system randomness which determines both the throughput as well as the losses suffered by each connection. One approach is through a fixed point argument; see e.g. [4]. If losses over the nodes (or links) traversed by the connections are sufficiently small and can be assumed to be additive, an alternative framework can be used where the throughputs of TCP are obtained as the solution of a convex optimization problem and where the loss probabilities are obtained as the Lagrange multipliers [15].

Although the methodologies in [4, 15] are quite useful due to their simplicity, no dynamical systems description is provided, hence the actual "processes" do not appear, and it is very difficult to add dynamical (say, queue and packet rate dependent) controls to the formulation. The way that packet losses affect individual sources and the consequent effects on the full system are not modeled explicitly, and it is difficult to analyze the oscillations or instabilities that might be caused by delays. They do not provide a sample-path or transient analysis nor allow us to compute the probability distribution of interacting connections. Models including some of these features appear in [2, 5] under simplified assumptions on the protocol's behavior (e.g. an assumption in [2] that loss probabilities do not depend on rates, or an assumption in [5] that all connections simultaneously lose a packet when the buffer is full). In order to analyze more complex systems which include buffer management, early discarding, and the impact of the delay in the feedback loop, an alternative line of research has emerged based on fluid models using delay differential equations methodology, see e.g. [8, 16].

Once such models are formulated, a key question is what is the relation between the fluid model and the original discrete random system. In [19], the authors establish conditions under which common fluid models based on delayed differential equations are obtained as limits of discrete systems with random variations of the non controlled connections, as the number of connections and the speed of the system grows. More detail on the relations between [19] with our work appears in the Section VII.

Our goal is to identify and analyze heavy traffic approximating models for multiplexing between AIMD and non controlled traffic, where the losses are a consequence of the underlying physical processes, and to show how alternative scalings lead to this model as their asymptotic limit, as well as to determine good controls for buffer management. Although the model is not deterministic, it is much simpler to handle than the original discrete stochastic system, and (as seen through numerical examples) it allows us to optimize the design of the control for buffer management, and analyze its properties.

The basic ideas. As with many models for TCP, we will use a "fluid" model for describing the rate of transmission; i.e., rather than work explicitly with the widow size; we work with the number of packets that are allowed to be sent per unit time, and do not explicitly control window sizes. We consider a model for AIMD traffic in the operating region where the system is near capacity. The analysis will be 'asymptotic," as the system grows in speed. In particular, the bandwidth (speed of the router) as well as the mean number of users will be roughly proportional to a parameter n, which is to go to infinity. The analysis will be of the so-called heavy traffic (HT) type [12], which has been of considerable help in studying complex queueing systems that would often be intractable otherwise. Several formulations of the demand process are given. In all cases, there are a certain number of controlled users of the order of n, each having a lot of data to transmit. These share the channel with a large and randomly varying number of users with smaller amounts of data, commonly referred to as "mice." While each of the mice (resp., each of the controlled users) has identical statistical properties, this is only for convenience in the numerical analysis: Any number of classes can be handled.

The packets created by the various users enter the system in some random order, then are transmitted to a buffer via various links, from which they are further processed. If the buffer capacity is exceeded, then a packet is said to be "lost." Unless noted otherwise, the round trip delay α is the same for all AIMD users. The timing of the various rates are as seen at the buffer (not at the sources). They depend on the feedback sent from the buffer α units of time ago, which reached the source t_1 units of time ago, was acted on and affected the rates at the input to the buffer t_2 units of time later, where $t_1 + t_2 = \alpha$.

We wish to identify a region of operation which is "near capacity" for large n, and a scaling under which the stochastic effects are apparent. One approach to asymptotic analysis is via a fluid model (e.g., [19]). These tend to average or eliminate the effects of stochastic variations in the number of users, mice, data rates, etc. But we are more concerned with demonstrating the actual random processes of losses and buffer content in terms of the random processes of arrivals, data levels, etc.

We are guided by the scaling used for heavy traffic models, as in [12]. There are two related aspects to being "near capacity." One is the difference between the mean packet creation rate and the speed of the system, and the other concerns the buffer size. Suppose that the mean rate of arrival of packets to the buffer is vn. In order for the system to be in the heavy traffic regime, the speed of exiting the buffer would have to be slightly greater than the mean arrival rate, but not so much faster that the buffer is virtually empty almost all of the time. If the arrival process is the superposition of many independent users, then (loosely speaking) the standard deviation of the "randomness" would be $O(\sqrt{n})$. This suggests that if the system is near capacity at that time, then both the buffer size and the extra capacity would be $O(\sqrt{n})$. Indeed, if either the buffer or speed are of a larger order, then the buffer level (scaled by $1/\sqrt{n}$) would go to zero as $n \to \infty$, and there would be no observable packet loss. Indeed, these are the usual orders in heavy traffic analysis. The amplitude scaling will be $1/\sqrt{n}$.

The heavy traffic regime is one important region of operation, one where small changes in the rates will have major consequences for buffer overflow (i.e, lost packets) and queueing delay. One can view the system as starting much below capacity, with a lower packet rate, and with the rates increasing until capacity is almost reached, at which point the control mechanisms are activated. Our analysis is confined to the time that the system is in this heavy traffic regime.

The controls. Typically, there are two types of rate control for each user. The first (the AI in AIMD) is the usual simple slow and steady linear increase in the allowed rate of packet creation when there are no buffer overflows.[1] As noted above, in the heavy traffic regime, the number of controlled users is proportional to n on the average, and the excess capacity is $O(\sqrt{n})$. This suggests that the cumulative effect of the first type of control should be a rate increase of $O(\sqrt{n})$ over all controlled users, which implies a rate increase of $O(1/\sqrt{n})$ per user. Otherwise, the system would experience very serious packet losses in short order. Thus we suppose that there is a constant u_1 such that the rate per source increases by $u_1/\sqrt{n}$. This is the correct order in the heavy traffic regime.[2] [The constant could be replaced by a function of the current buffer state and rate, if desired, but we stick to the more traditional form.] It will turn out that the cumulative effects of this control and of the buffer overflow controls are of the same order. The second type of control (the MD part) is the usual multiplicative decrease when there is a lost packet.

To improve the performance, we also use another type of control, called a *preemptive control*, by which packets are selected at random to be "marked" as they enter the buffer. This chance of being selected depends on the buffer-state and/or on the input rate, and is a control function to be chosen. (Early discarding has become very popular since it was proposed and deployed in the well known RED buffer management [7]). The selection probability will increase when the system nears a dangerous operating point. There are two choices of how to handle the marked packets. Either they are deleted so that no acknowledgment is sent, or they are not deleted and an altered acknowledgments are sent back [18] (in TCP, this avoids the need for retransmission). In either case, the source rate is decreased as though the packet were lost. This control, which anticipates the possibility of lost packets in the near future, can actually reduce the queueing as well as the rate of overflow considerably, which is desirable if we wish to protect other real-time connections (which can be considered to be part of the "mice"); numerical data will illustrate this point. In either case, the use of the preemptive control "spreads out" the "rejections," thereby helping to avoid oscillations or instability due to the effects of bursts of lost packets caused by the delays. Here, we work with the second option, and do not delete the selected packets, although the results are similar in both cases. Although the second option is not harder to implement than the first option, its advantages are enormous.

[1] This might approximate a fixed increase in the allowed number of packets that can be sent in each round trip interval, if the system allows this. We would then assume that the queueing time is small with respect to α so that the linear increase in each round trip interval can translate into a linear increase of the rate as a function of time.

[2] One could change the model, using fewer sources, each with a higher rate, and allow an accordingly faster increase in the AI control. The analysis would be similar.

Outline of the paper. A general model for the mice is discussed in the next section. Two properties are paramount. One concerns the asymptotic (scaled) total number of packets that have been transmitted by them over any time interval. The other concerns the current rate of creation of packets. The assumptions are intuitively reasonable. To emphasize this, we discuss one particular example in detail, starting from more "physical" assumptions. It is supposed (as is commonly done) that the mice enter with a fixed packet rate (possibly random among the individuals), but that they are in the system for a relatively short time, are not controlled and do not retransmit lost packets. This latter assumption can be dispensed with, but we prefer to keep the development relatively simple, so that the central issues are not obscured.

In Section 3, we consider the case where there are just n controlled users, analogously to the setup in [19]. Each of them has a very large (infinite, here) amount of data to be sent, and is subject to rate control. However, the randomness of the mice process has significant effects on the total throughput, due to the lost packets (buffer overflows), and the consequent rate control. Section 4 considers various extensions of the basic model of Section 2, including the case where there is no buffer and where the rate for the controlled users changes randomly, perhaps due to reinitializations; this can be useful to model a sequence of TCP connections that are opened consecutively by the application layer as is the case in the HTTP/1.1 version.

Section 5 deals with the case where the controlled users appear at random, each with a random amount of data to be sent, and vanish when their data has been transmitted. This introduces additional randomness, which (in the asymptotic limit) shows up via the addition of new Wiener processes in the dynamics for the rate process. There are analogs of the extensions noted in Section 4. Some typical data that show the advantages of the approach are in in Section 6.

2 The Model for the Mice

Recall that we use the name "mice" to describe any set of sources whose transmission rates are uncontrolled. Generally, they involve small numbers of packets. But various cases where the number of packets is large are covered by the assumptions. We suppose that the total rate at which mice packets are being put into the buffer at time t is $a_m n + \sqrt{n}\xi^n(t)$, where $a_m > 0$ and $\xi^n(\cdot)$ is a random process such that $\int_0^t \xi^n(s)ds$ converges weakly to a Wiener process $w_m(\cdot)$, with variance σ_m^2. More specifically (where $\Rightarrow$ denotes weak convergence),

$$\begin{aligned}&\frac{(\text{total number of mice packets by) } t - na_m t}{\sqrt{n}}\\ &= \int_0^t \xi^n(s)ds \Rightarrow w_m(t),\end{aligned} \tag{2.1a}$$

$$\begin{aligned}
&\frac{\text{mice rate}(\cdot) - na_m}{n} = \frac{\xi^n(\cdot)}{\sqrt{n}} \Rightarrow \text{“zero” process},\\
&\sup_n E \sup_{s\le t}\left|\int_0^s \xi^n(\tau)d\tau\right| < \infty, \quad \text{each } t>0.
\end{aligned} \tag{2.1b}$$

(2.1a) says that the total mice packet rate is the sum of a “fluid” component and a part that is essentially independent over short and disjoint intervals. It is motivated by the central limit theorem. Owing to the complicated way that packets from different users are scrambled in transmission, it might be hard to say more, or to specify the “mice” model more explicitly. However, one specific example, is given below. The sizes of the individual mice can grow with n, but slower than $O(n)$. All that we require is that (2.1) hold. If desired, the mice effects can have a more general form, where the Wiener process $w_m(\cdot)$ is replaced by a centered *Lèvy* process In any case, the analysis will generally have to be numerical, and the Lèvy form can be used there too.

Example of a “mice” model. Consider the following example, which was one of the motivations of the general conditions above. The example is meant to be illustrative, and does not exhaust the possibilities. Suppose that the mice arrive as a Poisson process with rate $\lambda_m n$, with each arrival having an exponentially distributed (and independent among arrivals) amount of packets, with mean v_m/μ_m. The packets are sent at a rate v_m. Then the number of active mice at any time is given by $N_m^n(t)$, where

$$dN_m^n(t) = n\lambda_m dt - \mu_m N_m^n(t)dt + dM_m^n(t),$$

where the quadratic variation process of the martingale $M_m^n(\cdot)$ is $\int_0^t [n\lambda_m + N_m^n(s)\mu_m]ds$. Let us work with the stationary processes. It follows from this that the process $N_m^n(\cdot)/n$ converges weakly to the process with constant values λ_m/μ_m. The rate at which mice packets arrive is $N_m^n(t)v_m$. Write $N_m^n(t) = n\lambda_m/\mu_m + \sqrt{n}\eta_m^n(t)$. Then

$$d\eta_m^n(t) = -\mu_m\eta_m^n(t)dt + dM_m^n(t)/\sqrt{n}.$$

The quadratic variation of the martingale $M_m^n(\cdot)/\sqrt{n}$ is the integral of $\lambda_m + \mu_m N_m^n(s)/n$, which converges weakly to the constant process with values $2\lambda_m$, and $M_m^n(\cdot)/\sqrt{n}$ converges weakly to a Wiener process $\tilde{w}_m(\cdot)$ with variance $2\lambda_m$. The process $\eta_m^n(\cdot)$ converges weakly to $\eta_m(\cdot)$, where $d\eta_m(t) = -\mu_m\eta_m(t)dt + d\tilde{w}_m(t)$. The arrival rate process from the mice satisfies (which defines $\xi(\cdot)$)

$$\frac{v_m N_m^n(\cdot) - nv_m\lambda_m/\mu_m}{\sqrt{n}} \Rightarrow v_m\eta_m(\cdot) = \xi(\cdot).$$

Note that (2.1b) holds.

The variance of (mice packet rate at $t)/\sqrt{n}$ is, asymptotically, $v_m^2\lambda_m/\mu_m$. The (packet rate) correlation function is this times $e^{-\mu_m t}$. For large enough μ_m and v_m, this is an “approximation to white noise.” To show that (2.1a) holds “approximately,” write (neglecting the initial condition),

$$\eta_m(t) = \int_0^t e^{-\mu_m(t-s)}d\tilde{w}_m(s),$$

$$\begin{aligned}\int_0^t \xi(s)ds &= v_m \int_0^t \int_0^s e^{-\mu_m(s-\tau)} d\tilde{w}_m(\tau) \\ &= \frac{v_m}{\mu_m}\tilde{w}_m(t) - \frac{v_m}{\mu_m}\int_0^t e^{-\mu_m(t-s)} d\tilde{w}_m(s).\end{aligned}$$

The dominant part is the Wiener process. Thus, in (2.1a), $a_m = v_m \lambda_m / \mu_m$ and the variance of the Wiener process is $\sigma_m^2 = 2\lambda_m [v_m/\mu_m]^2$. The stationary variance of the error process (the last term on the right) is $(v_m^2/\mu_m^2)\lambda_m/\mu_m$. Suppose that μ_m is large, with σ_m^2 kept "moderate." Then the error process is close to the "zero" process, in that it converges weakly to it as $\mu_m \to \infty$.

We could also suppose, alternatively, that the individual mice send their packets all at once, but they are interleaved randomly with those from other sources along the way, then we come even closer to (2.1).

3 n Controlled Users, Each With Infinite Backlog

In this section, there are a fixed number, namely n, of controlled users, with each having a very large (infinite here, for modeling simplicity) amount of data to be sent. Let $r_i(t)$ denote the rate for controlled source i at time t, and suppose that there are positive a_i such that $a_0 \le r_i(0) \le a_1$, so that no single source dominates. Thus $\int_0^t r_i(s)ds$ is the total number of packets generated by controlled source i by time t. Since n is large, it is unimportant that this integral will not always have integer values. Define $\bar{r}^n(t) = \sum_{i=1}^n r_i(t)/n$, and $v_1 = \bar{r}^n(0)$, $v_2 = \sum_i [r_i^n(0)]^2/n$, and $\rho^n(t) = \left[\sum_i r_i(t) - nv_1\right]/\sqrt{n}$. We suppose that v_1 is the desired average rate of packet transmission per user and is approximately the level at which we enter the heavy traffic regime. The analysis commences at the point at which this regime is entered. With this in mind, the service rate (channel speed in packets per second) is $C^n = nv_1 + a_m n + b\sqrt{n}, b > 0$, which covers the mean requirements (for both persistent connections as well as the mice process) and gives an excess (over the mean requirements) $b\sqrt{n}$.

When the buffer overflows and a packet is lost, that packet is assumed to come at random from the various users, in proportion to their individual current rates of packet creation: The various users (mice and controlled) would send their packets in some order, and the order would be more or less scrambled in the course of transmission, so that buffer overflows can be assigned at random to the various users.

As noted in the introduction, the standard multiplicative decrease control is activated by lost packets. I.e., there is some constant $\kappa \in (0,1)$ such that, if the dropped packet was from connection i, then the rate $r_i(t-)$ at $t-$ is changed to $r_i(t) = (1-\kappa)r_i(t-)$.

The "preemptive" control. It is often claimed that the system would have better properties if the sources were also signaled to reduce their rates as the buffer level or total input rate increases, but before actual buffer overflow. The type of control, called the *preemptive control,* attempts to do just this (see e.g. [7]). It selects packets on arrival, either at random or in some deterministic way with the appropriate averages, and in a buffer state and total rate dependent way. There are two possibilities in the treatment of the selected packets. They could be deleted, in which case no acknowledgment of receipt would be sent, and the source would decrease its rate accordingly. The idea is that a rejection of relatively few packets, as the system nears congestion, would reduce the buffer overflows even more. The selected packets could even be "marked" low priority ones. Of course, this comes at the cost of (buffer-state-dependent) rate reduction.

An alternative is to modify the protocol, so that the selected packets are not deleted, but a modified acknowledgment is sent, which is used to reduce the flow, similarly to what would happen if the packet were actually lost [18]. The treatment of both is similar, the main difference concerns the retransmission of the selected packets, where appropriate. We will use the latter approach, since it seems to be more promising in the sense of improving system performance with minimal extra complications. We suppose that there is a $\kappa_1 \in (0,1)$ so that the rate for such a selected source i is reduced from $r_i(t-)$ to $r_i(t) = (1-\kappa_1)r_i(t-)$. This preemptive control is to be chosen by the system designer and, when suitably selected, it can have a major beneficial effect on the overall operation.

Buffer input-output equations. We have

$$\begin{aligned}\rho^n(t) = \rho^n(0) + u_1 t - &[\text{overflow control effects}] \\ -&[\text{preemptive control effects}],\end{aligned} \tag{3.1}$$

The term $x(t)$, with or without superscripts) denotes $1/\sqrt{n}$ times the number of packets in the buffer at time t. Then

$$\begin{aligned}x^n(t) &= x^n(0) \\ &+ [\text{total input - total output - overflow by } t]\,/\sqrt{n}.\end{aligned}$$

If the buffer is not empty, then the output rate is C^n. If this output rate is always used, then we must correct for "fictitious" outputs when the buffer is empty. This is done by adding an "underflow" correction term to assure that the buffer never goes negative, as is usual in heavy traffic analysis [12]. The terms $L(\cdot)$ (resp, $U(\cdot)$) (with or without superscripts) are $1/\sqrt{n}$ times the buffer underflow (resp., overflow) or their limits. This leads to

$$x^n(t) = x^n(0) + \int_0^t [\rho^n(s) - b + \xi^n(s)]\,dt - U^n(t) + L^n(t). \tag{3.2}$$

By the proof of [12, Theorem 3.4.1, 3.5.1], there is a constant C such that, for each T,

$$U^n(T) \le C \sup_{t\le T}\left[x^n(0) + \int_0^t [\rho^n(s) + \xi^n(s)]\,ds\right]. \tag{3.3}$$

Thus, for each t, $\sup_n EU^n(t) < \infty$, and so the number of buffer overflows on any bounded interval is bounded by $O(\sqrt{n})$.

Approximating the buffer overflow or lost packet control. Suppose that there is a single overflow at time $t-\alpha$. I.e., $\sqrt{n}dU^n(t-\alpha) = 1$. Let $I_i(t-\alpha)$ denote the indicator function of the event that the overflow is associated with controlled source i. Then $r_i(t) = r_i(t-)(1-\kappa I_i(t-\alpha))$ and

$$\begin{aligned}\frac{1}{\sqrt{n}}\sum_{i=1}^{n}&[r_i(t)-r_i(t-)]\\ &= -\kappa\sum_{i=1}^{n} r_i(t-)I_i(t-\alpha)dU^n(t-\alpha).\end{aligned} \tag{3.4}$$

The user with the lost packet is selected at random, with the probability that controlled user i is selected being (its rate divided by the total rate, all at $t-\alpha$)

$$f_i^n(t-\alpha) = \frac{r_i(t-\alpha)}{\sum_j r_j(t-\alpha) + na_m + \sqrt{n}\xi^n(t-\alpha)}. \tag{3.5}$$

Use (3.5) to center (3.4) about the conditional mean (given that $dU^n(t-\alpha) > 0$) and rewrite (3.4) as

$$\begin{aligned}-\kappa\sum_{i=1}^{n} r_i(t-)&\frac{r_i(t-\alpha)dU^n(t-\alpha)}{\sum_j r_j(t-\alpha) + na_m + \sqrt{n}\xi^n(t-\alpha)}\\ &+dM_1^n(t),\end{aligned} \tag{3.6}$$

where $M_1^n(\cdot)$ is the martingale

$$\int_0^t \kappa\sum_{i=1}^{n} r_i(s-)\left[f_i^n(s-\alpha) - I_i(s-\alpha)\right]dU^n(s-\alpha).$$

By the random association of buffer overflow to user,

$$E|M_1^n(t)|^2 = O(1)E\sum_{s\le t}|dU^n(s)|^2 = O(1/\sqrt{n})EU^n(t).$$

Hence $M^n(\cdot)$ converges weakly to zero, and the left side of (3.6) can be used for (3.4), as $n\to\infty$.

By (3.3) and the conditions on $\xi^n(\cdot)$, and dividing each part of the term

$$\frac{\sum_i r_i(t-)r_i(t-\alpha)}{\sum_j r_i(t-\alpha) + na_m + \sqrt{n}\xi^n(t-\alpha)}$$

by n, we see that it converges weakly to the constant process, with values $v_2/[v_1+a_m]$, as $n\to\infty$. The above computations imply that, as $n\to\infty$, the buffer overflow control term in (3.1) is well approximatable by $(\kappa v_2/[v_1+a_m])U^n(t-\alpha)$. A very similar argument can be used if there is more than one buffer overflow at the same time.

Approximating the preemptive control. The preemptive control is defined by a bounded function $u_2(x,\rho)$ (it might depend on one or both of its arguments and is to be chosen) such that an incoming packet at time t is selected with probability $u_2(x^n(t), \rho^n(t))/\sqrt{n}$.[3] The effects of the preemptive control are analyzed similarly to those of the buffer overflow control and only a few comments will be made. The mean rate at which packets are selected at time $t - \alpha$ is

$$\begin{aligned}&\left[u_2(x^n(t-\alpha), \rho^n(t-\alpha))/\sqrt{n}\right]\\ &\qquad \times \left[\textstyle\sum_j r_i(t-\alpha) + na_m + \sqrt{n}\xi^n(t-\alpha)\right].\end{aligned}$$

The probability that any one is associated with controlled user i is $f_i^n(t-\alpha)$, and the chances that more than one are chosen from the same user in any finite time interval goes to zero as $n \to \infty$. Following the arguments used for the buffer overflow control, the conditional mean rate of change of $\sum_i r_i(\cdot)/\sqrt{n}$ at t is approximatable by, for large n,

$$\begin{aligned}&\frac{u_2(x^n(t-\alpha), \rho^n(t-\alpha))}{n} \sum_i r_i(t-) r_i(t-\alpha)\\ &\qquad\qquad \approx v_2 u_2(x^n(t-\alpha), \rho^n(t-\alpha)).\end{aligned} \tag{3.7}$$

The error terms and the martingale associated with the randomizations all go to zero as $n \to \infty$. Hence, this term times κ_1 well approximates the effects of the preemptive control in (3.1).

The limit dynamical equations. The Lipschitz condition in the proof of [12, Theorem 3.4.1, 3.5.1] and the tightness criterion in [12, Theorem 2.5.7] or [10, Theorem 2.7b] assures that the sequence $\{x^n(\cdot), \rho^n(\cdot), U^n(\cdot), L^n(\cdot)\}$ is tight in the Skorohod topology. The fact that some arguments are delayed is irrelevant. The equations satisfied by the weak sense limits (using the model (2.1) for the mice) are

$$\begin{aligned} d\rho(t) = u_1 dt - v_2 \Big[&\frac{\kappa}{v_1 + a_m} dU(t-\alpha)\\ &+ \kappa_1 u_2(x(t-\alpha), \rho(t-\alpha)) dt\Big],\end{aligned} \tag{3.8}$$

$$x(t) - x(0) = \int_0^t [\rho(s) - b]\, ds + w_m(t) + L(t) - U(t). \tag{3.9}$$

Equations (3.8) and (3.9) are suggestive even for more general models. They capture much of the essence of the AIMD and the preemptive control mechanisms, and retain the fundamental role of the randomness, all for an aggregated and scaled system.

Cost functions and nearly optimal controls for the physical system. In order to optimize the performance of the AIMD connections one can design the controls u_i (where in particular, u_2 will be implemented by the buffer management). Typically, $u_1(\cdot)$ is constant,

[3] By the limit theory, in the heavy traffic regime, $x^n(\cdot)$ is approximatable by a diffusion, hence varies "relatively slowly," even though the individual inputs and outputs are fast. $\rho^n(t)$ is computed from the current total input rate, and can often be well estimated. The selection need not be at random, provided that the average selection rates are close to the probabilities. This allows the possibility of selecting marked low priority packets, etc.

and we leave it that way in the sequel (we shall not optimize it, since it is assumed to be part of the description of the standardized source behavior). The quantities to penalize in the cost are queueing delay (to which the mice traffic may be sensitive) measured by $x(\cdot)$, the loss of throughput due to the control (measured by $-\rho(\cdot)$), and buffer overflow (measured by $U(\cdot)$)[4]. Let us work with a discounted cost criterion, where $\beta > 0$ can be as small as we wish, $c_0 > 0$, and the $k_i(\cdot) \geq 0$ are Lipschitz continuous:

$$\begin{aligned} &W(u_2) = \\ &\beta E \int_0^\infty e^{-\beta t} \left([k_1(x(t)) - k_2(\rho(t))] \, dt + c_0 dU(t) \right). \end{aligned} \tag{3.10}$$

The possibly nonlinear $k_i(\cdot)$ are useful, since (e.g.) we might wish to heavily penalize long queues, but not be too concerned with short queues.

Using the methods of heavy traffic analysis for controlled problems [12], it can be shown that the optimal costs for the physical problem converge to the optimal cost for the limit problem. The optimal control for the limit problem is of the switching curve type: For example if the delay is zero, then $u_2(x, \rho)$ takes the maximum value on one side and is zero on the other, and the separating curve is smooth. The switching curve character follows from a formal examination of the Bellman equation for the optimal value, since the control appears linearly in the dynamics and does not appear in the cost. The smoothness was borne out by the numerical computations. See. for example, 1, for the data given in Section VI. Such switching optimal controls are nearly optimal for the physical system for large n. We note that the cost (3.10) is well defined, since it can be shown that $E|\rho(t)| + EU(t) \leq a_1 + a_2 t$, for some $a_i \geq 0$.

We shall also consider using an ergodic cost criterion

$$\gamma(u) = \lim_{T \to \infty} E \frac{1}{T} \left[\int_0^T [c_1 x(s) - c_2 \rho(s)] \, ds + c_3 U(T) \right]. \tag{3.11}$$

At present, there is little theory concerning stability or ergodicity theory for delayed reflected diffusions such as ((3.8), (3.9)) or ((3.9), (5.15)) for the model of the next section. If the delay is zero then, for any control $u_2(\cdot)$, stability can be shown and the model ((3.9), (5.5)) has a unique invariant measure. In the numerical computations (where zero delay was always used), we were always able to compute an optimal control for the ergodic cost criterion (with cost and control well approximated by those for the discounted problem for small β), and both stability and convergence to the stationary distribution under the optimal (or other reasonable) controls was apparent.

[4] Penalizing buffer overflow may be important for two reasons. First, if the mice correspond to real time applications, then these applications will suffer due to losses. Secondly, the AIMD themselves may correspond to real time applications which are "TCP friendly", in which case lost packets are typically not retransmitted. Losses due to overflow then again degrade the quality of the communication.

4 Extensions of the Model of Section 3

No Buffer. Suppose that there is no buffer, so that if the total current packet rate exceeds the channel speed, then the excess packets are rejected. The forms of the input processes and channel speed (service rate) are as in the last section, but $U^n(\cdot)$ needs to be defined. Define

$$\begin{aligned} y^n(t) &= \left[C_n - \left(\sum_i r_i(t) + a_m n + \sqrt{n}\xi^n(t)\right)\right] /\sqrt{n} \\ &= \left[b - \rho^n(t) - \xi^n(t)\right], \end{aligned} \tag{4.1}$$

the scaled difference between the channel speed and input packet rate at t. Then the scaled number of rejected packets is

$$U^n(t) = \int_0^t [y^n(s)]^- \, ds = \int_0^t [\xi^n(s) + \rho^n(s) - b]^+ \, ds \tag{4.2}$$

Now, the value of the preemptive control $u_2(\cdot)$ at time t will be a function of only the scaled excess capacity $[y^n(t-\gamma)]^+$.

Suppose that the correlation time of $\xi^n(\cdot)$ is short (e.g., large μ_m in the special mice model). Then a law of large numbers argument can be used to show that $\xi^n(t)$ can be "integrated out" of (4.2), in that, as $n \to \infty$ and the correlation time goes to zero, the integrand can be replaced by the average over $\xi^n(t)$. This simplifies the expression for $U^n(\cdot)$, but it does not simplify the control $u_2(\cdot)$.

To simplify the control, suppose that a low pass filter is applied either to the scaled excess capacity $[y^n(\cdot)]^+$ or to the scaled current rate, and then the control $u_2(\cdot)$ is applied to the output of this filter. In fact, such filters are sometimes used in practice to reduce the dependence of the controls on sudden bursts. Suppose further that the correlation time of $\xi^n(\cdot)$ is short relative to that of $\rho^n(\cdot)$, so that the output of the filter can be approximated by $b - \rho^n(\cdot)$. Then the control takes the simpler form $u_2([b-\rho^n(t)]^+)$, delayed by α, and the limit equation for the scaled and centered rate process $\rho^n(\cdot)$ is

$$\begin{aligned} &\dot{\rho}(t) = u_1 dt \\ &-v_2\Big[\frac{\kappa}{v_1 + a_m}\dot{U}(t-\alpha) + \kappa_1 u_2([b - \rho(t-\alpha]^+)dt\Big], \end{aligned} \tag{4,3}$$

where $\dot{U}(t) = E_\rho\left[\xi^n(t) + \rho(t) - b\right]^+$, and the expectation is over the $\xi^n(t)$, as $n \to \infty$.

The randomness due to the mice in the arrival process does not appear explicitly in (4.3), but it affects the value of the expectation which yields the buffer overflow rate $\dot{U}(t)$. An analogous result holds for the model of Section 5, but there the randomness due to the arrival and departure processes of the controlled users remains in the limit. As for the case of this paragraph, only the $\xi^n(t)$ would be integrated out.

Random $r_i(0)$. In the rest of this section, we suppose that there is a buffer as in the original problem of Section 3. Suppose that the initial values of the rates are random, identically distributed, and mutually independent, with $Er_i(0) = v_1$ and $E[r_i(0)]^2 = v_2$. Then all the asymptotic results continue to hold.[5]

Randomly changing rates. In some internet applications, where a user sends a sequence of consecutive TCP connections, the rate of transmission is reinitialized for each new TCP transfer (e.g. HTTP/1.1). We propose a model of which this scenario is a special case. Suppose that the users change the packet transmission rates at random, and each with rate λ_0. The new rates (which are uniformly bounded) are chosen randomly with the same first two moments. More precisely, there are mutually independent Poisson processes $P_i(\cdot)$ all with rate λ_0. When $P_i(\cdot)$ jumps, the rate for user i is replaced. The set of replacements, over all users and time, is mutually independent, and independent of all other "driving" processes. Let q denote the canonical rate replacement, and define $v_1 = Eq$, $v_2 = Eq^2$, $\bar{v}_2 = E[q - v_1]^2 = v_2 - v_1^2$. Define $R^n(t) = \sum_i r_i(t)$. Then

$$\begin{aligned} dR^n(t) = \sqrt{n}u_1(t)dt - [\text{effects of controls}] \\ -\lambda_0 \left[R^n(t) - nv_1\right] dt + dM_r^n(t), \end{aligned} \tag{4.3}$$

where the martingale $M_r^n(\cdot)$ can be shown to have quadratic variation process

$$\begin{aligned} &\lambda_0 n \int_0^t \sum_i E\left[r_i(s) - q\right]^2 ds \\ &= \lambda_0 n \int_0^t \left[\frac{\sum_i r_i^2(s)}{n} - \frac{2v_1 \sum_i r_i(s)}{n} + v_2\right] ds, \end{aligned} \tag{4.4}$$

where the expectation is over q only. It can be shown that, on dividing (4.3) by $\sqrt{n}$ and taking limits, we get

$$d\rho(t) = u_1(t)dt - \text{effects of controls} - \lambda_0 \rho(t)dt + dw_r(t), \tag{4.5}$$

where the Wiener process $w_r(\cdot)$ has variance $2\lambda_0\bar{v}_2$, and the effects of the controls are represented as in (3.8). The limit system equations are (3.9) and (4.5).

Delay depending on the user. Up to now, all users had the same delay. The general theory can handle user-dependent delays. Suppose that user i has delay $\alpha_i \leq D < \infty$. If a buffer overflows at time s. The information will reach user i at time $s + t_{1,i}$. Thus, at time t, user i (if selected for the overflow) receives information concerning overflows at time $t - t_{1,i}$, and its response reaches the buffer $t_{2,i}$ units of time later, with $t_{1,i} + t_{2,i} = \alpha_i$. This leads to $dU^n(t - \alpha)$ in (3.6) being replaced by $dU^n(t - \alpha_i)$. To simplify matters in this brief

[5] This can be useful in adding to the modeling of AIMD connections a "slow start" phase such as in TCP, where the transition level (slow start threshold) between the slow start phase and the congestion avoidance phase varies (randomly) between the short connections. The slow start phase is then modeled for simplicity as an instantaneous jump to that transition level, since during slow start, the rate increases exponentially fast.

presentation, suppose that all initial rates are equal: $r_i(0) = v_1$. Then, for large n, the main term in (3.6) is approximately

$$\begin{aligned} -\kappa \sum_{i=1}^{n} r_i(t-) & \frac{r_i(t-\alpha_i)dU^n(t-\alpha_i)}{\sum_j r_j(t-\alpha_j) + na_m + \sqrt{n}\xi^n(t-\alpha_i)} \\ \approx & \quad -\frac{\kappa v_1^2}{a_m + v_1} \frac{1}{n} \sum_{i=1}^{n} dU^n(t-\alpha_j). \end{aligned}$$

More succinctly, with $\beta^n(\cdot)$ being a measure with mass $1/n$ at α_i, write

$$\frac{1}{n}\sum_{i=1}^{n} dU^n(t-\alpha_i) = \int_{t-D}^{t} dU^n(t-\alpha)\beta^n(d\alpha).$$

Suppose that the distribution of delays $\beta^n(\cdot)$ converges weakly to a distribution $\beta(\cdot)$. Then the $dU(t-\alpha)$ in (3.8) is replaced by $\int dU(t-\alpha)\beta(d\alpha)$. All else remains the same. Details of the proof are omitted.

5 A stochastic process of finite AIMD connections

In the model of Section 3, the number of users is fixed at n. Now, we consider a model where the controlled users arrive independently and randomly and leave at random, with the arrival process independent of the mice process. New users come from an unlimited population, with (Poisson) arrival rate λn. Each new user comes with an exponentially distributed number of data packets, with mean v_1/μ, and independent of the mice process and arrival time, where v_1 is a constant.[6]

We suppose that $1/\mu$ is large relative to the delay α. With this model, as with the previous ones, the buffer overflows (i.e., packet losses) are created by the physical process and not imposed. Note that, in this model, the mean amount of data in a new source does

[6] Exponential distribution of interarrival times and session duration are more appropriate for telephone calls than to data connections. Thus this model is expected to be more useful for VoIP applications that use TCP friendly mechanisms to regulate their rate. The "exponential" assumptions can be helpful even for the data connections for some preliminary dimensioning purposes.

Non exponential distributions can be handled as well, with an increase in the dimensionality of the limit model. For example, a k-stage Erlang model would require a k-dimensional process to represent the rate process. The mathematical development and results are similar. This higher dimensionality is a handicap for numerical computations, say via the Markov chain approximation method [11], or a pathwise approximation method. But it is not a serious handicap for simulation. Indeed, simulating the approximating limit model is substantially simpler than simulating the physical process, when there are very many users.

Experimentation with the basic model can lead to insights that are useful for more general cases. For example, numerical results for the basic model indicate that threshold controls, based on the rate only, provide good approximations to the values obtained by optimal controls. This observation provides a basis for getting good controls, which would be very hard to compute otherwise, for more general large size systems.

not depend on n. The parameter n scales the system speed and mean number of users only.[7] The source (i.e., the user) stays "active" until all data is sent, and then disappears. Time is still measured at the buffer and the mice model is (2.1). For simplicity, suppose that the initial rate of each new controlled source is v_1. There are analogs of all of the extensions of Section 4, but we stick to the basic model.

First suppose that there are no controls (constant transmission rate from each source) and buffer overflows are not retransmitted. Then the packets are sent from each active source to the buffer at a rate v_1. The mean time that a source is active is $1/\mu$, and the total rate at which the sources drop out at t is $\mu N^n(t)$, where $N^n(t)$ denotes the number of active sources. The (stationary) mean number of sources in the system is $n\lambda/\mu$. Hence, the analog of the channel speed C_n of Section 3 is

$$C^n = v_1 \frac{\lambda}{\mu} n + a_m n + b\sqrt{n},$$

where, again, $b\sqrt{n}$ denotes the excess capacity over the mean rate $n[v_1\lambda/\mu + a_m]$. On departure of a user, its rate v_1 is lost.[8]

Now suppose that the input rates from the non-mice sources are actually controlled. There are several approaches that one can take for the source departure process. One approach supposes that the departure rate (of an AIMD connection) is μ, and does not depend on the current packet transmission rate for the source. Then the lost packet rate if connection i leaves is $r_i(t)$. This situation arises when the AIMD connections correspond to real time applications that have a dynamic compression rate (which is then "TCP friendly"). In these applications, lost packets are not retransmitted (the possibility of lost packets might be anticipated in the coding). For simplicity in the development, this is the approach that will be taken. [9]

The dynamics and limit for the rate process. Only a sketch will be given, since the details are similar to those of Sections 3 and 4, except for the treatment of the randomness due to controlled user arrivals and departures. Write $N^n(t) = n\hat{N} + \sqrt{n}\nu^n(t)$, $\hat{N} = \lambda/\mu$. Since the user arrival process is Poisson and the user departure rate is constant,

$$dN^n(t) = \lambda n dt - \mu N^n(t) dt + dM_a^n(t) - dM_d^n(t). \tag{5.1}$$

Here $M_a^n(\cdot)$ is the martingale associated with the arrival process and has quadratic variation process $n\lambda t$, and $M_d^n(\cdot)$ is the martingale associated with the departure process and has

[7] The rate of arrivals of new users can be a smaller order of n, and then they would each have an amount of data that would depend on n. E.g., rate of arrival $O(\sqrt{n})$, with data $O(\sqrt{n})$. In this case the rate of work on each source is $O(\sqrt{n})$, so that the average sojourn in the system is still $O(1)$.

[8] Strictly speaking a source should not depart until an acknowledgment of its last transmission has been received. But our approximation to the actual departure rule has little effect, since the order of lost packets is still $O(\sqrt{n})$, and μ is large.

[9] An alternative approach replaces the value of μ by a time varying quantity to reflect the fact that even if the service rate per source changes the total amount of data per source doesn't. For example, if the allowed data rate for an AIMD connection is cut in half due to an increase in the number of sources, then the value of the connection departure rate for that source should be cut in half. The mathematical development of this situation is much harder.

quadratic variation process $\mu \int_0^t N^n(s)ds$. For simplicity, suppose that $N^n(\cdot)$ is stationary. It follows from this, (5.1), and the cited values of the quadratic variations, that the sequence $N^n(\cdot)/n$ converges weakly to a process with constant value λ/μ, as $n \to \infty$. Also, $\nu^n(\cdot)$ satisfies

$$d\nu^n(t) = -\mu\nu^n(t)dt + [dM_a^n(t) - dM_d^n(t)]/\sqrt{n}. \tag{5.2}$$

The quadratic variation of the scaled martingale term in (5.2) is $\lambda t + \mu \int_0^t N^n(s)ds/n$, which converges weakly to 2λ. The sequence $\nu^n(\cdot)$ converges weakly to $\nu(\cdot)$, where

$$d\nu(t) = -\mu\nu(t)dt + dw(t), \tag{5.3}$$

where $w(\cdot)$ is a Wiener process with variance 2λ.

Returning to the rate process, write $\sum_i r_i(t) = R^n(t) = n\hat{R} + \sqrt{n}\rho^n(t)$, where $\hat{R} = v_1\lambda/\mu$. The process $R^n(\cdot)$ satisfies

$$\begin{aligned} dR^n(t) &= \lambda v_1 n dt - \mu R^n(t)dt + u_1(t)\sqrt{n}dt \\ &\quad -[\text{effects of overflow and preemptive controls}] \\ &\quad +v_1 dM_a^n(t) - dM_{d,1}^n(t), \end{aligned} \tag{5.4}$$

where $M_{d,1}^n(\cdot)$ is the martingale associated with the "rate departure" process and it has quadratic variation process $\mu \int_0^t \sum_i r_i^2(s)ds$. This, divided by n, converges weakly to the process with values $v_1^2\lambda t$, as $n \to \infty$. Finally, it is not hard to show that $\rho^n(\cdot) \Rightarrow \rho(\cdot)$, where

$$\begin{aligned} d\rho(t) &= -\mu\rho(t)dt + [\lambda/\mu]u_1 dt \\ &\quad -v_1^2\Big[\frac{\kappa[\lambda/\mu]}{v_1[\lambda/\mu] + a_m} dU(t-\alpha) \\ &\quad +\kappa_1[\lambda/\mu]u_2(x(t-\alpha), \rho^n(t-\alpha))dt\Big] + v_1 dw. \end{aligned} \tag{5.5}$$

Approximations to the optimal control via the limit model. The limit system equations are (3.9) and (5.5). The comments made below (3.10) concerning the costs and controls, and on the approximation of optimal costs and controls for the physical system by the optimal controls for the limit, also hold here.

Comment concerning retransmission of lost packets. The model that has been discussed did not involve retransmission of lost packets. Such retransmission is not hard to model, if desired. Basically, a lost packet implies that more data remains to be transmitted, hence a reduction in the user and data departure rate. This reduction can be expressed in terms of the buffer overflow process $U^n(\cdot)$, and the details will be given elsewhere.

6 Numerical Data: Optimal Preemptive Controls

It is not possible at present to compute optimal policies when there is a delay, so we set $\alpha = 0$. But the results do shed light on the system behavior. The results suggest that, even with a non zero delay, a rate based threshold control will yield good results.

Numerical results were obtained for the optimal control and costs for the model of Section 5 for cost (3.10), with $k_1(x) = c_1 x, k_2(\rho) = c_2\rho$, and either small β, or the ergodic cost analog. The results were nearly the same when $\beta \leq .02$, and the ergodic case will be described. The Markov chain approximation method [11], the most appropriate current numerical method for controlled reflected diffusions, was used. Only a few details can be given here. Use $u_1 = 1, \lambda/\mu = 4, b = 1, v_1 = 1.5, a_m = 4, \kappa = \kappa_1 = .5$, $\sigma_m^2 = 4$, the upper bound on $u_2(x,\rho) \leq 1$, the buffer capacity is $12.8\sqrt{n}$ packets, and $c_0 = 100, c_1 = 1, c_2 = 5$, reflecting our desire to penalize lost packets most heavily. The mice account for about 40% of the traffic and the system is quite "noisy," since the variances of the Wiener processes driving (x,ρ) are $(9, 4.5)$.

The optimal preemptive controls are determined by a switching curve: $u_2(x,\rho) = 0$ below the curve and equals its maximum value above the curve. The curve obtained for our example in the asymptotic regime is given in Fig 1. As we see, in (x,ρ) space, the curve is initially (for small x) almost a straight line with a slightly decreasing slope as x increases. As the buffer fills up, the slope becomes sharply more negative, as expected. The optimal cost for the problem with preemptive control was about 1/10th of that without. In general, The values of the cost components (stationary mean values of $x(t), \rho(t)$, and $\lim_{t\to\infty} EU(t)/t$) are more significant than the optimal cost, since they give us information on the tradeoffs. Optimal control is not of interest for its own sake, but rather for the information provided on good design, and tradeoffs among the cost components as the weights change.

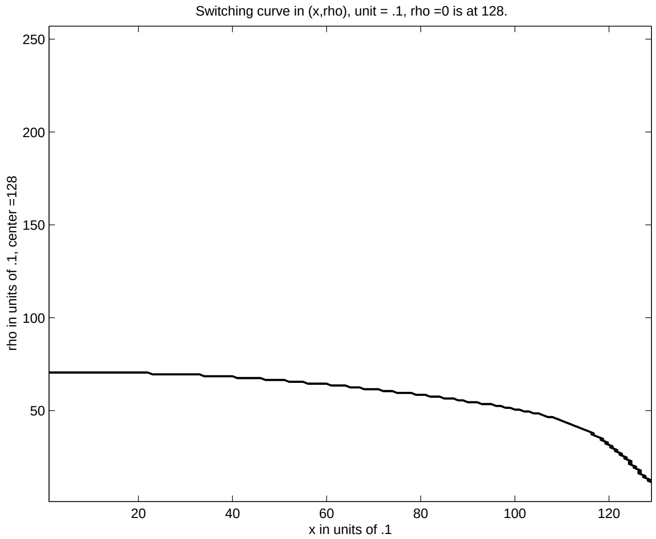

Figure 1: The switching curve

For the uncontrolled problem, the total buffer overflow rate (for all users) was $5.35\sqrt{n}$, vs $0.28\sqrt{n}$, under optimal control for the given cost coefficients. The mean queue was virtually full for the uncontrolled case, compared to an average of one-third full under optimal control. The total input rate for the controlled users was reduced by an average of $0.36\sqrt{n}$ under optimal preemptive control, compared with an increase of $6.3\sqrt{n}$ with no control. Thus to get an improvement in overflow of about 20 times cost a fractional reduction in the throughput of $(6.3 + 0.36)/[(v_1\lambda/\mu) + a_m]\sqrt{n}) = 0.666/\sqrt{n}$.

If the buffer size is increased, its average percentage occupancy is about the same (queue size is not weighted heavily), the average $E\rho$ increases, and the average overflow rate does not change dramatically (e.g., doubling the buffer only halves the overflow, under our parameters). The optimal system adapts to an increased buffer size mainly by increasing the average flow, keeping the queue size roughly in proportion to the buffer size, an interesting fact in itself. Of course, a larger weight on x will reduce the average queue size.

These numbers illustrate the type of tradeoffs that are possible. One pays for reduced overflow by reduced packet rate. But the packet rate is reduced only where it does the most good. The tradeoffs vary with the cost coefficients. To use the method effectively, one makes a series of runs, varying the coefficients c_i This yields a set of possible tradeoffs between the competing criteria. In each case, the tradeoff is under an optimal control. The approach to the use of numerical methods and heavy traffic approximations is similar to what was done for the problem of input control of a multiplexer system in [13]. A comparison with threshold controls shows that the effects of the optimal control can be well approximated by a threshold control depending on ρ (or on the overall low pass filtered rate, as discussed above (4.3)) only, for appropriate values of the threshold. The cost components for the no control, optimal, and threshold cases are summarized in Table 1. If the threshold controls are activated only when the buffer exceeds some modest level, their performance is even better. Keep in mind that the described optimal control and costs are for a very heavy weight on overflow.

Table 1. Cost components.

under run type	buf overflow/$\sqrt{n}$	Ex	$E\rho$
no cont.	5.35	11.92	6.35
opt. cont.	.28	4.4	-.36
thresh $\rho = 0$	.69	7.6	1.46
thresh $\rho = -1$	.48	6.4	.98
thresh $\rho = -3$	.33	4.9	.2

7 Appendix: Comparison With a Fluid Model

Reference [19] also dealt with a limit approximation for large systems and justified the use of a delayed deterministic differential equation as an approximation for a certain class of problems. Since there are major differences between that work and this, apart from the different scaling, a brief discussion of a few of the differences is worthwhile.

In the basic model of Section 3, capacity (i.e., bandwidth) scales linearly with n, and so does the number of sources. The packet rate for each source is $O(1)$. Our general approach also allows the possibility that the number of sources grows more slowly with n, with the packet rate per source growing accordingly faster. While there are no explicit capacity constraints in [19], it is clear that the bandwidth (BW) is proportional to their n^2,and we use this fact below. They use a fixed number of connections of the order of $\sqrt{\mathrm{BW}}$ (and no analog of the models of Sections 4 and 5), each sending packets at rate $O(\sqrt{\mathrm{BW}})$. The number of mice connections grows linearly with $\sqrt{\mathrm{BW}}$, and so does the rate of each mouse. Time is divided into "decision intervals" of length $O(1/\sqrt{\mathrm{BW}})$, and the rates are (perhaps unrealistically) averaged over these successive intervals before feedback and decisions. This averaging, over $O(\sqrt{\mathrm{BW}})$ packets before feedback, effectively eliminates the randomness due to the mice. We work closer to system capacity where the effects of random variations are greater, and it is the true instantaneous randomness that causes the losses and activates the controls.

The total overall rates of increase of the packet rate due to the slow additive control is the same here and in [19]. In [19] the rate of each connection in the nth model (the one corresponding to n TCP connections) increases by $1/n$ per each time slot, so that in terms of real time the rate of increase does not depend on n. Thus the total rate of increase is of the order of $\sqrt{\mathrm{BW}}$. In our model, the packet losses of each AIMD source is random and determined by the loss process associated with that source. this is in particular in conformance with the objectives of buffer management schemes [7]. In [19], in contrast, all AIMD sources have the same instantaneous dynamics, hence identical losses.

References

[1] E. Altman, K. Avratchenkov and C. Barakat, "A stochastic model of TCP/IP with stationary random losses", *ACM SIGCOMM 2000*.

[2] F. Baccelli and D. Hong, "A.I.M.D, Fairness and Fractal Scaling of TCP Traffic" Technical Report, April 2001, RR-4155, INRIA Rocquencourt, France, 2001.

[3] S. Ben Fredj, T. Bonald, A. Proutiere, G. Regnié and J. W. Roberts, "Statistical bandwidth sharing: a study of congestion at flow level", *SIGCOMM'*, 2001.

[4] T. Bu and D. Towsley, "Fixed point approximation for TCP behaviour in an AQM network", *ACM SIGMETRICS*, June 2001.

[5] P. Brown, "Resource sharing of TCP connections with different round trip times", *IEEE Infocom*, Mar 2000.

[6] V. Dumas, F. Guillemin and P. Robert, "A Markovian analysis of AIMD algorithms", *Advances in Applied Probability*, 34(1) 85-111, 2002.

[7] S. Floyd and V. Jacobson, "Random Early Detection gateways for Congestion Avoidance" *IEEE/ACM Transactions on Networking*, 1(4):25–39, 1993.

[8] C. Hollot, V. Misra, D. Towsley and W.-B. Gong, "A control theoretic analysis of RED" IEEE INFOCOM, 2001

[9] P. Kuusela, P. Lassila, J. Virtamo, "Stability of TCP-RED Congestion Control", in proceedings of ITC-17, Salvador da Bahia, Brazil, Dec. 2001, pp. 655-666.

[10] T.G. Kurtz. *Approximation of Population Processes*, volume 36 of *CBMS-NSF Regional Conf. Series in Appl. Math.* SIAM, Philadelphia, 1981.

[11] H.J. Kushner and P. Dupuis, *Numerical Methods for Stochastic Control Problems in Continuous Time*, Springer-Verlag, Berlin and New York, 1992: Second edition, 2001

[12] H.J. Kushner. *Heavy Traffic Analysis of Controlled Queueing and Communication Networks.* Springer-Verlag, Berlin and New York, 2001.

[13] H.J. Kushner, D. Jarvis, and J. Yang. Controlled and optimally controlled multiplexing systems: A numerical exploration. *Queueing Systems*, 20:255–291, 1995.

[14] T.V. Lakshman and U. Madhow, "The performance of TCP/IP for networks with high bandwidth-delay products and random loss", *IEEE/ACM Transactions on Networking*, Jun 1997.

[15] S. H. Low, "A Duality Model of TCP and Queue Management Algorithms", ITC Specialist Seminar on IP Traffic Measurement, Modeling and Management, September 18-20, 2000, Monterey, CA. To appear IEEE/ACM Trans. on Networking, 2003.

[16] Laurent Massoulie, "Stability of distributed congestion control with heterogeneous feedback delays", IEEE Transactions on Automatic Control 47(2002) 895-902.

[17] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose, "Modeling TCP throughput: A simple model and its empirical validation", *ACM SIGCOMM*, Sep 1998.

[18] K. K. Ramakrishnan, S. Floyd, and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP" RFC 3168, Proposed Standard, September 2001,
available at ftp://ftp.isi.edu/in-notes/rfc3168.txt

[19] S. Shakkottai and R. Srikant. How good are deterministic fluid models of internet congestion control. In *Proc., IEEE INFOCOM, 2002*, New York, 2002. IEEE Press.

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399