Christos G. Cassandras and John Lygeros

Stochastic Hybrid Systems: Recent Developments and Research Trends

CRC PRESS
Boca Raton London New York Washington, D.C.

Contents

0-8493-0052-5/00/$0.00+$.50

Chapter 1

Stochastic Hybrid Modeling of On-Off TCP Flows

João Hespanha
University of California, Santa Barbara

Most of today's Internet traffic uses the Transmission Control Protocol (TCP) and it is therefore not surprising that modeling TCP's behavior has been attracting the attention of academia and industry ever since this protocol was proposed.

TCP is responsible for regulating the rate at which a source of data transmits packets, a task known as *congestion control*. TCP follows a greedy algorithm that constantly attempts to increase the rate at which data is transmitted. Eventually, the network is unable to carry the data and packets are dropped. When this is detected, the TCP source decreases its sending rate by roughly dividing it in half. However, shortly after that, the TCP source returns to its continuous attempt to increase the sending rate and the cycle repeats itself. During the first cycle (i.e., until the first drop) the rate is increased in an exponential fashion to rapidly reach an adequate value. This is called TCP's *slow-start* mode. However, in subsequent cycles, the increase is more gentle and roughly follows a linear law. This more gentle increase is crucial for stability (see, e.g., [6]). This second phase is called *congestion-avoidance*. The reader may want to consult a textbook, such as [33], for a more detailed description of TCP.

As noted by Bohacek et al. [7], hybrid systems provide a natural framework to model TCP because this protocol is characterized by different modes of operation with distinct dynamics. The need for stochastic hybrid systems (SHS) arises because many of the events that drive traffic models — such as packet drops and the

Manuscript date: August 30, 2006

0-8493-0052-5/00/$0.00+$.50

start/termination of transmissions — are well modeled by stochastic processes.

In this chapter, we construct a SHS model for TCP. Each realization of this model is meant to represent the traffic generated by a single user that initiates a TCP session, waits until the transfer terminates (the "transmission-on" period), spends some time "processing" the file received (the "transmission-off" period), and then initiates a new TCP session. This behavior is continuously repeated, but the transfer-sizes and the durations of the off-periods are selected from pre-specified distributions. The transfer-sizes implicitly determine the duration of the on-periods, taking into account TCP's instantaneous transfer rate. This type of on-off model was also considered by [2].

Since we do not restrict our attention to infinitely long TCP sessions (usually called long-lived flows), we need to explicitly model TCP's slow-start mode that dominates short transfers [23]. Moreover, we take into account the delay between the time instant at which a drop is detected and the corresponding reaction by TCP, typically one round-trip time later.

Using moment closure techniques inspired by Bohacek [5] and further discussed in [13], we build systems of ordinary differential equations (ODEs) that provide accurate approximations to the dynamics of the average sending rate as well as its higher order moments, including the standard deviation.

When our results are specialized to long-lived flows (always-on), the average sending rates are consistent with previously established models (at least for slowly varying drop-probabilities), which validates our modeling methodology.

The most surprising results are obtained for on-off flows. For a few transfer-size distributions reported in the literature, we show that the standard deviation is much larger than the average sending rate of individual TCP flows. Moreover, the packet drop probability has a surprisingly small effect on the average sending rate, but provides a strong control on its standard deviation. The explanation seems to be that, even with a heavy tail, the bulk of the data "slips-through" with very few drops. In practice, either all packets are sent during the slow-start mode or shortly after TCP enters the congestion-avoidance mode. The precise time at which the first drop occurs has a tremendous influence on the average sending rate of the flow, thus the very high standard deviation.

The fact that the "heaviness" of the tail seems to have little impact on Internet's performance was also pointed out by Liu et al. [17]. However, the fact that the packet drop probability exerts a much larger impact on the standard deviation of the sending rate than on its average has significant implications for congestion control. In particular, it seems to indicate that previously used long-lived flow models may not be suitable for the analysis and design of congestion control algorithms for on-off TCP flows. It also questions the validity of the "TCP-friendly" formula for aggregate on-off TCP flows.

The remainder of this chapter is organized as follows. In Section 1.1 we discuss our overall modeling approach to TCP in the context of alternative modeling techniques. In Section 1.2 we present the formal Stochastic Hybrid Systems (SHS) model for a single-user on-off TCP flow. A short introduction to the class of SHS considered in this chapter is provided in the Appendix for the interested reader. In

Section 1.3 we derive an infinite-dimensional system of ordinary differential equations that describe the dynamics of the statistical moments of TCP's sending rate. In Section 1.4 we show how this model can be truncated to obtain approximate models that are amenable to the investigation of TCP's behavior. We examine the properties of the model obtained for long-lived flows (always-on) as well as for on-off flows with two realistic transfer-size distributions.

1.1 Related work

There is an extensive literature on models that describe the properties of TCP congestion control. We start by presenting several widely used models for long-lived TCP flows (corresponding to infinitely long transfers). We then discuss a few more recent models for finite TCP flows.

1.1.1 Models for long-lived flows

A great deal of effort has been placed in characterizing the steady-state behavior of long-lived TCP flows [20, 22, 26–28, 31]. In particular, in studying the relation between the average transmission rate μ, the average round-trip time RTT, and the per-packet drop probability p_{drp}. The so called "TCP-friendly" formula

$$\mu = \frac{c}{RTT\sqrt{p_{\text{drp}}}} \tag{1.1}$$

has been derived by several authors, with small variations on the value of the constant c: Ott et al. [26] obtained $c = 1.310$; Mahdavi and Floyd [20] and Mathis et al. [22] obtained $c = 1.225$; and Bohacek et al. [6] obtained $c = 1.270$ for small values of p_{drp}. This formula reflects the fact that drops are the sole mechanism that keeps TCP's transmission rate bounded, when there is an infinite amount of data to transmit. Moreover, it specifies that as the drop probability goes to zero, the transmission rate should grow to infinity inversely proportional to $\sqrt{p_{\text{drp}}}$.

In general, the receiver acknowledges the arrival of each data packet by sending a short ACK packet back to the source. However, the protocol allows for the transmission of acknowledgments to be delayed, so that a single ACK packet can acknowledge the arrival of multiple data packets. Padhye et al. [27, 28] considered the general case of delayed acknowledgments and obtained $c = 1.225/\sqrt{n_{\text{ack}}}$ for (1.1), where n_{ack} denotes the number of acknowledgments per ACK packet. Typically $n_{\text{ack}} = 2$ when there are delayed acknowledgments and $n_{\text{ack}} = 1$ in their absence.

The primary mechanism for detection of drops is through the *triple duplicate ACKs mechanism*. In essence, a source declares a data packet dropped if it did not yet receive an acknowledgment for that packet, but it already received ACK packets for three data packets that were subsequently transmitted. However, since an ACK packet is only generated when a data packet successfully reaches the destination,

this mechanism for drop detection fails when not enough data packets reach the destination (after a packet is dropped). A timeout mechanism is used to recover from this situation. Padhye et al. [27, 28] further improved (1.1) by considering the effect of timeouts due to insufficient ACKs to detect a drop through the triple duplicate ACKs mechanism. They also took into account receiver-imposed limitations on the congestion window size. This led to

$$\mu = \min\left\{\frac{W_{\max}}{RTT}, \Big(RTT\sqrt{\frac{2\,n_{\text{ack}}\,p_{\text{drp}}}{3}} + T_0\,p_{\text{drp}}(1+32p_{\text{drp}}^2)\min\{1,\sqrt{6\,n_{\text{ack}}\,p_{\text{drp}}}\}\Big)^{-1}\right\}, \quad (1.2)$$

where $W_{\max}$ denotes the maximum congestion window size and T_0 the period during which no packets are sent following a timeout (typically determined experimentally). This formula proved reasonably accurate even when a significant portion of drops are detected through timeouts, for which (1.1) is not. Sikdar et al. [31] derived an alternative formula to (1.2), which proved equally accurate. However, both derivations assume that losses within a window are strongly correlated. In particular that when one drop occurs all subsequent packets in the same window are also dropped. This assumption is reasonable for drop-tail queuing, but not for active queuing policies such as Random Early Detection (RED) [10].

The derivations of (1.1) and (1.2) in the papers mentioned above analyze the evolution of the congestion window size between consecutive drop events for a single flow. Therefore, μ should be understood as a *time-average for a single TCP flow*. This type of approach was pursued in [15, 16, 18, 19, 27, 28, 30] to derive dynamic models for the congestion-avoidance stage of long-lived TCP flows. Kunniyur and Srikant [15] and Lakshmikantha et al. [16] proposed the continuous-time model

$$\dot{\mu}(t) = \frac{1}{RTT^2} - \frac{2}{3}p_{\text{drp}}(t-RTT)\mu(t)\mu(t-RTT) \quad (1.3)$$

where $\mu(t)$ denotes an "instantaneous" average sending rate, RTT the window size, and $p_{\text{drp}}(t)$ the per-packet drop probability; whereas Low et al. [19] proposed the continuous-time model

$$\dot{\omega}(t) = \frac{\big(1-p_{\text{drp}}(t-\tau)\big)\omega(t-\tau)}{RTT(t-\tau)\omega(t)} - \frac{1}{2}\frac{p_{\text{drp}}(t-\tau)\omega(t)\omega(t-\tau)}{RTT(t-\tau)}, \quad \mu(t) = \frac{\omega(t)}{RTT(t)},$$

where ω denotes the "instantaneous" average congestion window size, and τ a constant "equilibrium" round-trip time. Shakkottai and Srikant [30] proposed the discrete-time model

$$\mu(t+1) = \mu(t) + \frac{1}{RTT^2} - \frac{2}{3}p_{\text{drp}}(t-RTT)\mu(t-RTT)\big(\mu(t-RTT)+a\big) \quad (1.4)$$

where μ denotes the average congestion window size, RTT the average window size (in discrete-time units), and a the average of the "uncontrolled" competing flow;

whereas Low [18] proposed the discrete-time model

$$\mu(t+1) = \mu(t) + \frac{1 - p_{\text{drp}}(t)}{RTT^2} - \frac{2}{3} p_{\text{drp}}(t)\mu(t)^2 \tag{1.5}$$

In these discrete-time models, the time units must be sufficiently large for the time average to be meaningful.

In the models (1.3)–(1.5) the state should be understood as a time averaged quantity over an interval of time for which the fluid approximation is meaningful. In particular, for a "drop-rate" to be meaningful the averaging period T_{ave} must be sufficiently large to include several drops. Using (1.1), we can compute the (steady-state) average number of drops that occur in T_{ave}, which should be much larger than one. Since the average number of drops in an interval of length T_{ave} is given by the number of packets $\mu\, T_{\text{ave}}$ sent in this period times the drop probability p_{drp}, we conclude that we must have

$$p_{\text{drp}}\, \mu\, T_{\text{ave}} = \frac{c\sqrt{d}T_{\text{ave}}}{RTT} \gg 1 \;\Leftrightarrow\; T_{\text{ave}} \gg \frac{RTT}{c\sqrt{p_{\text{drp}}}}\ (\gg RTT).$$

One should therefore not expect these models to be valid over time scales of the order of $\frac{RTT}{c\sqrt{p_{\text{drp}}}}$ or smaller. It turns out that if one linearizes[1] (1.3) around its equilibrium point, one obtains a stable system with a time constant of $\frac{1.633RTT}{\sqrt{p_{\text{drp}}}}$, which is never significantly larger than $\frac{RTT}{c\sqrt{p_{\text{drp}}}}$. Similar conclusions can be obtained for the other models discussed so far. At least from a theoretical perspective, this seems to compromise the validity of these single-flow models. However, we will see shortly that these models can be re-interpreted as multi-flow ensemble models, for which these concerns do not necessarily arise.

When one wants to examine the dynamics of TCP flows for time scales on the order of only a few round-trip times, the time averaging period must be shortened. Any fluid approximation must use an averaging period no smaller than one round-trip time because otherwise this would violate approximating the sending rate by the congestion window size divided by the round-trip time. Bohacek et al. [7] proposed a modeling framework where quantities are only averaged over time periods of roughly one round-trip time. Drops were kept as discrete-events and were modeled explicitly, leading to a *hybrid control system.* The models developed were shown to be very accurate, even looking at time-traces of individual flows. The models developed for TCP flows correspond to the transfer of finitely many packets, capturing slow-start, congestion-avoidance, fast-recovery, and timeouts.

Misra et al. [24, 25] took a different approach to avoid averaging over long time intervals. They still consider averaging over time periods of roughly one round-trip time, but utilize *ensemble averages* for drop events. Using Itô Calculus they derive the following approximate model for the congestion-avoidance stage of long-lived

[1] For simplicity, we ignore the delays which are much smaller than T_{ave} as well as the dependence of RTT on μ.

TCP flows:

$$\dot{\omega}(t) = \frac{1}{RTT(t)} - \frac{\omega(t)\,p_{\mathrm{drp}}\big(t-RTT(t)\big)\,\omega\big(t-RTT(t)\big)}{2RTT\big(t-RTT(t)\big)}, \quad \mu(t) = \frac{\omega(t)}{RTT(t)}, \tag{1.6}$$

where ω should be understood as the *ensemble average* of the congestion window size and p_{drp} the per-packet drop probability (also in an ensemble sense). Interestingly, aside from the different interpretation of the quantities involved, the ensemble-average model (1.6) resembles very much the time-average models (1.3)–(1.5). This model was refined to include the effect of slow-start and timeouts, leading to

$$\dot{\omega}(t) = \frac{p_{\mathrm{ss}}\omega(t) + (1-p_{\mathrm{ss}})}{RTT(t)} - \Big(\frac{\omega(t)}{2}(1-p_{\mathrm{to}}) + (\omega(t)-1)p_{\mathrm{to}}\Big) \frac{p_{\mathrm{drp}}\big(t-RTT(t)\big)\,\omega\big(t-RTT(t)\big)}{RTT\big(t-RTT(t)\big)}, \tag{1.7}$$

where p_{ss} denotes the probability that a flow is in slow-start mode and p_{to} the probability that a drop leads to timeout. However, (1.7) still ignores the zeroing of the sending rate during the timeout period and does not provide values for the probability p_{ss}. It is suggested that p_{to} can be approximated by $\min\{1,3/\omega\}$ based on an argument by Padhye et al. [28] that assumes that losses within a window are strongly correlated. As mentioned above, this assumption seems reasonable for drop-tail queuing but probably not for RED.

Shakkottai and Srikant [30] also used stochastic aggregation to reduce the time-scales over which a model is valid. In particular, they showed that the aggregation of n discrete-time models like (1.3) can described by

$$\dot{\mu}_n(t) \;=\; \frac{1}{RTT^2} \;-\; \frac{2}{3}p_{\mathrm{drp}}(t-RTT)\mu_n(t-RTT)\big(\mu_n(t-RTT)+a\big), \tag{1.8}$$

where μ_n denotes the average congestion window size with respect to the ensemble of n flows and the parameter a models the effect of competing uncontrolled background traffic. They showed that for large n this model is valid over time scales n times smaller than those of the original discrete-time model (1.4), which was only valid for time scales much larger than one round-trip time.

A key feature of TCP's behavior is the existence of a delay between the occurrence of a drop and its detection and eventual reaction by TCP. This delay has been identified as one of the causes for queue-length instabilities (cf., e.g., the survey [19]). The models (1.3)–(1.8) attempt to capture this by introducing delays in all the right-hand-side terms related to the detection of drops. However, since delayed differential equations are difficult to analyze, these models are usually simplified by solely considering a delay in the term p_{drp}, which turns these equation back into (time-varying) ordinary differential equations.

We pursue here a stochastic version of the hybrid models proposed in [7]. As in [24, 25] time averaging is done over intervals of roughly one round-trip time to obtain continuously varying sending rates, and we then investigate the dynamics of

ensemble averages. However, we do not just consider the evolution of averages as in [24, 25, 30]. Instead, we also study the dynamics of high-order statistical moments. Our models explicitly considers the delay between drop occurrence and detection by TCP and take this into account in the ensemble averaging.

1.1.2 Models for on-off flows

Analytical studies of finite TCP flows have been pursued in relatively few papers. Mellia et al. [23] proposed a stochastic model for short-lived flows that predicts the flow's completion time as a function of the transfer-size, drop probability, and round-trip time. This model ignores congestion-avoidance altogether and is therefore only valid for very short flows, for which most data is sent during the slow-start mode. Zheng et al. [34] proposed an improved model to predict a flow's completion time that considers congestion-avoidance and is therefore applicable both to short and long-lived flows. However, neither [23] nor [34] provide explicit dynamic models for TCP traffic, such as the ones described before for long-lived flows.

In [2], Baccelli and Hong extended their models for long-lived TCP flows proposed in [3] to on-off flows. For each source, they extract a sequence of random independent and identically distributed (i.i.d) transfer-sizes and a sequence of random i.i.d. "think times." Each source then alternates between on-periods (during which data is transmitted) and off-periods (the think times). The duration of the on-periods is determined by the transfer-sizes and the corresponding throughput. We will borrow this type of on-off behavior for our SHS model of TCP. Baccelli and Hong [2] stop short of analyzing the resulting stochastic model and simply use it to produce Monte Carlo simulations that run much faster than packet-level simulators. The computational gains are achieved by considering fluid models and by foregoing the identity of individual packets, very much as discussed in the previous two chapters.

Marsan et al. [21] proposed a fairly sophisticated model for aggregate TCP flows of several classes, each with different routes through the network. The state of their model consists of functions $P^i_{ss}(w,t,\ell)$ that keep track of the number of flows of class i in slow-start mode at time t, with congestion window size no larger than w and remaining transfer-size no larger than ℓ. Similarly, functions $P^i_{ca}(w,t,\ell)$ keep track of the number of flows in the congestion-avoidance mode. The evolution of all the $P_{ss}(\cdot)$ and $P_{ca}(\cdot)$ are governed by a system of Partial Differential Equations (PDEs).[2] Their model can also capture fast-recovery, the maximum window size of the TCP sources, and both RED and drop-tail queues. This is achieved by introducing additional states, which keep track of how many flows are in fast recovery, how many flows have reached the maximum window size, etc. This modeling framework is very powerful but since there is no explicit characterization of individual flows, it is not possible to build the type of on-off models proposed by Baccelli and Hong [2]. Instead, the

[2]For exponentially distributed transfer-sizes, the ℓ dependence can be dropped and the resulting PDEs are one-dimensional.

model takes as inputs the rates at which new TCP flows start and finish. To emulate the type of on-off behavior in [2], Marsan et al. [21] propose to use heuristic rules to adjust these rates as a function of the loss probability and other model parameters. The main shortcoming of the models proposed by Marsan et al. [21] is an infinite-dimensional state-space that makes it challenging to use these models to gain insight into TCP's behavior or to analyze the stability of congestion control algorithms.

The models proposed in this chapter consider ensembles of single-user on-off TCP flows, similar to the ones proposed by Baccelli and Hong [2]. The off-periods are assumed exponentially distributed whereas the on-periods are determined by the amount of data being transfered. We take as given the probability distribution of the transfer-sizes and this implicitly determines the distribution of the on-periods, by taking into account the (time-varying) sending rate, which is implicitly determined by the drop probability. This allows us to obtain directly from the model parameters such as the probability p_{ss} in (1.7) of a flow being in slow-start mode.

A key difference between our work and [21] is that we use our SHS model to construct a *system of ODEs* that describes the evolution of the mean and higher order moments for TCP's sending rate, whereas Marsan et al. [21] construct a *system of PDEs* to describe what essentially amounts to the whole distribution of the congestion window size. Although this distribution could also be computed for our SHS model [12], we opted to work with a more parsimonious state-space that only describes first and second order moments. Even though the resulting models are more complex than the ones discussed in Section 1.1.1 for long-lived flows, the closed-form computation of steady-state throughput and a stability analysis is still tractable.

As mentioned above, we combine the hybrid modeling framework for network modeling introduced by [7] with the stochastic drop models used by [24, 25] for RED. Both these models have been validated and do not seem too controversial. The third crucial element to an on-off TCP model is the distribution of the transfer-sizes, which is a much more controversial issue. Some studies seem to indicate that these distributions are heavy-tailed (cf., the survey [29]) but others conclude that there is little evidence to support such claim [8, 17]. Settling this issue is beyond the scope of this chapter so we opted to present results for two distributions: One with and another without a significant tail. Both distributions approximate data found in the literature. One consists of a mixture of two exponentials that approximately models the file distribution observed in the UNIX file system [14]. The parameters chosen for the mixture of exponentials model fairly well the "waist" of the distribution but somewhat underestimate its tail. The second distribution consists of a mixture of three exponentials and approximates the data reported by Arlitt et al. [1] obtained from monitoring transfers from a world-wide web proxy within an Internet Service Provider. This approximation captures fairly well the tail of the distribution (at least up to 100 MB, for which data is available). The idea of approximating heavy tail distributions for transfer-sizes using a mixture of exponentials was proposed by Feldmann and Whitt [9]. Like us, Marsan et al. [21] also used a mixture of exponentials to model TCP transfer-sizes.

1.2 A stochastic model for TCP

In this section we present a SHS model for single-user on-off TCP flows. Many-user models can be obtained by aggregating several of these single-user models. Our model is based on the hybrid modeling framework proposed by Bohacek et al. [7] and the stochastic models by Misra et al. [24, 25].

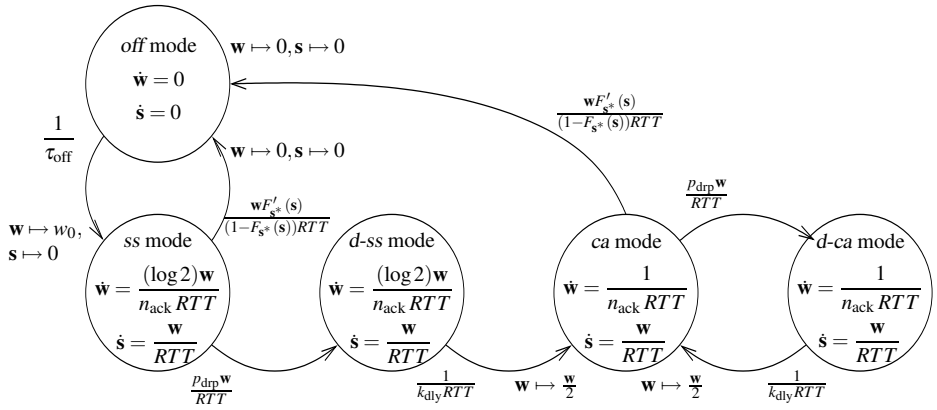


FIGURE 1.1: Stochastic hybrid model for a single-user TCP flow.

The SHS that we use to model a single-user on-off TCP flow is represented graphically in Figure 1.1. This model has five modes: *off* which corresponds to flow inactivity; *ss* which corresponds to slow-start; *ca* which corresponds to congestion-avoidance; and two other modes (*d-ss* and *d-ca*) that will be explained shortly. Each mode corresponds to a node in the graph in Figure 1.1.

As time progresses, the model transitions from one mode to another according to specific rules that will be discussed below. While on each mode, two continuous variables evolve according to the differential equations displayed inside the corresponding node. These variables are TCP's congestion window size $\mathbf{w}$ and the cumulative number of packets $\mathbf{s}$ sent so far in a particular connection. The evolution of these variables obeys the following rules:

1. During the *off* mode the flow is inactive and we simply have $\mathbf{w} = \mathbf{s} = 0$.

2. The *ss* mode corresponds to TCP's slow-start. In this mode, $\mathbf{w}$ packets are sent each round-trip time RTT and the congestion window size $\mathbf{w}$ increases by one for each ACK packet received. Until a drop occurs this means that the rate at which packets are sent is equal to $\mathbf{r} = \frac{\mathbf{w}}{RTT}$ and the number of ACK packets received is equal to $\frac{\mathbf{r}}{n_{\text{ack}}}$, where n_{ack} denotes the number of data packets

acknowledged per each ACK packet received. This can be modeled by

$$\dot{\mathbf{w}} = \frac{(\log 2)\mathbf{r}}{n_{\text{ack}}} = \frac{(\log 2)\mathbf{w}}{n_{\text{ack}} RTT}, \qquad \dot{\mathbf{s}} = \mathbf{r} = \frac{\mathbf{w}}{RTT}. \tag{1.9}$$

The $(\log 2)$ factor compensates for the error introduced by approximating the discrete increments by a continuous increase.

Without delayed ACKs ($n_{\text{ack}} = 1$), it is straightforward to check that if the round-trip time RTT were approximately constant this model would lead to the usual doubling of $\mathbf{w}$ every RTT. Typical implementations of TCP that use delayed ACKs set $n_{\text{ack}} = 2$. In this case, (1.9) leads to a multiplication of $\mathbf{w}$ by $\sqrt{2}$ every RTT. This is consistent with the analysis by Sikdar et al. [32], which shows that for $n_{\text{ack}} = 2$ the number of packets sent in the nth round-trip time of slow-start is approximately equal to $(1 + \frac{\sqrt{2}}{2})\sqrt{2}^n$. This formula is matched exactly by the fluid model in (1.9) when one sets $\mathbf{w} = 1.428$ in the beginning of *ss*. On the other hand, for $n_{\text{ack}} = 1$, the number of packets sent in the nth round-trip time of slow-start should be equal to 2^{n-1}. This matches the fluid model by making $\mathbf{w} = .693$ at the beginning of *ss*.

3. The *ca* mode corresponds to TCP's congestion-avoidance. In this mode $\mathbf{w}$ increases by $1/\mathbf{w}$ for each ACK packet received and, as in slow-start, $\mathbf{w}$ packets are sent each round-trip time RTT. This can be modeled by

$$\dot{\mathbf{w}} = \frac{1}{\mathbf{w}} \frac{\mathbf{r}}{n_{\text{ack}}} = \frac{1}{n_{\text{ack}} RTT}, \qquad \dot{\mathbf{s}} = \mathbf{r} = \frac{\mathbf{w}}{RTT}. \tag{1.10}$$

In practice, both in (1.9) and in (1.10), the round-trip time RTT is generally a time-varying quantity.

When a drop occurs, this event will not be immediately detected by TCP. To account for the delay, we consider two additional "delay" modes: *d-ss* and *d-ca*. When a drop occurs while the system is in *ss* or in *ca* it immediately transitions to *d-ss* or to *d-ca*, respectively. However, the dynamics of $\mathbf{w}$ and $\mathbf{r}$ remain unchanged. Only after a time period of roughly one round-trip time, TCP will react to the drop and adjust its congestion window size $\mathbf{w}$ and the sending rate $\mathbf{r}$.

The transitions between modes, which are represented by arrows in the graph in Figure 1.1, are stochastic events that occur at specific "rates." Informally, the rate at which a transition occurs corresponds to the expected number of times that transition will take place in a unit of time (cf. the Appendix and [11]). In Figure 1.1, these rates are shown as labels next to the start of the arrows. Some transitions trigger instantaneous changes (resets) of $\mathbf{w}$ and/or $\mathbf{s}$. These are represented near the end of the arrow. The following events are associated with transitions:

1. Drop occurrences correspond to transitions from the *ss* and *ca* modes to the *d-ss* and *d-ca* modes. These events occur at a rate given by $p_{\text{drp}}\mathbf{r}$, where p_{drp} denotes the per-packet drop probability and $\mathbf{r} := \frac{\mathbf{w}}{RTT}$ the packet sending rate. The drop rate p_{drp} will generally be time-varying.

2. Drop detections correspond to transitions from the *d-ss* and *d-ca* modes to the *ca* mode. These events occur at a rate given by $\frac{1}{k_{\text{dly}}RTT}$, which is consistent with an exponentially-distributed average delay of $k_{\text{dly}}RTT$. Typically $k_{\text{dly}} = 1$. The detection of a drop leads to a division of the congestion window size $\mathbf{w}$ by two.

3. The start of new flows corresponds to transitions from the *off* to the *ss* mode. These events occur at a rate given by $\frac{1}{\tau_{\text{off}}}$, which is consistent with an exponentially distributed duration of the off-periods with average τ_{off}. At the start of each new flow the congestion window size $\mathbf{w}$ is set equal to

$$w_0 := \begin{cases} .693 & n_{\text{ack}} = 1 \\ 1.428 & n_{\text{ack}} = 2. \end{cases}$$

   and the number of packets sent $\mathbf{s}$ is reset to zero.

4. The termination of flows correspond to transitions from the *ss* and *ca* modes to the *off* mode. These events occur at a rate given by

$$\frac{\mathbf{r}F'_{\mathbf{s}^*}(\mathbf{s})}{1-F_{\mathbf{s}^*}(\mathbf{s})}, \qquad \mathbf{r} := \frac{\mathbf{w}}{RTT}, \tag{1.11}$$

   which is consistent with a distribution $F_{\mathbf{s}^*} : [0,\infty) \to [0,1]$ for the number $\mathbf{s}^*$ of packets sent in each TCP session (cf. Appendix). When a flow terminates both $\mathbf{w}$ and $\mathbf{s}$ are reset to zero. The rate in (1.11) is usually called the *hazard rate* of the transfer-size distribution $F_{\mathbf{s}^*}$.

Two main simplifications were considered: We ignored fast-recovery after a drop is detected by three duplicate ACKs and we ignored timeouts. Fast-recovery takes relatively little time and has little impact on the overall throughput unless the number of drops is very high [6]. As mentioned before, timeouts typically occur when several consecutive packets are dropped, preventing the detection of drops by the triple duplicate ACKs mechanism. Therefore, timeouts have an especially severe impact on the throughput when drops are highly correlated. However, here we are mostly interested in RED for which high correlations are unlikely and therefore timeouts only occur under very high drop rates.
A few specific instances of the general model in Figure 1.1 are of interest.

Long-lived flows This model is obtained by assuming that the number of packets to transmit is infinitely large, i.e., that $F_{\mathbf{s}^*}(s) = 0$, $\forall s < \infty$. In this case, we can ignore the *off*, *ss*, and *d-ss* modes since they will only be active for a brief initial period and after this the SHS will continuously switch between the *ca* and *d-ca* modes.

Exponential transfer-sizes This model is obtained by assuming that the number of packets to transmit is exponentially distributed with average k, i.e., $F_{\mathbf{s}^*}(s) = 1 - e^{-\frac{s}{k}}$, $\forall s \geq 0$. In this case, the termination of flows occurs at a rate given by

$$\frac{\mathbf{w}F'_{\mathbf{s}^*}(\mathbf{s})}{(1-F_{\mathbf{s}^*}(\mathbf{s}))RTT} = \frac{k^{-1}\mathbf{w}}{RTT},$$

which is independent of the continuous state $\mathbf{s}$. For exponential transfer-sizes we can therefore ignore this state variable.

Pareto transfer-sizes It has been observed that modeling the distribution of transfer-sizes as an exponential is an over-simplification. For example, it has been argued that heavy-tail models are more fitting to experimental data (cf., e.g., [1, 4, 29]). Inspired by this work, we also consider a model for which the distribution of the number of packets to transmit is a shifted-Pareto[3] with shape parameter a and scale parameter b, i.e., $F_{\mathbf{s}^*}(s) = 1 - \left(\frac{b}{s+b}\right)^a$, $\forall s \geq 0$. Assuming that $a > 1$, this corresponds to an average number of packets equal to $\frac{ab}{a-1}$. The existence of high-order statistical moments depends on the value of a and the kth moment exists only for $k < a$. For this distribution, the termination of flows occurs at a rate given by

$$\frac{\mathbf{w} F'_{\mathbf{s}^*}(\mathbf{s})}{(1 - F_{\mathbf{s}^*}(\mathbf{s}))RTT} = \frac{a\,\mathbf{w}}{(\mathbf{s}+b)RTT}.$$

Mixed-exponential transfer-sizes An alternative to a Pareto distribution that turns out to be computationally more attractive and still fits well with experimental data is a mixture of exponentials [9]. According to this model, transfer-sizes are sampled from a family of M exponential random variables $\mathbf{s}_i$, $i \in \{1, 2, \ldots, M\}$ by selecting a sample from the ith random variable $\mathbf{s}_i$ with probability p_i. Each $\mathbf{s}_i$ corresponds to a distinct average transfer-size k_i.

To model this as a SHS, we consider M alternative

$$\{ss_i, d\text{-}ss_i, ca_i, d\text{-}ca_i : i = 1, 2, \ldots, M\}$$

modes, each corresponding to a specific exponential distribution for the transfer-sizes. The transitions from the inactive mode *off* to the slow-start mode ss_i corresponding to an average transfer-size of k_i occurs with probability p_i, which corresponds to a transition rate given by $\frac{p_i}{\tau_{\text{off}}}$. To obtain the desired distribution for the transfer-size, the transitions from ca_i and ss_i to the inactive mode *off* occur at a rate given by $\frac{k_i^{-1} w}{RTT}$. A similar technique could be used to obtain a mixture of exponentials for the distribution of the off-periods.

In this chapter, we will mostly focus our attention on long-lived flows and mixed-exponential transfer-sizes (with exponential as the special case $M = 1$). The resulting SHS have polynomial vector fields, reset maps, and transition intensities, which facilitates their analysis [13]. Pareto transfer-sizes appear to be more difficult to analyze, but can be well approximated by the more tractable mixed-exponential model [9].

[3] The usual Pareto distribution takes values on (b, ∞) whereas the shifted-Pareto distribution used here takes values from $(0, \infty)$.

1.3 Analysis of the TCP SHS models

To investigate the dynamics of the moments of the sending rate $\mathbf{r}(t) = \frac{\mathbf{w}(t)}{RTT(t)}$, we denote by $\mu_{q_0,n}(t)$ the nth-order (uncentered) statistical moment of $\mathbf{r}(t)$ restricted to a particular mode $\mathbf{q} = q_0 \in \mathscr{Q} := \{off, ss_i, d\text{-}ss_i, ca_i, d\text{-}ca_i : i = 1,2,\ldots,M\}$. This is captured by the following definition

$$\mu_{q_0,n}(t) := \mathrm{E}\left[\psi_{q_0,n}(\mathbf{q}(t),\mathbf{w}(t),t)\right], \quad \psi_{q_0,n}(q,w,t) := \begin{cases} \frac{w^n}{RTT(t)^n} & q = q_0 \\ 0 & \text{otherwise,} \end{cases} \tag{1.12}$$

$\forall n \geq 0$, $q_0 \in \mathscr{Q}$. The probability that the flow is in mode $q_0 \in \mathscr{Q}$ at time t is then given by

$$\mathrm{P}(\mathbf{q}(t) = q_0) = \mu_{q_0,0}(t), \quad \forall t \geq 0;$$

the nth-order statistical moment of $\mathbf{r}(t)$, conditioned to the flow being in mode $q_0 \in \mathscr{Q}$ is given by

$$\mathrm{E}[\mathbf{r}^n(t) \mid \mathbf{q}(t) = q_0] = \frac{\mu_{q_0,n}(t)}{\mathrm{P}(\mathbf{q}(t) = q_0)} = \frac{\mu_{q_0,n}(t)}{\mu_{q_0,0}(t)}, \quad \forall t \geq 0; \tag{1.13}$$

and the nth-order statistical moment of the overall sending rate is given by

$$\mathrm{E}[\mathbf{r}^n(t)] = \sum_{q \in \mathscr{Q}} \mu_{q,n}(t), \quad \forall t \geq 0.$$

The following result shows that these moments are the solution of an infinite-dimensional system of ODEs that can be obtained by direct application of results in [11, 13]. Details are provided in the appendix.

THEOREM 1.1 **Infinite-dimensional models.** *The statistical moments of the long-lived flows model satisfy the following equations*[4]*:*

$$\dot{\mu}_{ca_i,n} = \frac{n\mu_{ca_i,n-1}}{n_{\mathrm{ack}} RTT^2} - \frac{nR\dot{T}T\mu_{ca_i,n}}{RTT} - p_{\mathrm{drp}}\mu_{ca_i,n+1} + \frac{\mu_{d\text{-}ca_i,n}}{2^n k_{\mathrm{dly}} RTT}, \tag{1.14a}$$

$$\dot{\mu}_{d\text{-}ca_i,n} = \frac{n\mu_{d\text{-}ca_i,n-1}}{n_{\mathrm{ack}} RTT^2} - \frac{nR\dot{T}T\mu_{d\text{-}ca_i,n}}{RTT} + p_{\mathrm{drp}}\mu_{ca_i,n+1} - \frac{\mu_{d\text{-}ca_i,n}}{k_{\mathrm{dly}} RTT}. \tag{1.14b}$$

The statistical moments of the mixed-exponentials transfer-sizes model:

$$\dot{\mu}_{\mathrm{off},0} = -\frac{\mu_{\mathrm{off},0}}{\tau_{\mathrm{off}}} + \sum_{j=1}^{M} k_j^{-1}(\mu_{ss_j,1} + \mu_{ca_j,1}), \tag{1.15a}$$

[4] To simplify the notation, we omit the time-dependence of RTT and p_{drp}.

$$\dot{\mu}_{ss_i,n} = \frac{p_i w_0^n \mu_{\text{off},0}}{\tau_{\text{off}} RTT^n} + n\frac{(\log 2) - n_{\text{ack}} R\dot{T}T}{n_{\text{ack}} RTT}\mu_{ss_i,n} - (p_{\text{drp}} + k_i^{-1})\mu_{ss_i,n+1}, \tag{1.15b}$$

$$\dot{\mu}_{d\text{-}ss_i,n} = n\frac{(\log 2) - n_{\text{ack}} R\dot{T}T}{n_{\text{ack}} RTT}\mu_{d\text{-}ss_i,n} + p_{\text{drp}}\mu_{ss_i,n+1} - \frac{\mu_{d\text{-}ss_i,n}}{k_{\text{dly}} RTT}, \tag{1.15c}$$

$$\dot{\mu}_{ca_i,n} = \frac{n\mu_{ca_i,n-1}}{n_{\text{ack}} RTT^2} - \frac{nR\dot{T}T\mu_{ca_i,n}}{RTT} - (p_{\text{drp}} + k_i^{-1})\mu_{ca_i,n+1} + \frac{\mu_{d\text{-}ss_i,n} + \mu_{d\text{-}ca_i,n}}{2^n k_{\text{dly}} RTT}, \tag{1.15d}$$

$$\dot{\mu}_{d\text{-}ca_i,n} = \frac{n\mu_{d\text{-}ca_i,n-1}}{n_{\text{ack}} RTT^2} - \frac{nR\dot{T}T\mu_{d\text{-}ca_i,n}}{RTT} + p_{\text{drp}}\mu_{ca_i,n+1} - \frac{\mu_{d\text{-}ca_i,n}}{k_{\text{dly}} RTT}. \tag{1.15e}$$

The statistical moments of the Pareto transfer-sizes model:

$$\dot{\nu}_{\text{off},0,0} = -\frac{\nu_{\text{off},0,0}}{\tau_{\text{off}}} + a(\nu_{ss,1,1} + \nu_{ca,1,1}), \tag{1.16a}$$

$$\dot{\nu}_{ss,n,m} = \frac{w_0^n \nu_{\text{off},0,0}}{b^m RTT^n \tau_{\text{off}}} + n\frac{(\log 2) - n_{\text{ack}} R\dot{T}T}{n_{\text{ack}} RTT}\nu_{ss,n,m} - p_{\text{drp}}\nu_{ss,n+1,m} - (m+a)\nu_{ss,n+1,m+1}, \tag{1.16b}$$

$$\dot{\nu}_{d\text{-}ss,n,m} = n\frac{(\log 2) - n_{\text{ack}} R\dot{T}T}{n_{\text{ack}} RTT}\nu_{d\text{-}ss,n,m} + p_{\text{drp}}\nu_{ss,n+1,m} - \frac{\nu_{d\text{-}ss,n,m}}{k_{\text{dly}} RTT} - m\nu_{d\text{-}ss,n+1,m+1}, \tag{1.16c}$$

$$\begin{aligned}\dot{\nu}_{ca,n,m} = {} & \frac{n\nu_{ca,n-1,m}}{n_{\text{ack}} RTT^2} - \frac{nR\dot{T}T\nu_{ca,n,m}}{RTT} - p_{\text{drp}}\nu_{ca,n+1,m} \\ & + \frac{\nu_{d\text{-}ss,n,m} + \nu_{d\text{-}ca,n,m}}{2^n k_{\text{dly}} RTT} - (m+a)\nu_{ca,n+1,m+1},\end{aligned} \tag{1.16d}$$

$$\dot{\nu}_{d\text{-}ca,n,m} = \frac{n\nu_{d\text{-}ca,n-1,m}}{n_{\text{ack}} RTT^2} - \frac{nR\dot{T}T\nu_{d\text{-}ca,n,m}}{RTT} + p_{\text{drp}}\nu_{ca,n+1,m} - \frac{\nu_{d\text{-}ca,n,m}}{k_{\text{dly}} RTT} - m\nu_{d\text{-}ca,n+1,m+1}, \tag{1.16e}$$

where $\nu_{q_0,n,0}$, $q_0 \in \mathcal{Q}, n \geq 0$ *is used to denote* $\mu_{q_0,n}$ *and, for every* $m > 0$,

$$\nu_{q_0,n,m}(t) := \mathrm{E}\left[\varphi_{q_0,n,m}(\mathbf{q}(t), \mathbf{w}(t), \mathbf{s}(t))\right],$$

$$\varphi_{q_0,n,m}(q,w,s,t) := \begin{cases} \frac{w^n}{RTT(t)^n (s+b)^m} & q = q_0 \\ 0 & \text{otherwise.} \end{cases}$$

1.4 Reduced-order models

The systems of infinitely many ODEs[5] that appear in Theorem 1.1 describe *exactly* the evolution of the moments of the sending rate $\mathbf{r}$, but finding a solution to these equations does not appear to be simple. However, as noted by Bohacek [5] and

[5] Notice that the integer n ranges from 0 to ∞.

others, Monte Carlo simulations reveal that the steady-state distribution of the sending rate is well approximated by a log-normal distribution. Assuming that on each mode the sending rate $\mathbf{r}$ approximately obeys a log-normal distribution even during transients, we can truncate the systems of infinitely many differential equations that appear in Theorem 1.1. This procedure is known as *moment closure*.

We recall that, if the random variable $\mathbf{x}$ has a log-normal distribution then $\mathrm{E}[\mathbf{x}^3] = \frac{\mathrm{E}[\mathbf{x}^2]^3}{\mathrm{E}[\mathbf{x}]^3}$. This means that if $\mathbf{r}$ is approximately log-normal distributed in the mode $q \in \mathscr{Q}$, we have that

$$\mu_{q,3} = \mu_{q,0}\,\mathrm{E}[\mathbf{r}^3 \mid \mathbf{q} = q] \approx \mu_{q,0}\frac{\mathrm{E}[\mathbf{r}^2 \mid \mathbf{q} = q]^3}{\mathrm{E}[\mathbf{r} \mid \mathbf{q} = q]^3} = \frac{\mu_{q,0}\,\mu_{q,2}^3}{\mu_{q,1}^3}, \tag{1.17}$$

where we used (1.13). Using (1.17) in (1.14)–(1.15), we can eliminate any terms $\mu_{q_0,n}$, $n \geq 3$ in the equations for $\dot{\mu}_{q_0,n}$, $n \leq 2$, thus constructing a finite-dimensional model to approximately describe the dynamics of the first two moments of the sending rate. The reader is referred to [13] for a more detailed treatment on the use of this type of truncations to analyze SHS.

1.4.1 Long-lived flows

The following model for long-lived TCP flows can be obtained from (1.14) using the approximation in (1.17).

$$\dot{\mu}_{ca,0} = \frac{1-\mu_{ca,0}}{k_{\mathrm{dly}}RTT} - p_{\mathrm{drp}}\,\mu_{ca,1}, \tag{1.18a}$$

$$\dot{\mu}_{ca,1} = \frac{\mu_{ca,0}}{n_{\mathrm{ack}}RTT^2} - \frac{R\dot{T}T\,\mu_{ca,1}}{RTT} - p_{\mathrm{drp}}\,\mu_{ca,2} + \frac{\mu_{d\text{-}ca,1}}{2k_{\mathrm{dly}}RTT}, \tag{1.18b}$$

$$\dot{\mu}_{d\text{-}ca,1} = \frac{1-\mu_{ca,0}}{n_{\mathrm{ack}}RTT^2} + p_{\mathrm{drp}}\,\mu_{ca,2} - \frac{R\dot{T}T\,\mu_{d\text{-}ca,1}}{RTT} - \frac{\mu_{d\text{-}ca,1}}{k_{\mathrm{dly}}RTT}, \tag{1.18c}$$

$$\dot{\mu}_{ca,2} = \frac{2\,\mu_{ca,1}}{n_{\mathrm{ack}}RTT^2} - \frac{2R\dot{T}T\,\mu_{ca,2}}{RTT} - \frac{p_{\mathrm{drp}}\,\mu_{ca,0}\,\mu_{ca,2}^3}{\mu_{ca,1}^3} + \frac{\mu_{d\text{-}ca,2}}{4k_{\mathrm{dly}}RTT}, \tag{1.18d}$$

$$\dot{\mu}_{d\text{-}ca,2} = \frac{p_{\mathrm{drp}}\,\mu_{ca,0}\,\mu_{ca,2}^3}{\mu_{ca,1}^3} + \frac{2\,\mu_{d\text{-}ca,1}}{n_{\mathrm{ack}}RTT^2} - \frac{2R\dot{T}T\,\mu_{d\text{-}ca,2}}{RTT} - \frac{\mu_{d\text{-}ca,2}}{k_{\mathrm{dly}}RTT}. \tag{1.18e}$$

This model has a stable equilibrium point at[6]

$$\mu_{ca,0} = 1 + \frac{2k_{\text{dly}}^2 p_{\text{drp}}}{n_{\text{ack}}} - k_{\text{dly}} p_{\text{drp}} W, \tag{1.19a}$$

$$\mu_{ca,1} = \frac{W}{RTT} - \frac{2k_{\text{dly}}}{n_{\text{ack}} RTT}, \tag{1.19b}$$

$$\mu_{ca,2} = \frac{2n_{\text{ack}} - (n_{\text{ack}} W - 2k_{\text{dly}})k_{\text{dly}} p_{\text{drp}}}{n_{\text{ack}}^2 p_{\text{drp}} RTT^2}, \tag{1.19c}$$

$$\mu_{d\text{-}ca,1} = \frac{2k_{\text{dly}}}{n_{\text{ack}} RTT}, \qquad \mu_{d\text{-}ca,2} = \frac{8k_{\text{dly}} W}{3n_{\text{ack}} RTT^2}, \tag{1.19d}$$

$$p_{\text{drp}} \approx \frac{12n_{\text{ack}}}{n_{\text{ack}} W - 2k_{\text{dly}}} \frac{1}{15k_{\text{dly}} + \sqrt{48n_{\text{ack}}^2 W^2 - 168k_{\text{dly}} n_{\text{ack}} W + 45k_{\text{dly}}^2}}, \tag{1.19e}$$

where W denotes the *delay-throughput product* defined by

$$W := RTT\,\mathrm{E}[\mathbf{r}] = (\mu_{ca,1} + \mu_{d\text{-}ca,1})RTT,$$

which is also equal to the average window size $\mathrm{E}[\mathbf{w}]$. Figure 1.2 compares the equilibrium points obtained from this reduced model with the steady-state values obtained from Monte Carlo simulations of the original SHS. The Monte Carlo simulations were obtained using the algorithm described in [12, Section 2.1]. The match is essentially perfect, confirming the validity of the log-normal approximation.

In the same plot we also included what the TCP-friendly formula (1.1) would have predicted for the total sending rate. We can see that both the Monte Carlo simulations of the original SHS and the reduced model (1.18) basically agree with the TCP-friendly formula (1.1) for the average sending rate. However, the stochastic models also provide information about the variability in the sending rate from flow to flow (measured by the standard deviation).

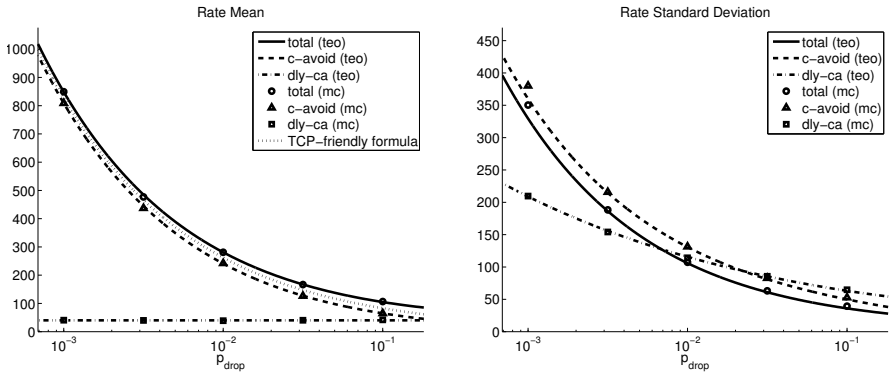
1.4.2 Mixed-exponential transfer-sizes

We consider now a TCP SHS model for mixed-exponentially distributed transfer-sizes. To keep the model small, we assume that the delays are negligible $k_{\text{dly}} \approx 0$. This model can be obtained from (1.15) by making $k_{\text{dly}} \downarrow 0$ and realizing that in this case the variables $\mu_{d\text{-}ss,n}$ and $\mu_{d\text{-}ca,n}$ exhibit fast dynamics with equilibrium at

$$\frac{\mu_{d\text{-}ss_i,n}}{k_{\text{dly}} RTT} = \frac{n_{\text{ack}}\, p_{\text{drp}} \mu_{ss_i,n+1}}{n_{\text{ack}} - k_{\text{dly}} n\big((\log 2) - n_{\text{ack}} R\dot{T}T\big)}.$$

$$\frac{\mu_{d\text{-}ca_i,n}}{k_{\text{dly}} RTT} = \frac{n\mu_{d\text{-}ca_i,n-1}}{n_{\text{ack}} RTT^2 (1 + nk_{\text{dly}} R\dot{T}T)} + \frac{p_{\text{drp}} \mu_{ca_i,n+1}}{1 + nk_{\text{dly}} R\dot{T}T}.$$

[6] The approximate value for p_{drp} was obtained by considering a quadratic approximation around $p_{\text{drp}} = 0$ to the equation $\dot{\mu}_{d\text{-}ca,2} = 0$ with the variables $\mu_{ca,0}$, $\mu_{ca,1}$, $\mu_{ca,2}$, $\mu_{d\text{-}ca,1}$, $\mu_{d\text{-}ca,2}$ eliminated through the use of the remaining equations. It is possible to compute the exact value of p_{drp} at equilibrium, but (1.19e) is much simpler and provides a very good approximation.



(a) (b)

FIGURE 1.2: Steady-state values for the average (a) and standard deviation (b) of the sending rate as a function of the drop probability, for the long-lived SHS model. The dashed and dot-dashed lines provide the contributions to the sending rate from the *ca* and *d-ca* modes, respectively. All lines were obtained from (1.19) with $RTT = 50$ ms, $k_{\rm dly} = 1$, $n_{\rm ack} = 1$. The circle, triangle, and square symbols were obtained from Monte Carlo simulations. The dotted line corresponds to the TCP-friendly formula (1.1).

As $k_{\rm dly} \downarrow 0$, we obtain

$$\frac{\mu_{d\text{-}ss_i,n}}{k_{\rm dly}RTT} \to p_{\rm drp}\mu_{ss_i,n+1}, \quad \frac{\mu_{d\text{-}ca_i,n}}{k_{\rm dly}RTT} \to \frac{n\mu_{d\text{-}ca_i,n-1}}{n_{\rm ack}RTT^2} + p_{\rm drp}\mu_{ca_i,n+1} \to p_{\rm drp}\mu_{ca_i,n+1}.$$

Replacing this into (1.15), yields[7]

$$\dot{\mu}_{{\rm off},0} = -\frac{\mu_{{\rm off},0}}{\tau_{\rm off}} + \sum_{j=1}^{M} k_j^{-1}(\mu_{ss_j,1} + \mu_{ca_j,1}), \tag{1.20a}$$

$$\dot{\mu}_{ss_i,n} = \frac{p_i w_0^n \mu_{{\rm off},0}}{\tau_{\rm off} RTT^n} + n\frac{(\log 2) - n_{\rm ack}R\dot{T}T}{n_{\rm ack}RTT}\mu_{ss_i,n} - (p_{\rm drp} + k_i^{-1})\mu_{ss_i,n+1}, \tag{1.20b}$$

$$\dot{\mu}_{ca_i,n} = \frac{n\mu_{ca_i,n-1}}{n_{\rm ack}RTT^2} - \frac{nR\dot{T}T\mu_{ca_i,n}}{RTT} - (p_{\rm drp} + k_i^{-1})\mu_{ca_i,n+1} + \frac{p_{\rm drp}}{2^n}(\mu_{ss_i,n+1} + \mu_{ca_i,n+1}). \tag{1.20c}$$

We can now use (1.17) to construct from (1.20) the following finite dimensional approximate model.

$$\dot{\mu}_{ss_i,0} = \frac{p_i\big(1 - \sum_{j=1}^{M}(\mu_{ss_i,0} + \mu_{ca_i,0})\big)}{\tau_{\rm off}} - (p_{\rm drp} + k_i^{-1})\,\mu_{ss_i,1}, \tag{1.21a}$$

[7] These equations could also be derived directly from a SHS model without the delay modes *d-ss* and *d-ca*.

$$\dot{\mu}_{ca_i,0} = p_{\text{drp}}\,\mu_{ss_i,1} - k_i^{-1}\mu_{ca_i,1}, \tag{1.21b}$$

$$\dot{\mu}_{ss_i,1} = \frac{w_0\,p_i\big(1-\sum_{j=1}^{M}(\mu_{ss_i,0}+\mu_{ca_i,0})\big)}{\tau_{\text{off}}\,RTT} + \frac{(\log 2) - n_{\text{ack}}\,R\dot{T}T}{n_{\text{ack}}\,RTT}\mu_{ss_i,1} - (p_{\text{drp}}+k_i^{-1})\,\mu_{ss_i,2}, \tag{1.21c}$$

$$\dot{\mu}_{ca_i,1} = \frac{\mu_{ca_i,0}}{n_{\text{ack}}\,RTT^2} - \frac{R\dot{T}T\,\mu_{ca_i,1}}{RTT} + \frac{p_{\text{drp}}\mu_{ss_i,2}}{2} - \big(\frac{p_{\text{drp}}}{2}+k_i^{-1}\big)\mu_{ca_i,2}, \tag{1.21d}$$

$$\dot{\mu}_{ss_i,2} = \frac{w_0^2\,p_i\big(1-\sum_{j=1}^{M}(\mu_{ss_i,0}+\mu_{ca_i,0})\big)}{\tau_{\text{off}}\,RTT^2} + \frac{(\log 4)\,\mu_{ss_i,2}}{n_{\text{ack}}\,RTT} - \frac{2R\dot{T}T\,\mu_{ss_i,2}}{RTT} - (p_{\text{drp}}+k_i^{-1})\frac{\mu_{ss_i,0}\,\mu_{ss_i,2}^3}{\mu_{ss_i,1}^3}, \tag{1.21e}$$

$$\dot{\mu}_{ca_i,2} = \frac{2\,\mu_{ca_i,1}}{n_{\text{ack}}\,RTT^2} - \frac{2R\dot{T}T\,\mu_{ca_i,2}}{RTT} + \frac{p_{\text{drp}}}{4}\frac{\mu_{ss_i,0}\,\mu_{ss_i,2}^3}{\mu_{ss_i,1}^3} - \Big(\frac{3\,p_{\text{drp}}}{4}+k_i^{-1}\Big)\frac{\mu_{ca_i,0}\,\mu_{ca_i,2}^3}{\mu_{ca_i,1}^3}. \tag{1.21f}$$

We present next simulations of these differential equations for a few representative parameter values.

Figure 1.3 corresponds to a transfer-size distribution that results from the mixture of two exponentials ($M = 2$) with parameters

$$\begin{aligned} k_1 &= 3.5\text{ KB}, & k_2 &= 246\text{ KB}, \\ p_1 &= 88.87\%, & p_2 &= 11.13\%. \end{aligned} \tag{1.22}$$

The first exponential corresponds to small "mice" transfers (3.5 KB average) and the second to "elephant" mid-size transfers (246 KB average.) The small transfers are assumed more common (88.87%.) These parameters result in a distribution with an average transfer-size of 30.58 KB and for which 11.13% of the transfers account for 89.7% of the total volume transfered. This is consistent with the file distribution observed in the UNIX file system [14]. However, it does not accurately capture the tail of the distribution (it lacks the "mammoth" files that will be considered later.) The results obtained with the reduced model (1.21) still match quite well those obtained from Monte Carlo simulations of the original SHS, especially taking into account the very large standard deviations. Also here, the Monte Carlo simulations were obtained using the algorithm described in [12, Section 2.1]. It is worth to point out that the simulation of (1.21) takes just a few seconds, whereas each Monte Carlo simulation takes orders of magnitude more time.

Two somewhat surprising conclusions can be drawn from Figure 1.3 for this distribution of transfer-sizes and off-periods:

1. The *average total sending rate varies very little with the per-packet drop probability*, at least up to the drop probabilities of 33% shown in Figure 1.3(b). This is perhaps not surprising when most packets are transmitted in the slow-start mode, but this phenomenon persists even when a significant fraction of packets are sent in the congestion-avoidance mode [which occurs for drop probabilities above .8%, as seen in Figure 1.3(b)].

2. The *dynamics of individual TCP flows are dominated by second order moments.* In Figure 1.3, the standard deviation is 5 to 20 times larger than the average sending rate, which is very accurately predicted by the reduced model.

This behavior is quite different from the one observed for the long-lived flows considered in Section 1.4.1, where the average sending rate varies significantly with the drop probability and its standard deviation is less than half of its average value. The reader is encouraged to compare the plots in Figure 1.2 for long-lived flows with the corresponding plots in Figure 1.3 for on-off flows.

We recall that both figures correspond to the statistics of a single-user TCP flow. It is not surprising to find out that for a single-user and the same packet drop probability, a long-lived flow will utilize much more bandwidth than an on-off flow. This results in the observed difference in the vertical-axis scales between the plots in Figures 1.2 and 1.3. When one aggregates the flows of n (independent) users, the vertical scales in Figures 1.2 and 1.3 will appear multiplied by n (for the average rate) and by $\sqrt{n}$ (for its standard deviation.) However, the shape of the curves (as p_{drop} varies) will not change and we still conclude that for on-off flows the drop probability exercises a much tighter control on the standard deviation than on the average sending rate of n users.

We consider next a transfer-size distribution that results from a mixture of three exponentials ($M = 3$) with parameters

$$k_1 = 6\ \text{KB}, \qquad k_2 = 400\ \text{KB}, \qquad k_3 = 10\ \text{MB}, \tag{1.23a}$$

$$p_1 = 98\%, \qquad p_2 = 1.7\%, \qquad p_3 = .02\%. \tag{1.23b}$$

The first exponential corresponds to small "mice" transfers, the second to mid-size "elephant" transfers, and the third to large "mammoth" transfers. The resulting distribution, shown in Figure 1.4, approximates reasonably well the one reported by Arlitt et al. [1] obtained from monitoring transfers from a world-wide web proxy within an Internet Service Provider (ISP), at least for transfer-sizes up to 100 MB, for which data is available. This distribution has a much heavier tail than the one considered before.

Figure 1.5 contains results obtained from the reduced model. We do not present Monte Carlo results because the simulation times needed to capture the tails of the transfer-size distribution are prohibitively large. It turns out that the main conclusions drawn before still hold: The average sending rate varies relatively little with the packet drop probability and the dynamics of TCP flows are dominated by second order moments. The mid-size "elephants" still dominate followed by the small "mice." The large "mammoth" transfers occur at a rate that is not sufficiently large to have a significant impact on the average sending rate.

1.5 Conclusions

We presented a stochastic model for on-off TCP flows that considers both slow-start and congestion-avoidance. This model takes directly into account the distribution of the transfer-sizes to determine the probability of flows being active.

One important observation that stems from this work is that for realistic transfer-size distributions, high-order statistical moments seem to dominate the dynamics of TCP flows, with a standard deviation of the sending rate much larger than its average value. Also, the drop probability appears to have a much more significant effect on the standard deviation of the sending rate than on its average value. This may have significant implications for the congestion control: It seems to indicate that previously used long-lived flow models are not suitable for the analysis and design of congestion control algorithms; and also questions the validity of the "TCP-friendly" formula for the aggregation of many single-user on-off TCP flows.

We are currently investigating how a maximum congestion window size imposed by the receiver affects the dynamics of TCP. It is straightforward to incorporate a maximum window size of $W_{\max}$ in the SHS model in Figure 1.1: In essence one would simply replace $\mathbf{r} = \mathbf{w}/RTT$ by $\mathbf{r} = \min\{\mathbf{w}, W_{\max}\}/RTT$. However, the repercussions of this modification will almost certainly be more complex than limiting the average sending rate to be below $W_{\max}/RTT$, especially in view of the large standard deviations. Future work also includes the analysis and design of active queue management algorithms, based on these models.

Acknowledgments. We thank Martin Arlitt for making available the processed data regarding the transfer-size distribution used in Section 1.4.2; and Stephan Bohacek and Katia Obraczka for insightful discussions. The material in this chapter is based upon work supported by the National Science Foundation under Grants No. CCR-0311084, ANI-0322476.

Appendix

Stochastic Hybrid Systems

For completeness we recall the definition of a *Stochastic Hybrid System* (SHS) introduced by Hespanha [11]. Formally, a SHS is defined by a differential equation

$$\dot{\mathbf{x}} = f(\mathbf{q}, \mathbf{x}, t), \tag{1.24}$$

a family of *m discrete transition/reset maps*

$$(\mathbf{q}, \mathbf{x}) = \phi_\ell(\mathbf{q}^-, \mathbf{x}^-, t), \qquad \ell \in \{1, \ldots, m\}, \tag{1.25}$$

and a family of *m transition intensities*

$$\lambda_\ell(\mathbf{q},\mathbf{x},t), \qquad \ell \in \{1,\ldots,m\}, \tag{1.26}$$

where $\mathscr{Q}$ denotes a (typically finite) set and $f : \mathscr{Q}\times\mathbb{R}^n\times[0,\infty)\to\mathbb{R}^n$, $\phi_\ell : \mathscr{Q}\times\mathbb{R}^n\times[0,\infty)\to\mathscr{Q}\times\mathbb{R}^n$, $\lambda_\ell : \mathscr{Q}\times\mathbb{R}^n\times[0,\infty)\to[0,\infty)$, $\forall\ell\in\{1,\ldots,m\}$. A SHS characterizes a jump process $\mathbf{q} : \Omega\times[0,\infty)\to\mathscr{Q}$ called the *discrete state*; a stochastic process $\mathbf{x} : \Omega\times[0,\infty)\to\mathbb{R}^n$ with piecewise continuous sample paths called the *continuous state*; and m stochastic counters $\mathbf{N}_\ell : \Omega\times[0,\infty)\to\mathbb{N}_{>0}$ called the *transition counters*.

In essence, between transition counter increments the discrete state remains constant whereas the continuous state flows according to (1.24). At transition times the continuous and discrete states are reset according to (1.25). Each transition counter $\mathbf{N}_\ell$ counts the number of times that the corresponding discrete transition/reset map ϕ_ℓ is "activated." The frequency at which this occurs is determined by the transition intensities (1.26). In particular, the probability that the counter $\mathbf{N}_\ell$ will increment and therefore that the corresponding transition takes place in an "elementary interval" $(t,t+dt]$ is given by $\lambda_\ell(\mathbf{q}(t),\mathbf{x}(t),t)dt$. In practice, one can think of the intensity of a transition as the instantaneous rate at which this transition occurs. The reader is referred to [11] for a mathematically precise characterization of a SHS.

It is often convenient to represent a SHS by a directed graph as in Figure 1.6, where each node corresponds to a discrete mode and each edge to a transition between discrete modes. The nodes are labeled with the corresponding discrete mode and the vector fields that determines the evolution of the continuous state in that particular mode. The start of each edge is labeled with the corresponding transition intensity and the end is labeled with the reset map.

The following result can be used to compute expectations on the state of a SHS. For simplicity of presentation we omit a few technical assumptions that are straightforward to verify for the SHS considered here:

***THEOREM 1.2** [11] For every function $\psi : \mathscr{Q}\times\mathbb{R}^n\times[0,\infty)\to\mathbb{R}$ that is continuously differentiable with respect to its second and third arguments, we have that*

$$\frac{\mathrm{d}}{\mathrm{d}t}\,\mathrm{E}[\psi(\mathbf{q}(t),\mathbf{x}(t),t)] = \mathrm{E}[(L\psi)(\mathbf{q}(t),\mathbf{x}(t),t)], \tag{1.27}$$

where, for every $(q,x,t)\in\mathscr{Q}\times\mathbb{R}^n\times[0,\infty)$,

$$(L\psi)(q,x,t) := \frac{\partial\psi(q,x,t)}{\partial x}f(q,x,t) + \frac{\partial\psi(q,x,t)}{\partial t} + \\ + \sum_{\ell=1}^{m}\Big(\psi\big(\phi_\ell(q,x,t),t\big)-\psi(q,x,t)\Big)\lambda_\ell(q,x,t), \tag{1.28}$$

and $\frac{\partial\psi(q,x,t)}{\partial x}$ and $\frac{\partial\psi(q,x,t)}{\partial t}$ denote the gradient of $\psi(q,x,t)$ with respect to x and the partial derivative of $\psi(q,x,t)$ with respect to t, respectively. The operator $\psi\mapsto L\psi$ defined by (1.28) *is called the* extended generator *of the SHS.* □

The transition intensities and reset maps for the TCP SHS model in Figure 1.1 are defined as follows:

$$\lambda_{\text{drp}}(q,w,s,t) := \begin{cases} \frac{p_{\text{drp}}(t)w}{RTT} & q \in \{ss, ca\} \\ 0 & \text{otherwise} \end{cases} \qquad \phi_{\text{drp}}(q,w,s,t) := \begin{cases} (d\text{-}ss, w, s) & q = ss \\ (d\text{-}ca, w, s) & q = ca \\ (q,w,s) & \text{otherwise} \end{cases}$$

$$\lambda_{\text{d2ca}}(q,w,s,t) := \begin{cases} \frac{1}{k_{\text{dly}}RTT} & q \in \{d\text{-}ss, d\text{-}ca\} \\ 0 & \text{otherwise} \end{cases} \qquad \phi_{\text{d2ca}}(q,w,s,t) := \begin{cases} \left(ca, \frac{w}{2}, s\right) & q \in \{d\text{-}ss, d\text{-}ca\} \\ (q,w,s) & \text{otherwise} \end{cases}$$

$$\lambda_{\text{start}}(q,w,s,t) := \begin{cases} \frac{1}{\tau_{\text{off}}} & q = off \\ 0 & \text{otherwise} \end{cases} \qquad \phi_{\text{start}}(q,w,s,t) := \begin{cases} (ss, w_0, 0) & q = off \\ (q,w,s) & \text{otherwise} \end{cases}$$

$$\lambda_{\text{end}}(q,w,s,t) := \begin{cases} \frac{wF'_{s^*}(s)}{(1-F_{s^*}(s))RTT} & q \in \{ss, ca\} \\ 0 & \text{otherwise} \end{cases} \qquad \phi_{\text{end}}(q,w,s,t) := \begin{cases} (off, 0, 0) & q \in \{ss, ca\} \\ (q,w,s) & \text{otherwise,} \end{cases}$$

where

$$w_0 := \begin{cases} .693 & n_{\text{ack}} = 1 \\ 1.428 & n_{\text{ack}} = 2. \end{cases}$$

For the mixed-exponentials model, the intensities and reset maps are given by

$$\lambda_{\text{drp}}(q,w,s,t) := \begin{cases} \frac{p_{\text{drp}}(t)w}{RTT} & q \in \{ss_i, ca_i : \forall i\} \\ 0 & \text{otherwise} \end{cases} \qquad \phi_{\text{drp}}(q,w,s,t) := \begin{cases} (d\text{-}ss_1, w, s) & q = ss_1 \\ \vdots & \vdots \\ (d\text{-}ss_M, w, s) & q = ss_M \\ (d\text{-}ca_1, w, s) & q = ca_1 \\ \vdots & \vdots \\ (d\text{-}ca_M, w, s) & q = ca_M \\ (q,w,s) & \text{otherwise} \end{cases}$$

$$\lambda_{\text{d2ca}}(q,w,s,t) := \begin{cases} \frac{k_{\text{dly}}^{-1}}{RTT} & q \in \{d\text{-}ss_i, d\text{-}ca_i : \forall i\} \\ 0 & \text{otherwise} \end{cases} \qquad \phi_{\text{d2ca}}(q,w,s,t) := \begin{cases} \left(ca_1, \frac{w}{2}, s\right) & q \in \{d\text{-}ss_1, d\text{-}ca_1\} \\ \vdots & \vdots \\ \left(ca_M, \frac{w}{2}, s\right) & q \in \{d\text{-}ss_M, d\text{-}ca_M\} \\ (q,w,s) & \text{otherwise} \end{cases}$$

$$\lambda_i(q,w,s,t) := \begin{cases} \frac{p_i}{\tau_{\text{off}}} & q = off \\ 0 & \text{otherwise} \end{cases} \qquad \phi_i(q,w,s,t) := \begin{cases} (ss_i, w_0, 0) & q = off \\ (q,w,s) & \text{otherwise} \end{cases}$$

$$\lambda_{\text{end}}(q,w,s,t) := \begin{cases} \frac{k_1^{-1}w}{RTT} & q \in \{ss_1, ca_1\} \\ \vdots & \vdots \\ \frac{k_M^{-1}w}{RTT} & q \in \{ss_M, ca_M\} \\ 0 & \text{otherwise} \end{cases} \qquad \phi_{\text{end}}(q,w,s,t) := \begin{cases} (off, 0, 0) & q \in \{ss_i, ca_i : \forall i\} \\ (q,w,s) & \text{otherwise,} \end{cases}$$

where the λ_i and ϕ_i, $i \in \{1, 2, \ldots, M\}$ replace λ_{start} and ϕ_{start}, respectively, in the previous model.

Proofs

Proof of Theorem 1.1. Applying to the functions $\psi_{q_0,n}$ defined in (1.12) the extended generator L for the mixed-exponentials SHS model yields

$$\begin{aligned}
L\psi_{q_0,n}(q,w,s,t) &= \frac{\partial \psi_{q_0,n}(q,w,t)}{\partial w} f(q,w,s,t) + \frac{\partial \psi_{q_0,n}(q,w,t)}{\partial t} \\
&\quad + (\psi_{q_0,n}(\phi_{\text{drp}}(q,w,s,t),t) - \psi_{q_0,n}(q,w,t))\lambda_{\text{drp}}(q,w,s,t) \\
&\quad + (\psi_{q_0,n}(\phi_{\text{d2ca}}(q,w,s,t),t) - \psi_{q_0,n}(q,w,t))\lambda_{\text{d2ca}}(q,w,s,t) \\
&\quad + (\psi_{q_0,n}(\phi_{\text{end}}(q,w,s,t),t) - \psi_{q_0,n}(q,w,t))\lambda_{\text{end}}(q,w,s,t) \\
&\quad + \sum_{j=1}^{M} (\psi_{q_0,n}(\phi_j(q,w,s,t),t) - \psi_{q_0,n}(q,w,t))\lambda_j(q,w,s,t),
\end{aligned}$$

from which we obtain by direct computation that

$$\begin{aligned}
(L\psi_{\text{off},0})(q,w,t) &= \begin{cases} \frac{k_1^{-1}w}{RTT} & q \in \{ss_1, ca_1\} \\ \vdots & \vdots \\ \frac{k_M^{-1}w}{RTT} & q \in \{ss_M, ca_M\} \\ -\frac{1}{\tau_{\text{off}}} & q = off \\ 0 & \text{otherwise} \end{cases} \\
&= -\frac{\psi_{\text{off},0}(q,w,t)}{\tau_{\text{off}}} + \sum_{j=1}^{M} k_j^{-1}\left(\psi_{ss_j,1}(q,w,t) + \psi_{ca_j,1}(q,w,t)\right) \\
(L\psi_{ss_i,n})(q,w,t) &= \\
&= \begin{cases} \frac{n\left((\log 2)n_{\text{ack}}^{-1} - R\dot{T}T\right)w^n - (p_{\text{drp}} + k_i^{-1})w^{n+1}}{RTT^{n+1}} & q = ss_i \\ \frac{p_i w_0^n}{\tau_{\text{off}} RTT^n} & q = off \\ 0 & \text{otherwise} \end{cases} \\
&= \frac{p_i w_0^n \psi_{\text{off},0}(q,w,t)}{\tau_{\text{off}} RTT^n} + n\frac{(\log 2) - n_{\text{ack}} R\dot{T}T}{n_{\text{ack}} RTT}\psi_{ss_i,n}(q,w,t) - (p_{\text{drp}} + k_i^{-1})\psi_{ss_i,n+1}(q,w,t) \\
(L\psi_{\text{d-}ss_i,n}(q,w,t)) &= \begin{cases} \frac{p_{\text{drp}} w^{n+1}}{RTT^{n+1}} & q = ss_i \\ \frac{n\left((\log 2)n_{\text{ack}}^{-1} - R\dot{T}T\right)w^n - k_{\text{dly}}^{-1}w^n}{RTT^{n+1}} & q = \text{d-}ss_i \\ 0 & \text{otherwise} \end{cases} \\
&= n\frac{(\log 2) - n_{\text{ack}} R\dot{T}T}{n_{\text{ack}} RTT}\psi_{\text{d-}ss_i,n}(q,w,t) + p_{\text{drp}}\psi_{ss_i,n+1}(q,w,t) - \frac{\psi_{\text{d-}ss_i,n}(q,w,t)}{k_{\text{dly}} RTT} \\
(L\psi_{ca_i,n}(q,w,t)) &= \\
&= \begin{cases} \frac{w^n}{2^n k_{\text{dly}} RTT^{n+1}} & q \in \{\text{d-}ss_i, \text{d-}ca_i\} \\ \frac{nn_{\text{ack}}^{-1}w^{n-1} - nR\dot{T}Tw^n - (p_{\text{drp}} + k_i^{-1})w^{n+1}}{RTT^{n+1}} & q = ca_i \\ 0 & \text{otherwise} \end{cases} \\
&= \frac{n\psi_{ca_i,n-1}(q,w,t)}{n_{\text{ack}} RTT^2} - \frac{nR\dot{T}T\psi_{ca_i,n}(q,w,t)}{RTT} - (p_{\text{drp}} + k_i^{-1})\psi_{ca_i,n+1}(q,w,t)
\end{aligned}$$

$$+\frac{\psi_{d\text{-}ss_i,n}(q,w,t)+\psi_{d\text{-}ca_i,n}(q,w,t)}{2^n k_{\text{dly}} RTT}$$

$$(L\psi_{d\text{-}ca_i,n}(q,w,t)) = \begin{cases} \frac{p_{\text{drp}} w^{n+1}}{RTT^{n+1}} & q = ca_i \\ \frac{n n_{\text{ack}}^{-1} w^{n-1} - nR\dot{T}Tw^n - k_{\text{dly}}^{-1} w^n}{RTT^{n+1}} & q = d\text{-}ca_i \\ 0 & \text{otherwise} \end{cases}$$

$$= \frac{n\psi_{d\text{-}ca_i,n-1}(q,w,t)}{n_{\text{ack}} RTT^2} - \frac{nR\dot{T}T\psi_{d\text{-}ca_i,n}(q,w,t)}{RTT} + p_{\text{drp}}\psi_{ca_i,n+1}(q,w,t) - \frac{\psi_{d\text{-}ca_i,n}(q,w,t)}{k_{\text{dly}} RTT}.$$

To obtain (1.15), we use (1.27) to conclude that

$$\dot{\mu}_{q_0,n} = \mathrm{E}[(L\psi_{q_0,n})(\mathbf{q},\mathbf{w},t)],$$

and replace in the right-hand-side of this equation the expectations of the $\psi_{q_0,n}$ by the corresponding $\mu_{q_0,n}$. Equation (1.14) can be obtained from (1.15) by setting all the $k_j = \infty$ and consequently $\mu_{off,n} = \mu_{ss_j,n} = \mu_{d\text{-}ss_j,n} = 0$, $\forall n, j$ since in this SHS all the modes ss_j, $d\text{-}ss_j$, and *off* are absent. Equation (1.16) can be obtained along the lines of the derivation of (1.15). Due to space limitations we omit these computations.

Bibliography

[1] M. Arlitt, R. Friedrich, and T. Jin. Workload characterization of a web proxy in a cable modem environment. Technical Report HPL-1999-48, Hewlett-Packard Laroratories, Palo Alto, CA, Apr. 1999.

[2] F. Baccelli and D. Hong. Flow level simulation of large IP networks. In *Proc. of the IEEE INFOCOM*, Mar. 2003.

[3] F. Baccelli and D. Hong. Interaction of TCP flows as billiards. In *Proc. of the IEEE INFOCOM*, Mar. 2003.

[4] P. Barford, A. Bestavros, A. Bradley, and M. Crovella. Changes in web client access patterns. *World Wide Web,* Special Issue on Characterization and Performance Evaluation, 2(1–2):15–28, 1999.

[5] S. Bohacek. A stochastic model of TCP and fair video transmission. In *Proc. of the IEEE INFOCOM*, Mar. 2003.

[6] S. Bohacek, J. P. Hespanha, J. Lee, and K. Obraczka. Analysis of a TCP hybrid model. In *Proc. of the 39th Annual Allerton Conf. on Comm., Contr., and Computing*, Oct. 2001.

[7] S. Bohacek, J. P. Hespanha, J. Lee, and K. Obraczka. A hybrid systems modeling framework for fast and accurate simulation of data communication networks. In *Proc. of the ACM Int. Conf. on Measurements and Modeling of Computer Systems (SIGMETRICS)*, June 2003.

[8] A. B. Downey. Evidence for long-tailed distributions in the internet. In *Proc. of ACM SIGCOMM Internet Measurement Workshop*, Nov. 2001.

[9] A. Feldmann and W. Whitt. Fitting mixtures of exponentials to long-tail distributions to analyze network performance models. In *Proc. of the IEEE INFOCOM*, Apr. 1997.

[10] S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Trans. on Networking*, 1(4):397–413, Aug. 1993.

[11] J. P. Hespanha. Stochastic hybrid systems: Applications to communication networks. In R. Alur and G. J. Pappas, editors, *Hybrid Systems: Computation and Control*, number 2993 in Lect. Notes in Comput. Science, pages 387–401. Springer-Verlag, Berlin, Mar. 2004.

[12] J. P. Hespanha. A model for stochastic hybrid systems with application to communication networks. *Nonlinear Analysis,* Special Issue on Hybrid Systems, 62(8):1353–1383, Sept. 2005.

[13] J. P. Hespanha. Polynomial stochastic hybrid systems. In M. Morari and L. Thiele, editors, *Hybrid Systems: Computation and Control*, number 3414 in Lect. Notes in Comput. Science, pages 322–338. Springer-Verlag, Berlin, Mar. 2005.

[14] G. Irlam. Unix file size survey – 1993. Available at http://www.base.com /gordoni/ufs93.html, Nov. 1994.

[15] S. Kunniyur and R. Srikant. Analysis and design of an adaptive virtual queue (AVQ) algorithm for active queue management. In *Proc. of the ACM SIGCOMM*, San Diego, CA, Aug. 2001.

[16] A. Lakshmikantha, C. Beck, and R. Srikant. Robustness of real and virtual queue based active queue management schemes. In *Proc. of the 2003 Amer. Contr. Conf.*, pages 266–271, June 2003.

[17] Y. Liu, W.-B. Gong, V. Misra, and D. Towsley. On the tails of web file size distributions. In *Proc. of 39th Allerton Conference on Communication, Control, and Computing*, Oct. 2001.

[18] S. H. Low. A duality model of TCP and queue management algorithms. *IEEE/ACM Trans. on Networking*, 11(4), Aug. 2003.

[19] S. H. Low, F. Paganini, and J. C. Doyle. Internet congestion control. *IEEE Contr. Syst. Mag.*, 22(1):28–43, Feb. 2002.

[20] J. Mahdavi and S. Floyd. TCP-friendly unicast rate-based flow control. Technical note sent to the end2end-interest mailing list, Jan. 1997.

[21] M. A. Marsan, M. Garetto, P. Giaccone, E. Leonardi, E. Schiattarella, and A. Tarello. Using partial differential equations to model TCP mice and elephants in large IP networks. *Proc. of the IEEE INFOCOM*, Mar. 2004.

[22] M. Mathis, J. Semke, J. Mahdavi, and T. Ott. The macroscopic behavior of the TCP congestion avoidance algorithm. *ACM Comput. Comm. Review*, 27(3), July 1997.

[23] M. Mellia, I. Stoica, and H. Zhang. TCP model for short lived flows. *IEEE Comm. Lett.*, 6(2):85–87, Feb. 2002.

[24] V. Misra, W. Gong, and D. Towsley. Stochastic differential equation modeling and analysis of TCP-windowsize behavior. In *Proc. of PERFORMANCE '99*, Istanbul, Turkey, 1999.

[25] V. Misra, W. Gong, and D. Towsley. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED. In *Proc. of the ACM SIGCOMM*, Sept. 2000.

[26] T. Ott, J. H. B. Kemperman, and M. Mathis. Window size behavior in TCP/IP with constant loss probability. In *Proc. of the DIMACS Workshop on Performance of Realtime Applications on the Internet*, Nov. 1996.

[27] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP throughput: a simple model and its empirical validation. In *Proc. of the ACM SIGCOMM*, Sept. 1998.

[28] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP Reno performance: A simple model and its empirical validation. *IEEE/ACM Trans. on Networking*, 8(2):133–145, Apr. 2000.

[29] K. Park and W. Willinger. Self-similar network traffic: An overview. In K. Park and W. Willinger, editors, *Self-Similar Network Traffic and Performance Evaluation*. Wiley Interscience, New York, NY, 1999.

[30] S. Shakkottai and R. Srikant. How good are deterministic fluid models of Internet congestion control? In *Proc. of the IEEE INFOCOM*, June 2002.

[31] B. Sikdar, S. Kalyanaraman, and K. Vastola. Analytic models for the latency and steady-state throughput of TCP Tahoe, Reno and SACK. In *Proc. of the IEEE GLOBECOM*, pages 25–29, Nov. 2001.

[32] B. Sikdar, S. Kalyanaraman, and K. Vastola. TCP Reno with random losses: Latency, throughput and sensitivity analysis. In *Proc. of the IEEE IPCCC*, pages 188–195, Apr. 2001.

[33] W. Stallings. *High-speed networks: TCP/IP and ATM design principles*. Prentice-Hall, London, 1998.

[34] D. Zheng, G. Lazarou, and R. Hu. A stochastic model for short-lived tcp flows. In *Proc. of the IEEE Int. Conf. on Communications*, volume 1, pages 76–81, May 2003.

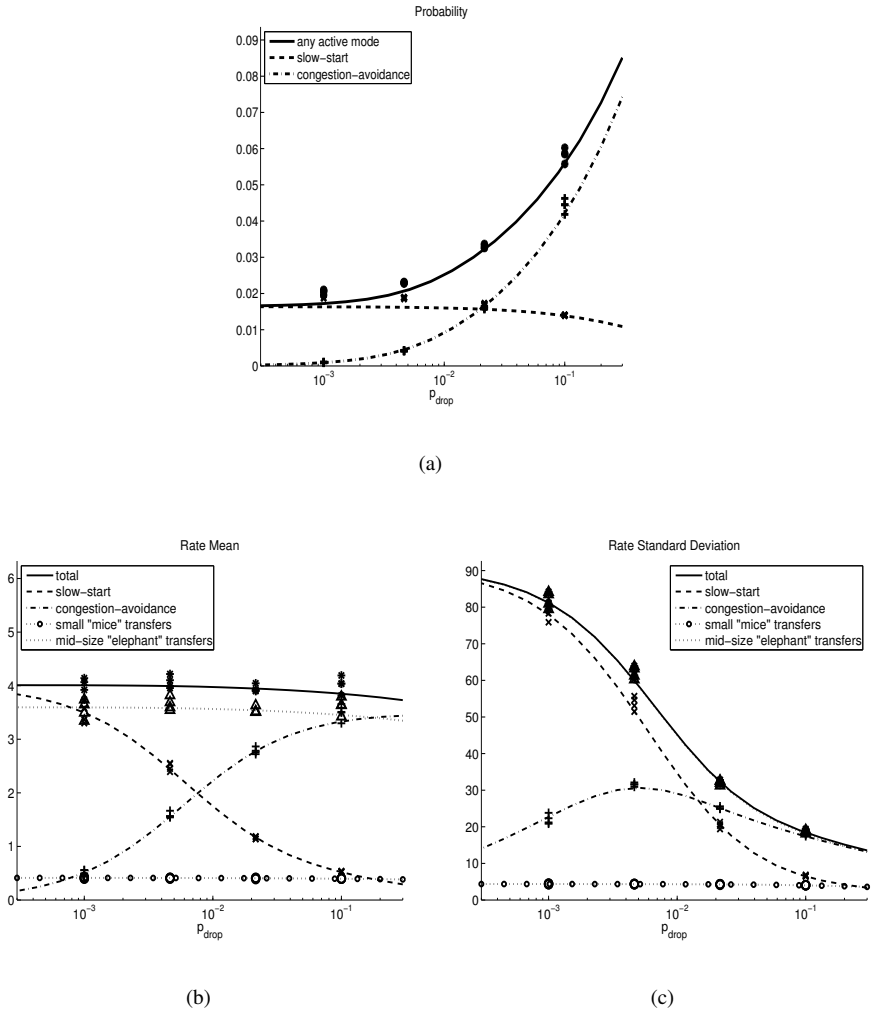


FIGURE 1.3: Steady-state values for the probability of a flow being on each mode (a) and the average (b) and standard deviation (c) of the sending rate as a function of the drop probability. The solid lines were obtained for the mixed-exponentials model (1.21) with $RTT = 50$ ms, $n_{\text{ack}} = 1$, and a transfer-size distribution that results from the mixture of two exponentials with the parameters in (1.22). The average off-period was set to $\tau_{\text{off}} = 5$ sec. The (larger) symbols were obtained from Monte Carlo simulations.

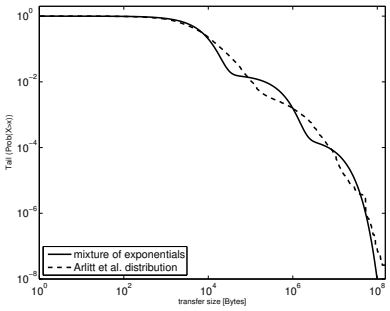


FIGURE 1.4: Complementary cumulative distribution function (ccdf) of transfer-sizes resulting from the mixture of three exponentials with the parameters in (1.23). This distribution was used in the simulations in Figure 1.5.

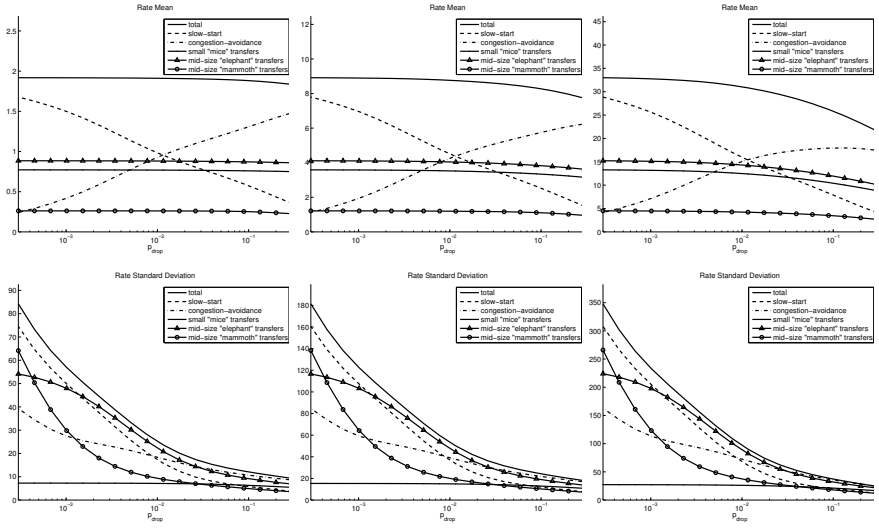


FIGURE 1.5: Steady-state values for the average (top) and standard deviation (bottom) of the sending rate as a function of the drop probability. These results were obtained from the mixed-exponentials model (1.21), with $RTT = 50$ ms, $n_{\text{ack}} = 1$, and a transfer-size distribution resulting from the mixture of three exponentials with the parameters in (1.23). The average off-period was set to $\tau_{\text{off}} = 5$ sec (left), 1 sec (middle), and 0.2 sec (right.)

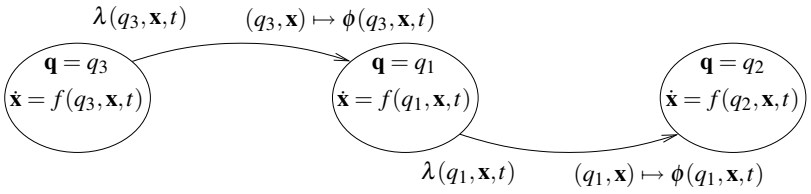


FIGURE 1.6: Graphical representation of a stochastic hybrid system.

Index

0-8493-0052-5/00/$0.00+$.50