Modeling RED with Idealized TCP Sources

P. Kuusela, P. Lassila, J. Virtamo and P. Key*

Networking Laboratory
Helsinki University of Technology (HUT)
P.O.Box 3000, FIN-02015 HUT, Finland
Email: {Pirkko.Kuusela, Pasi.Lassila, Jorma.Virtamo}@hut.fi

*Microsoft Research
Cambridge, United Kingdom
Email: peterkey@microsoft.com

Abstract

We analyze the dynamic behavior of a single RED controlled queue interacting with a large population of idealized TCP sources, i.e., sources obeying the rules of linear increase and multiplicative decrease. The aggregate traffic from this population is modeled in terms of the time dependent expected value of the packet arrival rate which reacts to the packet loss taking place in the queue. The queue is described in terms of the time dependent expected values of the instantaneous queue length and of the exponentially averaged queue length, for which we also derive a pair of differential equations. This provides us with a complete model for the dynamics of the system which we use to explore transient and equilibrium behavior. The accuracy of the model is verified by comparison with simulated results.

Keywords: congestion control, Random Early Detection, TCP

1 Introduction

Active queue management (AQM) algorithms have been recommended by IETF as effective mechanisms to control congestion in network routers. We consider one AQM method, namely the RED algorithm [1], and model the interaction between a RED controlled queue and an idealized TCP source population.

A full description of the problem consists of a closed system with two interacting parts (see Figure 1.): a) a queue receiving an aggregate flow of packets and reacting dynamically to the changes in the total flow rate under the control of the RED algorithm, and b) a population of TCP sources, each of which reacts to the packet losses at the queue as determined by the TCP flow control mechanism. Part a) of the problem was considered by Lassila and Virtamo [2]. Here we aim at closing the control loop back to the TCP sources describing the interaction of parts a) and b). However, we are not modeling the entire TCP behavior, but rather the so called congestion avoidance part of it, i.e., the part where a source increases its congestion window linearly and decreases it multiplicatively.

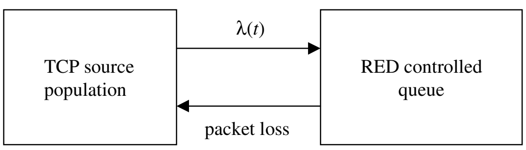

Figure 1.: Interaction between the TCP population and the RED controlled queue.

This paper describes a system of coupled differential equations for the population of TCP streams and the RED controlled queue. Such a model can be used to explore, for example, parameter settings or stability surfaces (which are difficult to study via simulation). Moreover, the model can be extended to handle multiple classes and different round trip times. The emphasis of this paper is to derive the model and verify its accuracy. A follow-up paper [3] utilizes this model in analyzing the stability of the TCP-RED interaction and showing how it depends on the system parameters. See also [3] for references on the stability of the TCP behavior.

We use a model similar to Kelly's [4] for TCP congestion avoidance behavior describing the average arrival rate $\lambda(t)$ of packets from a large collection of TCP sources. To rigorously derive the analytical model we then assume that the arrival process to the RED controlled queue can be approximated by an inhomogeneous Poisson process with a stochastically time varying arrival rate $\lambda(t)$. Such an approximation is reasonable as we are modeling the aggregate traffic from a large collection of TCP sources. We emphasize on deriving equations carefully and indicating the stochastic approximations that have been made. The accuracy of the model is verified by simulations. In the simulation, none of the stochastic approximations are used, and the very good agreement with the model shows that the stochastic assumptions are not important.

Recent independent modeling work by Misra et al. [5] is similar to our approach in which differential equations are used to describe the system variables. The work in [5] is complementary to ours; we develop a more detailed model for the queue dynamics, while [5] focuses more on the source dynamics. In addition, our term describing the increase in the TCP window size is more accurate. In the RED buffer model, [5] assumes an infinite and non-emptying buffer and the model derivation results in an additional modeling parameter to be tuned. On the contrary, our queue model takes overflows and empty queues into account, and does not have additional parameters. Overall, we find that the most accurate system model is obtained by using our RED model and dividing the TCP population into homogeneous subpopulations with identical link delays, each described by the Kelly's aggregate TCP model, and then making the straightforward model extensions to handle queue dependent round trip times, timeouts and the network case, as is done in [5].

In a recent paper Laalaoua et al. [6] model RED with adaptive traffic sources. The approach is to iterate the packet loss probability and the traffic intensity in tandem with solving a diffusion equation for the instantaneous queue length, thus giving the transient behavior of the queue between the iteration steps. Altman et al. [7] consider a TCP behavior together with losses that are presented by an exogenous stationary point process (and hence independent of the TCP window sizes) and derive the first two moments of the TCP window size. Also, Brown [8] provides a model for the window evolution of the TCP connections with different round trip times in a tail-drop queue.

This paper is organized as follows: We begin by describing the RED algorithm in Section 2. Models for the TCP population and the RED controlled queue are derived separately in Sections 3 and 4, and then combined in Section 5. In that section we also include the link delays resulting in a retarded functional differential equation (RFDE) model, and discuss various extensions of the model, e.g., the possibility to handle differentiated service classes with simple modifications and dealing with inhomogeneous round trip times. The accuracy of stochastic assumptions and approximations are verified in Section 6 where we compare the model to a simulated system. Conclusions are given in Section 7.

2 The RED algorithm

For the purpose of this paper we will use a simplified version of the original RED algorithm, which is typical in RED analyzes and also done, e.g., in [5]. To be specific, the following simplifications are made. We do not cover the case where the exponentially weighted average (EWA) queue length is computed differently when the arrival occurs into an empty queue. However, as we are primarily interested in heavily loaded queues, the effect of this special handling is of less importance. Additionally, we do not model the counter which measures the time (in terms of packet arrivals) since the previous packet drop. (Note that the system with the counter can be approximated by a counterless system simply by using a counterless system with twice the value for the RED parameter $p_{\max}$, see, e.g., [1], or [2].)

Our simplified RED algorithm works as follows: Consider a queue with a finite buffer which can hold up to K packets at a time. Let q_n be the queue length and s_n be the EWA queue length (to be defined below) at the n^{th} arrival. For each arriving packet we compute the EWA queue length s_n with

$$s_n = (1-\beta)s_{n-1} + \beta q_n,$$

where $0 < \beta < 1$ is an appropriate constant, typically of the order 10^{-3}. If the buffer is full, the packet is lost. If $s_n < T_{\min}$, the arriving packet is accepted; if $s_n > T_{\max}$ the packet is discarded. In the intermediate range $T_{\min} \le s_n \le T_{\max}$, the packet is discarded with probability

$$p(s_n) = \frac{p_{\max}(s_n - T_{\min})}{T_{\max} - T_{\min}},$$

and accepted otherwise. Typical recommendations for setting the parameters are: $T_{\min} = 5$, $T_{\max} = K/2$ and $p_{\max} = 0.1$, see [9]

3 The TCP model

We aim at modeling the behavior of a TCP connection in its congestion avoidance phase. Our model is essentially the same as in [4], however, we have altered its derivation to get a clearer idea of the approximations involved.

Consider m identical TCP sources with constant round trip time R (random queuing delays are not included). To simplify the notation, we temporarily neglect the delays in the time arguments and derive the equations as if the changes to the window sizes occurred instantaneously at packet transmission epochs. The time delays are introduced later in Section 5. Denote by $w_i(t)$ the window of TCP source i at time t. Correspondingly, the aggregate has a window $W(t) = \sum_i w_i(t)$. The throughput (arrival rate), $\lambda_i(t)$, of source i is $\lambda_i(t) = w_i(t)/R$. Consider a small time interval $(t, t+\Delta t)$, or Δt for brevity. Let $A_i(\Delta t)$ denote the event that there is exactly 1 arrival from source i during Δt. Additionally, given that there is a packet arrival in Δt from source i, let $L_i(t)$ denote the event that this packet is lost and $L_i^c(t)$ its complement. Our basic assumption on the traffic generated by a TCP source is that it is periodic with time dependent periods, but for each TCP source the packet transmission phase is random. This implies that given the current window size $w_i(t)$ of source i, the probability of an arrival in Δt is $\mathrm{P}\{A_i(\Delta t) \,|\, w_i(t)\} = \lambda_i(t)\Delta t = w_i(t)\Delta t/R$ and the probability of having more than 1 arrival in Δt equals zero. In TCP congestion avoidance, for each successfully transmitted packet source i increases its transmission window by $1/w_i(t)$ (corresponding to an increase by 1 for $w_i(t)$ packets), and for each lost packet the window is halved. We can compute the expectation of the change in the aggregate arrival rate $\lambda(t) = \sum_i \lambda_i(t)$, $\mathrm{E}[\Delta\lambda(t)]$, by conditioning on the set of window sizes $\{w_i(t)\}$ and $A_i(\Delta t)$ as follows

$$\begin{aligned}
\mathrm{E}[\Delta\lambda(t)] &= \frac{1}{R}\mathrm{E}\left[\mathrm{E}\left[\Delta W(t)\,|\,\{w_i(t)\}\right]\right] = \frac{1}{R}\mathrm{E}\left[\mathrm{E}\left[\sum_i \Delta w_i(t)\Big|\{w_i(t)\}\right]\right] \\
&= \frac{1}{R}\mathrm{E}\left[\sum_i \mathrm{E}\left[\Delta w_i(t)|\{w_i(t)\}, A_i(\Delta t)\right]\cdot \mathrm{P}\{A_i(\Delta t)\,|\,\{w_i(t)\}\}\right] \\
&= \frac{1}{R}\mathrm{E}\left[\sum_i \left(\mathrm{P}\{L_i^c(t)\,|\,\{w_i(t)\}\}\frac{1}{w_i(t)} - \mathrm{P}\{L_i(t)\,|\,\{w_i(t)\}\}\frac{w_i(t)}{2}\right)\frac{w_i(t)}{R}\Delta t\right] \\
&= \frac{\Delta t}{R^2}\mathrm{E}\left[\sum_i \left(\mathrm{P}\{L_i^c(t)\,|\,\{w_i(t)\}\} - \mathrm{P}\{L_i(t)\,|\,\{w_i(t)\}\}\frac{w_i^2(t)}{2}\right)\right].
\end{aligned}$$

If we further assume that the $w_i(t)$ processes are independent of each other and that the number of TCP sources is large, each individual source experiences a packet loss as determined by the queue receiving the aggregate traffic. The conditional probability $\mathrm{P}\{L_i(t)\,|\,\{w_i(t)\}\}$ corresponds then to the loss probability in the queue at time t, which is denoted here by $P_L(t)$. Proceeding with the derivation, we obtain

$$\mathrm{E}[\Delta\lambda(t)] = \frac{m\Delta t}{R^2}\left((1-P_L(t)) - P_L(t)\frac{\mathrm{E}[w_i^2(t)]}{2}\right).$$

We can next make a comment on the effect of the distribution of $w_i(t)$ on the coefficients above. Here we temporarily suppress the time dependence from the notation. We assume that the distribution of w_i is such that it scales with $\mathrm{E}[w_i]$, i.e. we assume that $\mathrm{E}[w_i^2] = k\cdot \mathrm{E}[w_i]^2$, where k is some constant. If w_i is a fixed constant then we have trivially that $\mathrm{E}[w_i^2] = \mathrm{E}[w_i]^2 = \mathrm{E}[W]^2/m^2$, i.e., $k=1$. On the other hand, if each TCP source in the population is assumed to vary its window between $\frac{2}{3}W/m$ and $\frac{4}{3}W/m$ and the phases in the population are random, then each w_i is uniformly distributed in $[\frac{2}{3}W/m, \frac{4}{3}W/m]$. Now large windows contribute more to the window decrease due to the packet loss. Then we obtain

$$\mathrm{E}[w_i^2] = \frac{3m}{2W}\int_{\frac{2W}{3m}}^{\frac{4W}{3m}} w_i^2\frac{3m}{2W}dw_i = \frac{28}{27}\left(\frac{W}{m}\right)^2,$$

i.e. we have $k = 28/27 \approx 1$. Thus, this extension practically results in the same model as above. In addition, numerical experiments with more realistic window size distributions yield only minor changes in the coefficient k. Hence, in this paper we use the simple assumption that $k=1$ and $\mathrm{E}[w_i^2] = \mathrm{E}[w_i]^2$.

Then, using the approximation $\mathrm{E}[w_i^2(t)] = \mathrm{E}[w_i(t)]^2 = \mathrm{E}[W(t)]^2/m^2$, we get

$$\mathrm{E}[\Delta\lambda(t)] = \left((1-P_L(t))\frac{m}{R^2} - P_L(t)\frac{\mathrm{E}[\lambda(t)]^2}{2m}\right)\Delta t$$

By the linearity of the Δ operator, $\mathrm{E}[\Delta\lambda(t)] = \Delta\mathrm{E}[\lambda(t)]$, denoting with $\mathrm{E}[\lambda(t)] = \bar{\lambda}(t)$, and letting $\Delta t \to 0$, we arrive at

$$\frac{d}{dt}\bar{\lambda}(t) = (1-P_L(t))\frac{m}{R^2} - P_L(t)\frac{\bar{\lambda}^2(t)}{2m}. \tag{1}$$

4 Model for the RED Controlled Queue

The aim is to develop a continuous time model for the transient behavior of the expectations of the two queue length processes, the exponentially averaged queue length $s(t)$ and the instantaneous queue length $q(t)$. These expectations are denoted here by $\bar{s}(t)$ and $\bar{q}(t)$, respectively.

With the earlier assumptions on a large TCP population and random phases, the aggregate stream can be approximated by a time varying inhomogeneous Poisson process with rate $\lambda(t)$. Strictly speaking, $\lambda(t)$ is itself a stochastic process but we may assume that the fluctuations are small and approximate $\lambda(t) \approx \mathrm{E}[\lambda(t)]$. With this, the queue part of our model consists of a RED controlled queue receiving an inhomogeneous Poisson arrival stream with a deterministic time varying rate $\mathrm{E}[\lambda(t)] = \bar{\lambda}(t)$ governed by Eq. (1). This system has been previously studied in [2], where by using a similar technique as in the previous section, the expected change in $\bar{q}(t)$ and $\bar{s}(t)$ during Δt has been obtained by conditioning on certain random variables. As a result the following approximation has been obtained for describing the time dependent behavior of $\bar{s}(t)$ and $\bar{q}(t)$:

$$\begin{cases} \dfrac{d}{dt}\bar{s}(t) = \bar{\lambda}(t)\beta(\bar{q}(t) - \bar{s}(t)), \\ \dfrac{d}{dt}\bar{q}(t) = \bar{\lambda}(t)\Big(1 - \pi_K(t)\Big)\Big(1 - p(\bar{s}(t))\Big) - \mu\Big(1 - \pi_0(t)\Big), \end{cases} \tag{2}$$

where π_0 and π_K denote the steady state probabilities for the queue to be empty and full, respectively. Note that, in the equation for $\bar{q}(t)$ above, the first term is the expected rate at which packets are admitted to the queue and the second term is the expected rate at which packets leave the queue.

To evaluate the terms $\pi_0(t)$ and $\pi_K(t)$ we use a quasi-stationarity approximation. The idea is to replace $\pi_0(t)$ and $\pi_K(t)$ with ones that relate π_0 and π_K directly to the system variable $\bar{q}(t)$ by using known steady state relations for π_0, π_K and $\bar{q}$, i.e., in our case those of the M/D/1/K system (in [2] an M/M/1/K queue was considered). To this end let $\pi_k^\infty(\rho)$ denote the steady state probability of state k in the infinite capacity M/D/1 system with load ρ. These state probabilities can be obtained by using, e.g., the algorithm by Virtamo [10]. Then the steady state probability of state k, $\pi_k(\rho)$, in the finite system can be obtained from the following relations (see [11, p. 230])

$$\begin{aligned} \pi_j(\rho) &= \frac{\pi_j^\infty(\rho)}{\pi_0^\infty(\rho) + \rho G(K)}, \quad j = 0, \ldots, K-1, \\ \pi_K(\rho) &= 1 - \frac{G(K)}{\pi_0^\infty(\rho) + \rho G(K)}, \\ G(K) &= \sum_{j=0}^{K-1} \pi_j^\infty(\rho). \end{aligned}$$

Additionally, we have trivially that $\bar{q}(\rho) = \sum_{j=0}^{K} j\pi_j(\rho)$.

The steady state functions $\pi_0(\bar{q})$ and $\pi_K(\bar{q})$ are computed by eliminating ρ from the above relations. This approximation technique is also similar to the one used in [12]. In practice, e.g., the function $\pi_0(\bar{q})$ is best derived by computing pairs of values $(\bar{q}(\rho), \pi_0(\rho))$ for a suitably dense discretization of ρ providing a discretized version of $\pi_0(\bar{q})$. Then one can use interpolation between the discretization points to compute $\pi_0(\bar{q})$ for any value of $\bar{q}$.

In this paper we use the M/D/1/K system as an example, but the above procedure applies more generally to any packet size distribution. The queue length distribution of the M/G/1 system is given by the well known Pollaczek-Khintchine formula [11, p. 230] and the distribution can be computed algorithmically as presented in, e.g., [13, p. 266]. Finally, the distribution of the M/G/1/K system is obtained from the infinite buffer system in the same way as was done above.

The applicability of the M/G/1/K approximation depends on how fast $\bar{\lambda}(t)$ can change. If $\bar{\lambda}(t)$ is slowly varying, the queue can be considered to be in a quasi-stationary state. In our case the changes in $\bar{\lambda}(t)$ are driven by packet losses, which are primarily due to the RED packet discarding (buffer overflows should be rare in a RED controlled queue). The RED packet discarding probability depends on the averaged queue length $s(t)$, whose time scale of changes is determined by β. In practice β is small (of the order 10^{-3}), and thus the process $s(t)$ moves slowly and, also, the changes in $\bar{\lambda}(t)$ occur relatively slowly. Moreover, our simulations in Section 6 verify the applicability of this approximation.

5 The Model for the Interacting System

In combining the models for the TCP population and the RED controlled queue we include the link delays in the system (but queuing delays are not modeled and hence the delays are not time dependent). In the following model $\bar{\lambda}(t)$ denotes the expected TCP throughput at the source. The acknowledgments arrive at the source with the delay R causing the update action to the current sending rate, and the amount of update depends on the current rate. Also, the TCP population will adjust its current sending rate based on packet losses having taken place in time $R/2$ prior to the current time. Similarly, the packets sent by the TCP population arrive at the queue after the link delay of $R/2$ and hence the queue will observe the $R/2$ delayed throughput. This gives rise to a delay differential equation (or RFDE):

$$\begin{cases} \frac{d}{dt}\bar{s}(t) = \bar{\lambda}(t-R/2)\beta(\bar{q}(t)-\bar{s}(t)), \\ \frac{d}{dt}\bar{q}(t) = \bar{\lambda}(t-R/2)(1-\pi_K(\bar{q}(t)))(1-p(\bar{s}(t))) - \mu(1-\pi_0(\bar{q}(t))), \\ \frac{d}{dt}\bar{\lambda}(t) = -\frac{P_L(t-R/2)}{2m}\bar{\lambda}(t)\bar{\lambda}(t-R) + (1-P_L(t-R/2))\,\frac{m}{R^2}\frac{\bar{\lambda}(t-R)}{\bar{\lambda}(t)}, \end{cases} \tag{3}$$

$$P_L(t) = p(\bar{s}(t)) + \pi_K(\bar{q}(t)) - p(\bar{s}(t))\pi_K(\bar{q}(t)).$$

Here we have taken a symmetric delay structure, but any other delay structure can be handled with obvious modifications. Also, note that in the TCP model in Section 3, P_L denoted the probability of the packet loss observed by the TCP population. In the above system, packets may be dropped due to the RED admission control or the buffer overflow. Hence we approximate the packet loss by $P_L = p(\bar{s}) + \pi_K(\bar{q}) - p(\bar{s})\pi_K(\bar{q})$ (see [2] for discussion on independence assumptions). Additionally, observe that in the last differential equation in (3), $\bar{\lambda}(t)$ appears with two time arguments: $\bar{\lambda}(t-R)$ represents the update rate at time t, whereas $\bar{\lambda}(t)$ arises from the amount of change to the current rate.

After the TCP population activates at time t_0, the queue dynamics activate at time $t_0+R/2$ whereas the TCP congestion dynamics activate at time t_0+R. Functions $\bar{q}, \bar{s} : [t_0, R/2] \to \mathbb{R}$ and $\bar{\lambda} : [t_0, R] \to \mathbb{R}$ are given as initial data. Also, note that when there are no packet losses, the slope of the linear increase in the throughput is slightly less than m/R^2 due to the term $\bar{\lambda}(t-R)/\bar{\lambda}(t)$, a fact also seen in the simulations.

5.1 Two traffic classes

We illustrate how the above model can be extended to different traffic classes following the treatment in [14]. Assume that we have a population of m TCP sources, each with a constant R, and the throughput of the aggregate in denoted by λ_1. In addition there is some UDP traffic with intensity λ_2. For simplicity both traffic sources are assumed to send packets of equal size and as a local approximation the packet arrival intensity from each source is assumed to be Poissonian. Also the weight parameter β in the RED algorithm is assumed to be the same in each class.

We refer to the TCP traffic as class 1 traffic and the UDP traffic is denoted by class 2. As in Section 4 we can write differential equations describing the evolution of EWA queue lengths $\bar{s}_i$ in each class: $d\bar{s}_i/dt = \bar{\lambda}_i\beta\,(\bar{q}_i - \bar{s}_i)$. As the packet size is the same in each class, we may consider the queue with the aggregated arrivals from both classes and denote the expected queue length by $\bar{q}$. In the queue we may label independently each packet in class i, $i = 1, 2$, with probability $\bar{\lambda}_i(1-p_i(\bar{s}_i))/(\bar{\lambda}_1(1-p_1(\bar{s}_1))+\bar{\lambda}_2(1-p_2(\bar{s}_2)))$, where we have taken into account that the RED drops p_1 and p_2 will thin the arrival processes. Hence the expected queue length in class i satisfies

$$\bar{q}_i = \frac{\bar{\lambda}_i(1-p_i(\bar{s}_i))}{\bar{\lambda}_1(1-p_1(\bar{s}_1))+\bar{\lambda}_2(1-p_2(\bar{s}_2))}\bar{q}.$$

The evolution of the total queue is given by

$$\frac{d}{dt}\bar{q} = (1-\pi_K(\bar{q})) \times \left(\bar{\lambda}_1(1-p_1(\bar{s}_1))+\bar{\lambda}_2(1-p_2(\bar{s}_2))\right) - \mu\left(1-\pi_0(\bar{q})\right).$$

Finally, the TCP population is described by

$$\frac{d}{dt}\bar{\lambda}_1 = -\frac{P_{L,1}}{2m}\bar{\lambda}^2 + (1-P_{L,1})\frac{m}{R^2},$$

where $P_{L,1} = p_1(\bar{s}_1) + \pi_K(\bar{q}) - p_1(\bar{s}_1)\pi_K(\bar{q})$. (Note that there is no ODE for UDP traffic.)

This model views classes 1 and 2 uncoupled in the sense that the random packet drop in a class is based on the EWA queue length of that class only. The RIO algorithm [15], in which class 1 has priority and class 2 is admitted into the queue based on the exponentially averaged total queue length, is obtained easily by replacing $p_2(\bar{s}_2)$ with $p_2(\bar{s}_1+\bar{s}_2)$ in above.

5.2 Approximating variable round trip times

In our model the simplest way to approximate a TCP population with variable round trip times is to partition the total TCP population into n sets so that in each set i, m_i TCPs have the same round trip time R_i. Each subpopulation i can be modeled and treated as TCP class i in the previous generalization. No priority structure between the classes is introduced if the classes are uncoupled and the RED parameters are the same in each class. This modeling paradigm is scalable; even a large number of differential equations can be solved in a reasonable time.

6 Model validation

Here we illustrate the accuracy of our differential equations by comparing against simulated results and we use the model to explore equilibrium surfaces as functions of the system parameters. Simulation is used to validate the accuracy of the model, i.e., that the stochastic assumptions are not critical. In the simulation none of the stochastic approximations have been used. Also, each TCP source is a periodic source with a time varying sending rate reacting independently to successful packet transmissions or random packet drops according to the congestion avoidance rate control rules. The only assumptions that have been retained in our simulations are the omission of the counter, that the RTT is a fixed constant, and that TCP congestion control works in the congestion avoidance phase.

6.1 Simulation implementation

To be specific, our simulation has been implemented in the following manner. The sources model the behavior of a greedy FTP source population transmitting constant length packets. Hence, each TCP source i has its own congestion window w_i (as opposed to our model), and R is a common constant for all sources. The idea is to describe the source as a fluid process with instantaneous rate w_i/R, and send a packet at those instants of time where the cumulative fluid attains an integer value. Thus each source has the following behavior i) it tries to send w_i packets during R and ii) it tries to space them in a "smooth" manner, i.e., the packets are

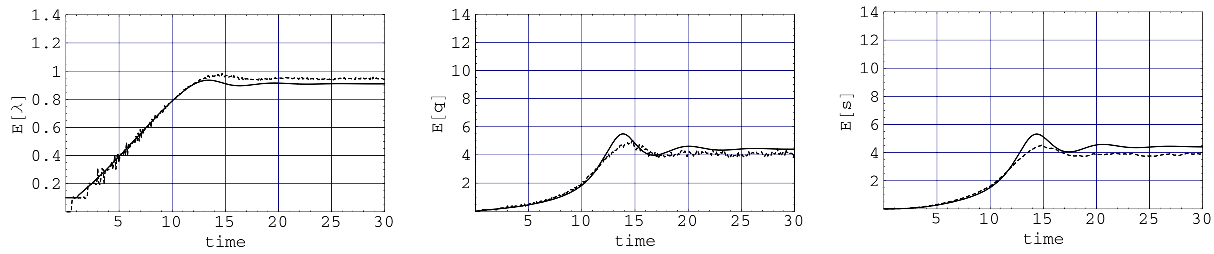

Figure 2.: Evolution of $\bar{\lambda}(t), \bar{q}(t)$ and $\bar{s}(t)$ (unit of time is 1000 on the x-axis).

not sent in one burst, as may happen with the real TCP. Each sent packet arrives into the queue after a delay of $R/2$. Upon arrival at the queue, an acknowledgement is sent back to the source whether the packet was accepted or discarded (either by RED control or buffer overflow). The acknowledgement reaches the source after a delay of $R/2$. The source either increases its congestion window by $1/w_i$, i.e., sets $w_i \leftarrow w_i + 1/w_i$, if the packet was accepted or halves the congestion window, i.e., sets $w_i \leftarrow w_i/2$. Additionally, to avoid the sources from being synchronized right from the start of the simulation, each source is initialized with a window size of 1 but the sources are "turned on", i.e., transmit their first packets, at a time which is uniformly distributed in the range $[0, R]$. Thus, the aggregate traffic into the queue consists of a superposition of periodic sources with a time varying sending rate.

In the numerical examples the following system parameters will remain constant. The buffer size $K = 75$, the service time $1/\mu = 1$, R=1000 and $\beta = 0.002$ (as is suggested in the literature). Also, the unit of time is taken to be $1/\mu$. Choosing R to be 1000 times the average packet transmission time corresponds to a real life system where, assuming the packet size to be 500 bytes and the link speed to be 155 Mbit/s, R is $(1000 \cdot 8 \cdot 500\,\text{bit})/155\,\text{Mbit/s} \approx 26\,\text{ms}$. Additionally, in the analytical model we start the system with initial data $\bar{q}(t) = \bar{s}(t) = 0$ for $0 \leq t \leq R/2$ and $\bar{\lambda}(t) = m/R$ for $0 \leq t \leq R$ which corresponds to setting $q_0 = s_0 = 0$ and $w_i = 1$ for each source in the simulation. The simulated results have been obtained by averaging over 100 independent simulation runs.

6.2 TCP examples

Here we illustrate the accuracy of our model with two examples. The examples have been chosen to demonstrate a case where the queue reaches equilibrium in a nice manner and a case where severe oscillations are encountered. In both examples we fix the number of TCP sources $m = 100$ and $T_{\max} = K/2 = 37.5$. (Note that a larger TCP population is expected to improve the accuracy of the analytical model.) First we look at an example where we set the other RED parameters to $T_{\min} = 0$ and $p_{\max} = 0.2$. The results are shown in Figure 2. for the time evolution of $\bar{\lambda}(t)$ (left figure), for $\bar{q}(t)$ (middle figure) and for $\bar{s}(t)$ (right figure). In all the figures in this section, the solid lines show the results obtained from the RFDE system (3) and the dashed lines correspond to simulated results. From the figures we can see that the results from the RFDE system match well with the simulated results and the equilibrium values are quite close. This also verifies, among other stochastic approximations, that our assumption in the model on the locally Poisson nature of the aggregate traffic is applicable and gives accurate results, even when the actual traffic consists of a superposition of non-Poisson sources.

Changing $T_{\min} = 5$ and increasing $p_{\max} = 0.5$ results in strong oscillations (see Figure 3.). Even in this case, we can observe rather good agreement between the analytical and the simulated results. The only discrepancy is that apparently the simulated system contains elements (phase mixing) that dampen the oscillations and prolong the oscillation periods, which does not take place in the solutions of the RFDE system.

As a continuation of the work reported here, we have derived and submitted for publication

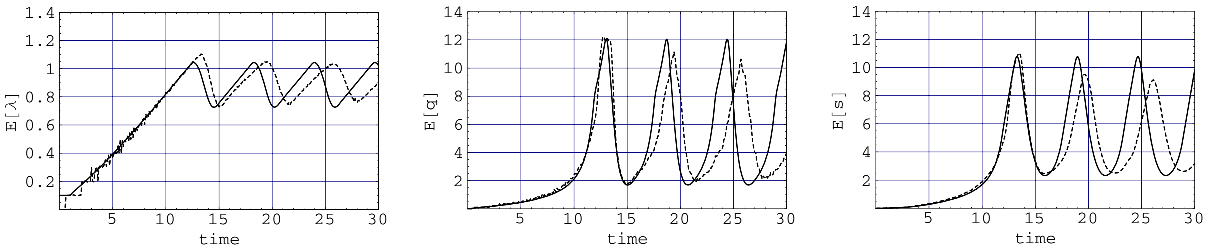

Figure 3.: Evolution of $\bar{\lambda}(t), \bar{q}(t)$ and $\bar{s}(t)$ (unit of time is 1000 on the x-axis).

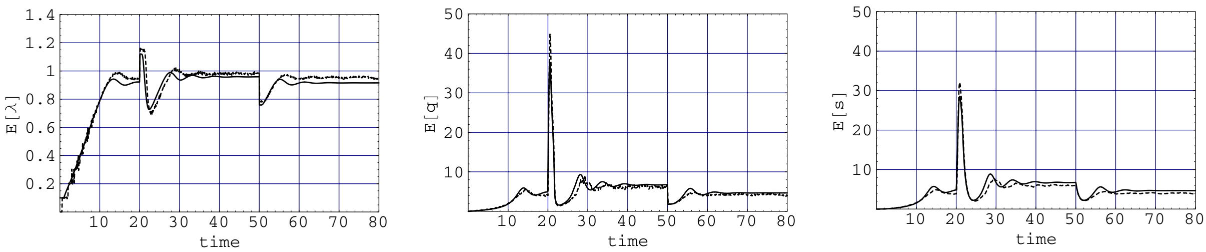

Figure 4.: Evolution of $\bar{\lambda}(t), \bar{q}(t)$ and $\bar{s}(t)$ (unit of time is 1000 on the x-axis).

(see Kuusela et al. [3]) the stability analysis of the system giving, not only sufficient, but also necessary conditions for the asymptotic stability. Note that the stability analysis done in Hollot et al. [16] as a continuation of [5], uses a different technique, and provides only sufficient conditions for their system. For the examples shown here our stability analysis verifies the visual observation of the first example being stable and the second one unstable.

6.3 Uncontrolled UDP traffic and the TCP steady state

Next we explore the effect of uncontrolled UDP traffic on the steady state of the system. In this case, the UDP traffic is here modeled as a pure Poisson source with arrival rate $\lambda = 0.2$. The rest of the system parameters are: $m = 100$ (same as earlier), $p_{\max} = 0.2$, $T_{\min} = 0$ and $T_{\max} = 40$. The system was started from an empty state, and at time 20 000 the UDP source was turned on and shut off at time 50 000. The results are shown in Figure 4. for $\bar{\lambda}(t)$ (left figure), for $\bar{q}(t)$ (middle figure), and for $\bar{s}(t)$ (right figure). Here $\bar{\lambda}(t)$ is the aggregate arrival rate from the TCP population and the UDP traffic. Again we can observe a good correspondence and the results show how the system first reaches an equilibrium, how it is affected by the UDP pulse and how the equilibrium is reached after the pulse disappears.

6.4 Equilibrium surfaces

Here we plot some equilibrium surfaces of the system (3) as functions of the system parameters. We consider a buffer of size 40 with RED parameters $T_{\min} = 10$ and $T_{\max} = 30$. The RED drop parameter $p_{\max}$ is taken as a free parameter that varies in $[0.01, 0.35]$. The equilibrium depends on the TCP population parameters only through TCPpar $= m/R$, which is allowed to vary in $[0.05, 0.7]$. Figure 5. (left figure) illustrates the queue size at the equilibrium as a function of $p_{\max}$ and TCPpar. We point out that if the RED drop probability is small but the TCP population is aggressive (either m is large or R is small) the system (3) does not have an equilibrium as the queue tends to fill up to $T_{\max}$ and then oscillates at values slightly below $T_{\max}$. In addition, even if an equilibrium exists it is not necessarily a stable one (see [3]). The surface of the expected goodput of the TCP population is illustrated in Figure 5. (right figure). Note that the surfaces clearly illustrate the problematic TCP feature (from the system designers point of view) that the system performance depends on the number of TCP connections, which

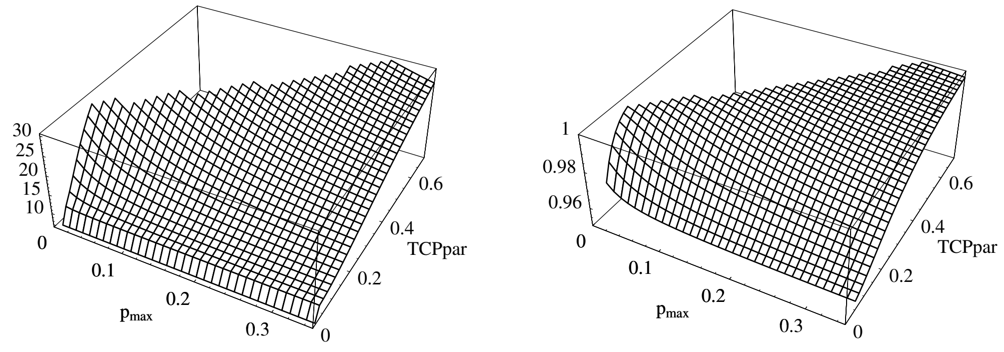

Figure 5.: The queue size at equilibrium (left figure) and the goodput of the TCP population at equilibrium (right figure).

is also discussed in [17].

If one wishes to set a fixed target queue length $\bar{q}^t \in [T_{\min}, T_{\max}]$ (which also coincides with the exponentially averaged queue length) to be achieved at the equilibrium, the relationship

$$\frac{m}{R} = \frac{\sqrt{P_L}\left(1 - \pi_0(\bar{q}^t)\right)}{\sqrt{2(1-P_L)}\,(1 - \pi_K(\bar{q}^t))\,(1 - p(\bar{q}^t))},$$

where $P_L = p(\bar{q}^t) + \pi_K(\bar{q}^t) - p(\bar{q}^t)\pi_K(\bar{q}^t)$, can be used to explore, e.g., how $p_{\max}$ has to be adjusted to compensate the changes in m/R. The above relationship has been derived from (3) at the equilibrium state.

7 Conclusions

We have developed a dynamical model to describe the evolution of the TCP aggregate and the RED controlled queue. The TCP sources were idealized to operate only in the congestion avoidance phase. It is worth mentioning that this approach can be easily applied to other additive-increase-multiplicative-decrease schemes by changing the differential equation for the throughput λ accordingly. In addition, the model of source and queue interaction can be used to explore, for example, various parameter settings or stability surfaces. Moreover, this approach can be easily extended to study the interaction of other traffic behaviors with RED, such as TCP variants, equation-based controls, rate adaptive traffic etc. Because the mathematical model is in terms of expected values of the source and queue variables, an equivalent simulation study would require averaging over a large number of replications, making it easily an infeasible approach.

In deriving the differential equation model several stochastic assumptions and approximations were necessary, in particular, the aggregate packet arrivals were approximated by an inhomogeneous Poisson process. We point out that our main interest is the congestion control aspect of the complete system, and partial justification to such approximation is the observation by Veres and Boda [18]: several statistical tests indicate that the aggregate TCP traffic entering through a bottleneck buffer is short-range dependent. Moreover, our careful simulation verification, in which neither TCP window nor stochastic assumptions were used, showed that the approximations were well justified.

The aim of this paper was to carefully derive and then validate the model. Additionally, in [3] we have successfully performed the stability analysis of the system presented here, deriving not only sufficient, but also necessary conditions for the system to be asymptotically stable. This follow-up paper contains several illustrative figures that show how the system parameters affect the system performance.

References

[1] S. Floyd and V. Jacobson, "Random Early Detection gateways in congestion avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 3, pp. 397–413, 1993.

[2] P. Lassila and J. Virtamo, "Modeling the dynamics of the RED algorithm," in *Proceedings of QofIS'00*, September 2000, pp. 28–42.

[3] P. Kuusela, P. Lassila, and J. Virtamo, "Stability of TCP-RED congestion control," submitted for publication, available at `http://www.tct.hut.fi/tutkimus/com2/publ/`, 2001.

[4] F. Kelly, "Mathematical modelling of the Internet," available at `www.statslab.cam.ac.uk./~frank/` (also an extended version in Mathematics Unlimited – 2001 and Beyond, Springer Verlag, 2000), 1999.

[5] V. Misra, W. Gong, and D. Towsley, "A fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED," in *Proceedings of ACM SIGCOMM*, August 2000, pp. 151–160.

[6] R. Laalaoua, S. Jędruś, T. Atmaca, and T. Czachórski, "Diffusion model of RED control mechanism," in *in Proceedings of International Conference on Networking*, July 11–13 2001, Colmar, France.

[7] E. Altman, K. Avrachenkov, and C. Barakat, "A stochastic model of TCP/IP with stationary random losses," in *Proceedings of ACM SIGCOMM*, 2000, pp. 231–242.

[8] Brown P., "Resource sharing of TCP connections with different round trip times," in *Proceedings of IEEE INFOCOM*, 2000, pp. 1734 – 1741.

[9] S. Floyd et al., "Discussions on setting parameters," `http://www.aciri.org/floyd/REDparameters.txt`, November 1997.

[10] J. Virtamo, "Numerical evaluation of the distribution of unfinished work in an M/D/1 system," *Electronics Letters*, vol. 31, no. 7, pp. 531–532, 1995.

[11] D. Gross and C. Harris, *Fundamentals of queueing theory*, Wiley, 3rd edition, 1998.

[12] Wang W., D. Tipper, and Banerjee S., "A simple approximation for modeling nonstationary queues," in *Proceedings of IEEE INFOCOM*, March 1996, pp. 255–262.

[13] H. C. Tijms, *Stochastic Models: An Algorithmic Approach*, Wiley, 1995.

[14] P. Kuusela and J. T. Virtamo, "Modeling RED with two traffic classes," in *Proceedings of NTS-15*, August 2000, pp. 271–282, available at `http://www.tct.hut.fi/tutkimus/com2/`.

[15] D. Clark and W. Fang, "Explicit allocation of best-effort packet delivery service," *IEEE/ACM Transactions of Networking*, vol. 6, no. 4, pp. 362–373, August 1998.

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

[17] T. J. Ott, T. V. Lakshman, and L. H. Wong, "SRED: Stabilized RED," in *Proceedings of IEEE INFOCOM*, March 1999.

[18] A. Veres and M. Boda, "The chaotic nature of TCP congestion control," in *Proceedings of IEEE INFOCOM*, March 2000, pp. 1715–1723.