Control of Communication Networks Using Infinitesimal Perturbation Analysis of Stochastic Fluid Models

Christos Panayiotou[1], Christos G. Cassandras[2], Gang Sun[3], and Yorai Wardi[4]

[1] Dept. of Electrical and Computer Engineering, University of Cyprus, Nicosia, Cyprus, christosp@ucy.ac.cy
[2] Dept. of Manufacturing Engineering and Center for Information and Systems Engineering, Boston University, Brookline, MA 02446, cgc@bu.edu
[3] Dept. of Manufacturing Engineering and Center for Information and Systems Engineering, Boston University, Brookline, MA 02446, gsun@bu.edu
[4] School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, wardi@ee.gatech.edu

1 Introduction

Managing and operating large scale communication networks is a challenging task and it is only expected to get worse as networks grow larger. The difficulties associated with network management stem from the fact that modeling and analysis of large scale communication networks is an excessively difficult task. On one hand, the enormous traffic volume in today's Internet makes packet-by-packet analysis infeasible. On the other hand, queueing systems (the natural modeling framework for packet-based communication networks) are largely based on Poisson processes and does not capture the bursty nature of realistic traffic. Moreover, the discovery of self-similar patterns in the Internet traffic distribution [1] and the resulting inadequacies of Poisson traffic models [2] further complicate queueing analysis. At the same time we need to account for the fact that the stochastic processes involved are time-varying, i.e., no stationarity assumptions hold. In addition, we need to explicitly model buffer overflow phenomena which typically defy tractable analytical derivations. Consequently, performance analysis techniques that do not depend on detailed traffic distributional information are highly desirable.

An alternative modeling paradigm based on fluid models has become increasingly attractive. The argument leading to the popularity of fluid models is that random phenomena may play different roles at different time scales. When the variations on the faster time scale have less impact than those on the slower time scale, the use of fluid models is justified. The efficiency of

a fluid model rests on its ability to aggregate multiple events. By ignoring the micro-dynamics of each discrete entity and focusing on the change of the aggregated flow rate instead, a fluid model allows the aggregation of events associated with the movement of multiple packets within a time period of a constant flow rate into a single rate change event. Introduced in [3] and later proposed in [4] for the analysis of multiplexed data streams and network performance [5], fluid models have been shown to be especially useful for simulating various kinds of high speed networks [6, 7, 8, 9]. A *Stochastic* Flow Model (SFM) has the extra feature that the flow rates are treated as general stochastic processes, which distinguishes itself from the approach adopted in [10, 11, 12] that deal with deterministic or Markov modulated fluid rates.

On the other hand, the fluid modeling paradigm forgoes the identity and dynamics of individual packets and focuses instead on the aggregate flow rate. As a result, this paradigm is more suitable for network-related measures, such as buffer levels and packet loss volumes, rather than packet-related measures such as sojourn times (although it is still possible to define fluid-based sojourn times [13]). A Quality of Service (QoS) metric that depends on the identity of certain packets, for example, cannot be obviously captured by a fluid model. Furthermore, for the purpose of performance analysis of networks with QoS requirements, the accuracy of SFMs depends on traffic conditions, the structure of the underlying system, and the nature of the performance metrics of interest. Moreover, some metrics may depend on higher-order statistics of the distributions of the underlying random variables involved, which a fluid model may not be able to accurately capture.

In this chapter, our goal is to explore the use of SFMs for the purpose of *control and optimization* rather than *performance analysis*. In this case, it is not unreasonable to expect that one can identify the solution of an optimization problem based on a model which captures only those features of the underlying "real" system that are needed to lead to the right solution, without the need to estimate the corresponding optimal performance with accuracy. Even if the exact solution cannot be obtained by such "lower-resolution" models, one can still obtain near-optimal points that exhibit robustness with respect to certain aspects of the model they are based on. Such observations have been made in several contexts (e.g., [14]), including recent results related to SFMs reported in [15] where a connection between the SFM and queueing-system-based solution is established for various optimization problems in queueing systems.

Using the SFM modeling framework, a new approach for network management is being developed which is based on Infinitesimal Perturbation Analysis (IPA) [16, 17, 18, 19] (IPA is covered in detail in [20, 21]). In this approach, we estimate the gradient of the performance measure of interest (e.g., packet loss rate) with respect to the control parameters of interest (e.g., buffer thresholds) and use them in standard stochastic approximation algorithms to determine the optimal parameter setting. This approach has some very important advantages.

- The gradient estimation is done *on-line* thus the approach can be implemented on the real system: as the operating conditions change, it will aim at *continuously* seeking to optimize a generally time-varying performance metric.
- The gradient estimation process does not require any knowledge of the system's underlying stochastic processes; in other words, it is model free.
- The estimators are shown to be unbiased when evaluated based on SFM sample paths[5].
- It turns out that the estimators consist only of accumulators and timers and are generally easy to implement.

It is also worth pointing out that, even though the estimators are derived based on a SFM, their simplicity allows us to evaluate them based on the sample paths of *discrete-event systems* . Furthermore, simulation results indicate that such an approach works nicely, although the SFM-based estimators evaluated based on discrete event sample paths may no longer be unbiased. On-line management is appealing in today's computer networks and will become even more important as high speed network technologies become popular. In such cases, huge amounts of resources may suddenly become available or unavailable. Since manually managing network resources has become unrealistic, it is critical for network components, i.e., routers and end hosts, to automatically adapt to rapidly changing conditions.

This chapter consists of a tutorial on some of the main results that have appeared in the literature of IPA of SFMs. Section 2 presents an overview of the general methodology employed when analyzing such systems. The subsequent sections present some specific results on different important models. Section 3 analyzes a single node with a single class of fluid. Section 4 extends the analysis to multiple classes of fluid, while Section 5 presents a series of nodes but with a single class of fluid. Subsequently, Section 6 presents some simulations examples and finally Section 7 concludes and presents plans for future directions.

2 General Methodology

The basic SFM, used in this chapter follows the ones described in [13, 16, 17, 19, 23] where the system is characterized by a number of stochastic processes, all defined on a common probability space $(\Omega, \mathcal{F}, P)$. In general, all stochastic processes are classified as *defining* or *derived*; *Defining processes* are all *external* inflow processes (typically denoted by $\{\alpha(t;\theta)\}$) and all service processes (denoted by $\{\beta(t;\theta)\}$) where θ is some controllable parameter. *Derived processes* are the ones that result from the defining processes, the system dynamics and the controllable parameters (θ); examples of such processes are the

[5] This is a desirable property that allows us to reliably use them with stochastic optimization algorithms e.g., [22].

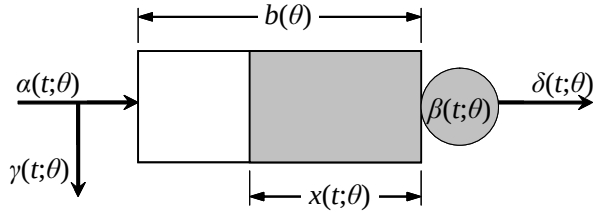

Fig. 1. Basic Stochastic Fluid Model (SFM)

buffer outflow (denoted by $\{\delta(t;\theta)\}$), buffer occupancy ($\{x(t;\theta)\}$) and fluid overflow ($\{\gamma(t;\theta)\}$). Examples of these processes are shown in Fig. 1 for the single node system. How the derived processes are derived from the defining processes is considered in detail in the following sections.

The main motivation behind this research is the optimization of some cost functions of the form

$$J(\theta;\mathbf{x}(0),T) = \mathbb{E}\left[\mathcal{L}(\theta;\mathbf{x}(0),T)\right]$$

where, θ constitutes the controllable parameter (possibly a vector of parameters) and $\mathcal{L}(\theta;\mathbf{x}(0),T)$ is some sample function of interest evaluated in the interval $[0,T]$ when the initial conditions are $\mathbf{x}(0)$. Loss volume, loss probability, average workload, and throughput are some of the cost functions that can be addressed in this approach. However, to limit the length of the chapter, in the sequel we only address the loss volume ($L(\theta;\mathbf{x}(0),T)$) and average workload ($Q(\theta;\mathbf{x}(0),T)$) which will be explicitly defined in the following sections. Note that from the workload metric it is possible to obtain a delay metric using appropriate forms of Little's law (e.g., see [13]). The general solution approach to the optimization problem above adopted in this chapter consists of three main steps which are briefly described in the following subsections.

2.1 Stochastic Approximation Algorithm

It is generally difficult (if at all possible) to obtain closed form expressions for $J(\theta;\mathbf{x}(0),T)$. Therefore, one needs to resort to iterative methods such as stochastic approximation algorithms (e.g., [22]) which are driven by estimates of the gradient of a cost function with respect to the parameters of interest. In the case of the cost minimization problem above, we are interested in estimating $dJ/d\theta$ based on directly observed (or simulated) data. We can then seek to obtain θ^* such that it minimizes $J(\theta;\mathbf{x}(0),T)$ through an iterative scheme of the form

$$\theta_{n+1} = \theta_n - \sigma_n H_n(\theta_n;\mathbf{x}(0),T,\omega_n^{SFM}), \quad n = 0,1,\cdots \tag{1}$$

where $H_n(\theta_n;\mathbf{x}(0),T,\omega_n^{SFM})$ is an estimate of $dJ/d\theta$ evaluated at $\theta = \theta_n$ and based on information obtained from a sample path of the SFM denoted

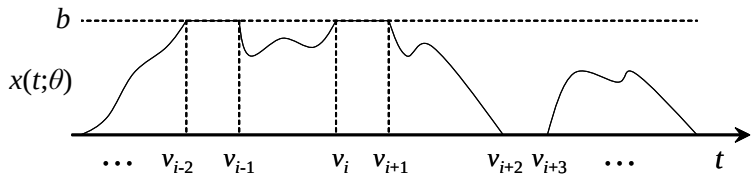

Fig. 2. Typical sample path of a single SFM node

by ω_n^{SFM}. Furthermore, $\{\sigma_n\}$ is an appropriate sequence of step sizes. To simplify the notation, in the sequel we assume that $\mathbf{x}(0) = \mathbf{0}$ and we will omit the initial condition, the observation interval T and the sample point $\omega \in \Omega$ unless it is necessary to stress the dependence. However, we emphasize that *all performance measures of interest are evaluated over a finite interval* $[0, T]$. Infinitesimal Perturbation Analysis (IPA) is used to obtain sample derivatives $d\mathcal{L}/d\theta$. These derivatives can be used in (1) if they are *unbiased* estimators of $dJ/d\theta$. The derivation of such estimators and their unbiasedness properties are addressed next.

2.2 IPA Derivative Estimates

In this section we show the general process for employing Infinitesimal Perturbation Analysis (IPA) [20, 21] to derive the sample derivatives $\mathcal{L}'(\theta) = d\mathcal{L}(\theta)/d\theta$. Before we present the general derivation approach, we define the notion of *events* along a sample path which indicate changes in either the defining or the derived processes. Of particular interest are what we refer to as *exogenous* and *endogenous* events (exogenous events refer to changes in the input (defining) processes and endogenous events refer to changes in the output (derived) processes). The precise definition of these events depends on the nature of the model under investigation. In general, however, for a given fixed $\theta \in \Theta$, an *exogenous event* coincides with a point where the *net inflow* (inflow minus outflow) to certain buffers changes sign either in a continuous fashion or due to a discontinuity of the defining processes ($\alpha(t;\theta)$ and $\beta(t;\theta)$). *Endogenous events* correspond to points where some buffer becomes either full or empty or points where the buffer content crosses certain thresholds. Fig. 2 shows a typical sample path of the buffer content of a single SFM node with buffer capacity b. The sequence $\{v_i : i = 1, 2, \cdots\}$ indicates examples of such events; v_{i-2} and v_{i+2} indicate the events *buffer becomes full* and *buffer becomes empty* respectively while, v_{i+1} and v_{i+3} indicate the events *buffer ceases to be full* and *buffer ceases to be empty* respectively. The latter two are exogenous events since they occur due to changes in the input processes, while the first two are endogenous events since their occurrence depends on the system dynamics (the detailed dynamics will be given in (2)).

Corollary 1. *Exogenous events are independent of the control parameters θ, in other words, the event time derivatives of exogenous events are equal to zero.*

The performance measures of interest are generally functions of the derived processes. Thus, obtaining sample derivatives involves the differentiation of these processes with respect to the control parameter θ which can be done if the sample paths are segmented into smaller intervals. It turns out that a "convenient" segmentation is to divide the sample path at points where some of the exogenous or endogenous events occur. For example, in Fig. 2 the segments $s_i = [v_i, v_{i+1})$, $i = 0, \cdots, N_T$ constitute such a sample path segmentation where in the observation interval $[0, T]$ there are $N_T < \infty$ (w.p. 1) such segments. Such segmentation makes the sample function differentiation easier, but the result is generally an iterative algorithm that determines the required sample derivatives during segment s_i given the sample derivatives of segment s_{i-1}. These iterative algorithms are usually sufficient to numerically evaluate the derivatives of the sample functions of interest. However, we point out that for certain systems it is possible to derive closed-form expressions for these derivatives (examples of such derivations will be given in the following sections where specific models will be investigated).

Once the sample function derivatives $d\mathcal{L}/d\theta$ are obtained we need to investigate whether they are unbiased estimates of the required $dJ/d\theta$ so they can be used in (1). This is done in the next section.

2.3 Unbiasedness

In this section we address the *unbiasedness* properties of the IPA estimators obtained above. As IPA estimator is *unbiased* if the following holds

$$\frac{dJ(\theta)}{d\theta} = \frac{d\mathbb{E}\left[\mathcal{L}(\theta)\right]}{d\theta} = \mathbb{E}\left[\frac{d\mathcal{L}(\theta)}{d\theta}\right] = \mathbb{E}\left[\mathcal{L}'(\theta)\right].$$

As mentioned earlier, this result allows us to use the derived IPA estimates in the stochastic approximation algorithm (1). In general, the unbiasedness of an IPA derivative $\mathcal{L}'(\theta)$ has been shown to be ensured by the following two conditions (see [24], Lemma A2, p.70):

Condition 1.

a. For every $\theta \in \Theta$ (where Θ is a closed bounded set), the sample derivative $\mathcal{L}'(\theta)$ exists w.p.1.

b. W.p.1, the random function $\mathcal{L}(\theta)$ is Lipschitz continuous throughout Θ, and the (generally random) Lipschitz constant has a finite first moment.

The existence of the sample derivatives studied in this chapter is guaranteed by Assumption 1 shown below.

Assumption 1.

a. W.p.1, all *defining processes* (e.g., arrival and service rate functions $\alpha(t) \geq 0$ and $\beta(t) \geq 0$) are piecewise analytic in the interval $[0, T]$.
b. For every $\theta \in \Theta$, w.p. 1, two events cannot occur at exactly the same time. An exception is allowed for pairs of events such that the occurrence of one forces the immediate occurrence of the other.
c. W.p.1, no two processes $\{\alpha(t)\}$ or $\{\beta(t)\}$, have identical values during any open subinterval of $[0, T]$.

All three parts of **Assumption 1** are mild technical conditions which hold for all problems considered in the sequel. Regarding parts *b* and *c*, we point out that even if they do not hold, it is possible to use one-sided derivatives and still carry out similar analysis. However, in order to keep the analysis and notation manageable we impose these conditions.

Consequently, establishing the unbiasedness of $\mathcal{L}'(\theta)$, reduces to verifying the Lipschitz continuity of the sample function $\mathcal{L}(\theta)$ with appropriate Lipschitz constants. For all estimators presented in the sequel this has been established. The unbiasedness proofs are rather tedious and the interested reader is referred to the appropriate reference for the details. Next, we apply the three steps briefly described above for controlling system parameters such as the buffer thresholds and the server processing rates for different systems.

2.4 More Notation and Preliminaries

In this section we define some of the concepts and quantities that we will use in the sequel.

Boundary and Non-Boundary Periods (B_k, $\bar{B}_k$)

These define a possible partition of a sample path. Boundary periods are maximal intervals where the buffer level $x(t;\theta)$ is constant and equal to some boundary or some threshold (i.e., $dx(t;\theta)/dt = 0$). Equivalently, non-boundary periods are intervals such that $x(t;\theta)$ is not on a boundary. In Fig. 2, $[v_i, v_{i+1})$ and $[v_{i+2}, v_{i+3})$ are boundary intervals while $[v_{i-1}, v_i)$ and $[v_{i+1}, v_{i+2})$ are non-boundary intervals. Also, with N_B and $N_{\bar{B}}$ we denote the random number of boundary and non-boundary periods respectively observed in the interval $[0, T]$.

Resetting Cycle ($\mathcal{C}_k$)

A non-boundary period followed by a boundary period forms a *resetting cycle* where the evaluation of event time derivatives is independent of the past history. However, one should not confuse the concept of *resetting cycle* with that of *regenerating cycle* because the evolution of the stochastic process itself might not always be independent of its past history. Furthermore, the kth resetting cycle $\mathcal{C}_k$ includes $R_k + 1$ events, that is $\mathcal{C}_k = [v_{k,0}, v_{k,R_k})$, where $v_{k,j}$

correspond to v_i defined earlier but re-indexed based on the resetting cycle they belong to. In Fig. 2 the intervals $[v_{i-1}, v_{i+1})$ and $[v_{i+1}, v_{i+3})$ correspond to resetting cycles each of which includes 3 events. In addition, with $N_{\mathcal{C}}$ we denote the random number of resetting cycles in the interval $[0, T]$.

Empty and Non-Empty Periods ($\mathcal{E}_k$, $\bar{\mathcal{E}}_k$)

Another sample path partitioning due to queueing theory is into *busy* and *idle* periods. Busy periods are periods where a buffer is not empty and idle otherwise. When using SFMs however, it is possible to have an empty buffer, while the server is not idle (e.g., when the inflow is less than the maximum outflow), so, we prefer to use the more appropriate terms of *empty* and *non-empty* periods. In Fig. 2 the interval $[v_{i-3}, v_{i+2})$ corresponds to a non-empty period while the interval $[v_{i+2}, v_{i+3})$ corresponds to an empty period. The kth non-empty period includes $S_k + 1$ events, thus $\bar{\mathcal{E}}_k = [v_{k,0}, v_{k,S_k})$, where again $v_{k,j}$ are re-indexed based on the non-empty period they belong to. Also, with $N_{\mathcal{E}}$ and $N_{\bar{\mathcal{E}}}$ we denote the random number of empty and non-empty periods respectively observed in the interval $[0, T]$.

The Prime Notation ($\cdot'$)

In the sequel we use the prime notation to indicate the derivative with respect to the control parameter of interest (typically either θ or ρ). For example, $x'(t;\theta)$ indicates the derivative of $x(t;\theta)$ with respect to θ and v_i' indicates the derivative of the event time v_i again with respect to θ.

3 Single-Class Single-Node System

In this section we investigate the single-class single-node system shown in Fig 1. We assume that the system inflow is $\alpha(t)$, the maximum outflow is $\rho\beta(t)$ and the buffer size is θ where, ρ and θ are the controllable parameters of interest. The parameter $\rho \in [0, 1]$ denotes the proportion of the server capacity (i.e., $\beta(t)$) allocated to the specific queue by the resource scheduler. The processes $\alpha(t)$ and $\beta(t)$ are independent of both parameters θ and ρ. The system dynamics are given by

$$\frac{dx(t;\theta,\rho)}{dt^+} = \begin{cases} 0 & \text{if } x(t;\theta,\rho) = 0 \text{ and } \alpha(t) - \rho\beta(t) < 0 \\ 0 & \text{if } x(t;\theta,\rho) = \theta \text{ and } \alpha(t) - \rho\beta(t) > 0 \\ \alpha(t) - \rho\beta(t) & \text{otherwise} \end{cases} \tag{2}$$

The performance measures of interest are the average workload $Q(\theta, \rho)$ and the loss volume $L(\theta, \rho)$ defined in (3) and (4) respectively.

$$Q(\theta,\rho) = \int_0^T x(t;\theta,\rho)dt \tag{3}$$

$$L(\theta,\rho) = \int_0^T \gamma(t;\theta,\rho)dt \tag{4}$$

Next we derive the sample function derivatives with respect to the parameters θ and ρ. The IPA derivation can be done using either the resetting cycles or empty and non-empty periods. In [16, 19] the derivation is done using empty and non-empty periods, so here we present an alternative analysis based on the resetting cycles. For this system, any resetting cycle (say the kth one) consists of two periods $[v_{k,0}, v_{k,1})$ where the system will be in a non-boundary period and $[v_{k,1}, v_{k,2})$ where the system will be in a boundary period (i.e., $R_k = 2$ for all k). Using this sample path partitioning, we can rewrite the objectives as:

$$Q(\theta,\rho) = \sum_{k=1}^{N_{\mathcal{C}}} q_k(\theta,\rho) = \sum_{k=1}^{N_{\mathcal{C}}} \int_{v_{k,0}}^{v_{k,2}} x(t;\theta,\rho)dt \tag{5}$$

$$L(\theta,\rho) = \sum_{k=1}^{N_{\mathcal{C}}} \int_{v_{k,1}}^{v_{k,2}} \gamma(t;\theta,\rho)dt \tag{6}$$

where, as mentioned earlier, $N_{\mathcal{C}}$ is the random number of such cycles that appear in the interval $[0,T]$ and $q_k(\cdot) = \int_{v_{k,0}}^{v_{k,2}} x(\cdot)dt$, $k = 0, 1, \cdots$.

3.1 IPA Derivative with respect to θ

In this section we assume $\rho = 1$ constant (so it is omitted from all expressions), and derive $Q'(\theta) = dQ(\theta)/d\theta$ and $L'(\theta) = dL(\theta)/d\theta$. Differentiating (5) with respect to θ we obtain

$$Q'(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} q'_k(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} \int_{v_{k,0}}^{v_{k,2}} x'(t;\theta)dt \tag{7}$$

where we used the fact that cycle beginning and ending points v_0 and v_2 respectively are independent of θ since they correspond to exogenous events and thus $v'_0 = v'_2 = 0$ (Corollary 1). The loss volume derivative is given by

$$L'(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} \left[-\gamma(v_{k,1};\theta)v'_{k,1} + \int_{v_{k,1}}^{v_{k,2}} \gamma'(t;\theta)dt \right].$$

Furthermore, the loss rate during any boundary period is

$$\gamma(t;\theta) = \begin{cases} 0 & \text{if } x(t;\theta) = 0 \\ \alpha(t) - \beta(t) & \text{if } x(t;\theta) = \theta \end{cases}, \; t \in [v_{k,1}, v_{k,2}), \; k = 1, \cdots, N_{\mathcal{C}}.$$

Both cases are independent of θ, therefore, $\gamma'(t;\theta) = 0$ and thus the above derivative simplifies to

$$L'(\theta) = -\sum_{k=1}^{Nc} \gamma(v_{k,1};\theta) v'_{k,1} \tag{8}$$

Next, we derive $q'_k(\theta)$ and $v'_{1,k}$ by analyzing each cycle independently. Before we proceed with the four possible cases, we recognize that during a non-boundary period the buffer content goes from one boundary to another. Therefore,

$$\theta \mathbf{1}[x(v_{k,0}) = \theta] + \int_{v_{k,0}}^{v_{k,1}} (\alpha(t) - \beta(t))\, dt = \theta \mathbf{1}[x(v_{k,1}) = \theta]$$

Differentiating both sides we get

$$(\alpha(v_{k,1}) - \beta(v_{k,1}))\, v'_{k,1} = \mathbf{1}[x(v_{k,1}) = \theta] - \mathbf{1}[x(v_{k,0}) = \theta] \tag{9}$$

Now, depending on the boundary where the cycle starts and ends (empty (E) or full (F)), we identify four possible cases.

Case EE: *The cycle starts and ends with an empty period.* In this case,

$$x(t;\theta) = \begin{cases} \int_{v_{k,0}}^{t} (\alpha(t) - \beta(t))\, dt & \text{for } t \in [v_{k,0}, v_{k,1}) \\ 0 & \text{for } t \in [v_{k,1}, v_{k,2}) \end{cases}$$

and $x'(t;\theta) = 0$ for all $t \in \mathcal{C}$. Also, $\gamma(t;\theta) = 0$ for all $t \in \mathcal{C}$ (no loss during the cycle), thus $\gamma(v_{k,1};\theta) = 0$. As a result

$$q'_k(\theta) = 0 \quad \text{and} \quad \gamma(v_{k,1};\theta) v'_{k,1} = 0 \tag{10}$$

Case EF: *The cycle starts with an empty and ends with a full period.* Thus,

$$x(t;\theta) = \begin{cases} \int_{v_{k,0}}^{t} (\alpha(t) - \beta(t))\, dt & \text{for } t \in [v_{k,0}, v_{k,1}) \\ \theta & \text{for } t \in [v_{k,1}, v_{k,2}) \end{cases}.$$

Differentiating with respect to θ we get,

$$x'(t;\theta) = \begin{cases} 0, & \text{for } t \in [v_{k,0}, v_{k,1}) \\ 1, & \text{for } t \in [v_{k,1}, v_{k,2}) \end{cases}.$$

Substituting into (7) we get, $q'_k(\theta) = v_{k,2} - v_{k,1}$. For the loss volume, $\gamma(v_{k,1},\theta) = \alpha(v_{k,1}) - \beta(v_{k,1})$ and, from (9), we get that $(\alpha(v_{k,1}) - \beta(v_{k,1}))v'_{k,1} = 1$, therefore

$$q'_k(\theta) = v_{k,2} - v_{k,1} \quad \text{and} \quad \gamma(v_{k,1};\theta) v'_{k,1} = 1. \tag{11}$$

Case FF: *The cycle starts and ends with a full period.* In this case,

$$x(t;\theta) = \begin{cases} \theta + \int_{v_{k,0}}^{t} (\alpha(t) - \beta(t))\, dt & \text{for } t \in [v_{k,0}, v_{k,1}) \\ \theta & \text{for } t \in [v_{k,1}, v_{k,2}) \end{cases}$$

therefore, $x'(t;\theta) = 1$ for the entire cycle. As a result, from (7), $q'_k(\theta) = v_{k,2} - v_{k,0}$. In addition, from (9) we get that $v'_{k,1} = 0$ and therefore

$$q'_k(\theta) = v_{k,2} - v_{k,0} \quad \text{and} \quad \gamma(v_{k,1};\theta)v'_{k,1} = 0. \tag{12}$$

Case FE: *The cycle starts with a full and ends with an empty period.* In this case,

$$x(t;\theta) = \begin{cases} \theta + \int_{v_{k,0}}^{t} (\alpha(t) - \beta(t))\, dt & \text{for } t \in [v_{k,0}, v_{k,1}) \\ 0 & \text{for } t \in [v_{k,1}, v_{k,2}) \end{cases}$$

and therefore $x'(t;\theta) = 1$ for $t \in [v_{k,0}, v_{k,1})$ and 0 for $t \in [v_{k,1}, v_{k,2})$. As a result, from (7) $q'_k(\theta) = v_{k,1} - v_{k,0}$. For the loss volume, $x(v_{k,1};\theta) = 0$ therefore $\gamma(v_{k,1};\theta) = 0$. Hence,

$$q'_k(\theta) = v_{k,1} - v_{k,0} \quad \text{and} \quad \gamma(v_{k,1};\theta)v'_{k,1} = 0. \tag{13}$$

Theorem 1. *The sample derivatives $Q'(\theta)$ and $L'(\theta)$ with respect to θ are*

$$Q'(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} [(v_{k,2} - v_{k,1})\mathbf{1}_{EF} + (v_{k,2} - v_{k,0})\mathbf{1}_{FF} + (v_{k,1} - v_{k,0})\mathbf{1}_{FE}]$$
$$L'(\theta) = -N_{\mathcal{C}_{EF}}$$

where $N_{\mathcal{C}_{EF}}$ is the number of EF cycles that were observed during $[0, T]$ and $\mathbf{1}_{yy}$ is the usual indicator function that takes the value of 1 if the cycle is of the yy type and 0 otherwise.

Proof: Follows immediately from (10)-(13). ∎

Corollary 2. *The estimators of Theorem 1 are precisely the estimators obtained in [16], in other words,*

$$Q'(\theta) = \sum_{j=1}^{N_{\bar{\mathcal{E}}}} (v_{j,S_j} - v_{j,1}) \quad \textit{and} \quad L'(\theta) = -N_{\mathcal{C}_{EF}}$$

where j counts the number of non-empty periods (not cycles), $v_{j,1}$ indicates the first overflow point of the jth non-empty period and v_{j,S_j} indicates the end of the jth non-empty period. If no overflow occurs $v_{j,1} = v_{j,S_j}$.

Proof: Follows by recognizing that the union of a non-empty period with the following empty period consists either of a single EE cycle or it starts with an EF cycle, followed by m FF cycles $m = 0, 1, 2, \cdots$, and ends with an FE

cycle. Furthermore, note that the number of non-empty periods with at least some loss is equal to the number of EF cycles $N_{\mathcal{C}_{EF}}$. ∎

In [16] it is also shown that the above estimators are *unbiased*. Furthermore, we point out that the implementation of the above estimators is extremely simple; they simply accumulate the time between certain events, or they count the number of EF cycles.

3.2 IPA Derivative with respect to ρ

In this section, we assume $\theta = b$ constant (so it is omitted from all expressions), and derive $Q'(\rho) = dQ(\rho)/d\rho$ and $L'(\rho) = dL(\rho)/d\rho$. Differentiating (5) with respect to ρ we obtain

$$Q'(\rho) = \sum_{k=1}^{N_{\mathcal{C}}} q'_k(\rho) = \sum_{k=1}^{N_{\mathcal{C}}} \left[x(v_{k,2};\rho)v'_{k,2} - x(v_{k,0};\rho)v'_{k,0} + \int_{v_{k,0}}^{v_{k,2}} x'(t;\rho)dt \right] \quad (14)$$

In this case, the cycle beginning and ending points $v_{k,0}$ and $v_{k,2}$ respectively are not always due to exogenous events. The events *buffer ceases to be empty or full* occur at points where a sign change of $\alpha(t) - \rho\beta(t)$ occurs. The sign change can be due to a jump (discontinuity) in either $\alpha(t)$ or $\beta(t)$ which corresponds to exogenous events, or it can occur in a continuous fashion, thus the switching time depends on ρ. Nevertheless, the following result holds.

Lemma 1. *At the cycle beginning and ending points $v_{k,0}$ and $v_{k,2}$ respectively,*

$$(\alpha(v_{k,0}) - \rho\beta(v_{k,0}))\, v'_{k,0} = (\alpha(v_{k,2}) - \rho\beta(v_{k,2}))\, v'_{k,2} = 0$$

Proof: We make the argument only for the beginning point of $\mathcal{C}_k$; the argument for the ending point is the same since it coincides with the beginning point of $\mathcal{C}_{k+1}$. As mentioned above, $v_{k,0}$ is due to a sign change of $\alpha(t) - \rho\beta(t)$ from positive to negative or vise versa. This can happen either due to a discontinuity in the processes $\alpha(t)$ and $\beta(t)$ or it can happen in a continuous fashion. If this happens due to a discontinuity of either $\alpha(t)$ or $\beta(t)$, then it corresponds to an exogenous event, thus $v'_{k,0} = 0$. If on the other hand, the sign change occurs in a continuous fashion, $\alpha(v_{k,0}) - \rho\beta(v_{k,0}) = 0$. Hence, the lemma is proved. ∎

Next, note that the buffer content is given by

$$x(t;\rho) = \begin{cases} b\mathbf{1}[x(v_{k,0};\rho) = b] + \int_{v_{k,0}}^{t} (\alpha(t) - \rho\beta(t))\, dt & \text{for } t \in [v_{k,0}, v_{k,1}) \\ b\mathbf{1}[x(v_{k,1};\rho) = b] & \text{for } t \in [v_{k,1}, v_{k,2}) \end{cases} \quad (15)$$

therefore, using Lemma 1, the derivative with respect to ρ is given by

$$x'(t;\rho) = \begin{cases} -\bar{\beta}(t, v_{k,0}) & \text{for } t \in [v_{k,0}, v_{k,1}) \\ 0 & \text{for } t \in [v_{k,1}, v_{k,2}) \end{cases} . \quad (16)$$

where

$$\bar{\beta}(t_2, t_1) = \int_{t_1}^{t_2} \beta(t)dt.$$

For the general case, the above integral can be evaluated numerically, however, to simplify the analysis, in the sequel we assume the $\beta(t) = \bar{\beta}$ constant, and thus $\bar{\beta}(t_2, t_1) = \bar{\beta}(t_2 - t_1)$. Therefore, the last term of (14) simplifies to

$$\begin{aligned}\int_{v_{k,0}}^{v_{k,2}} x'(t;\rho)dt &= -\int_{v_{k,0}}^{v_{k,1}} \bar{\beta}(t - v_{k,0})dt \\ &= -\frac{\bar{\beta}}{2}\left(v_{k,1}^2 - 2v_{k,1}v_{k,0} - v_{k,0}^2 + 2v_{k,0}^2\right) \\ &= -\frac{\bar{\beta}}{2}\left(v_{k,1} - v_{k,0}\right)^2.\end{aligned}$$

Substituting in (14), each term of the workload derivative simplifies to

$$q'_k(\rho) = x(v_{k,2};\rho)v'_{k,2} - x(v_{k,0};\rho)v'_{k,0} - \frac{\bar{\beta}}{2}\left(v_{k,1} - v_{k,0}\right)^2 \tag{17}$$

Similarly, the loss volume derivative is given by

$$L'(\rho) = \sum_{k=1}^{N_{\mathcal{C}}} \left[\gamma(v_{k,2};\rho)v'_{k,2} - \gamma(v_{k,1};\rho)v'_{k,1} + \int_{v_{k,1}}^{v_{k,2}} \gamma'(t;\rho)dt\right].$$

The loss rate during any boundary period is

$$\gamma(t;\rho) = \begin{cases} 0 & \text{if } x(t;\rho) = 0 \\ \alpha(t) - \rho\beta(t) & \text{if } x(t;\rho) = b \end{cases}, \; t \in [v_{k,1}, v_{k,2}), \; k = 1, \cdots, N_{\mathcal{C}}. \tag{18}$$

Therefore, $\gamma'(t;\rho) = -\beta(t)$ if $x(v_{k,1};\rho) = b$ and 0 otherwise. Also, from Lemma 1, the term $\gamma(v_{k,2};\rho)v'_{k,2} = (\alpha(v_{k,2}) - \rho\beta(v_{k,2}))v'_{k,2} = 0$. As a result, the above derivative simplifies to

$$L'(\rho) = \sum_{k=1}^{N_{\mathcal{C}}} \lambda'_k(\rho) = -\sum_{k=1}^{N_{\mathcal{C}}} \left[\gamma(v_{k,1};\rho)v'_{k,1} + \bar{\beta}(v_{k,2} - v_{k,1})\mathbf{1}[x(v_{k,1};\rho) = b]\right] \tag{19}$$

Next, we focus on a single cycle and derive $q'_k(\rho)$ and $\lambda'_k(\rho)$, but first we recognize that during any non-boundary period $x(t;\rho)$ goes from one boundary to another, thus

$$b\mathbf{1}[x(v_{k,0};\rho) = b] + \int_{v_{k,0}}^{v_{k,1}} \left(\alpha(t) - \rho\bar{\beta}\right) dt = b\mathbf{1}[x(v_{k,1};\rho) = b].$$

Differentiating both sides with respect to ρ, and using Lemma 1, we get

$$\left(\alpha(v_{k,1}) - \rho\bar{\beta}\right) v'_{k,1} - \bar{\beta}\left(v_{k,1} - v_{k,0}\right) = 0 \tag{20}$$

Depending on the boundary where the cycle starts and ends (empty (E) or full (F)), we identify four possible cases.

Case EE: *The cycle starts and ends with an empty period.* In this case, $x(v_{k,0};\rho) = x(v_{k,2};\rho) = 0$, and, from (17), $q'_k(\rho) = -\bar{\beta}\left(v_{k,1} - v_{k,0}\right)^2/2$. Also, $\gamma(t;\rho) = 0$ for all $t \in \mathcal{C}_k$ (no loss during the cycle), thus, from (19) $\lambda'_k(\rho) = 0$. Summarizing,

$$q'_k(\rho) = -\frac{\bar{\beta}}{2}\left(v_{k,1} - v_{k,0}\right)^2 \quad \text{and} \quad \lambda'_k(\rho) = 0 \tag{21}$$

Case EF: *The cycle starts with an empty and ends with a full period.* In this case, $x(v_{k,0},\rho) = 0$ and $x(v_{k,2},\rho) = b$ so, in (17) the term $x(v_{k,0},\rho)v'_{k,0} = 0$. Next, from (20) we get,

$$\gamma(v_{k,1};\rho)v'_{k,1} = \left(\alpha(v_{k,1}) - \rho\bar{\beta}\right)v'_{k,1} = \bar{\beta}\left(v_{k,1} - v_{k,0}\right).$$

Substituting in (17) and (19), we get

$$q'_k(\rho) = bv'_{k,2} - \frac{\bar{\beta}}{2}\left(v_{k,1} - v_{k,0}\right)^2 \quad \text{and} \quad \lambda'_k(\rho) = -\bar{\beta}(v_{k,2} - v_{k,0}) \tag{22}$$

Case FF: *The cycle starts and ends with a full period.* In this case, $x(v_{k,0},\rho) = b$ and $x(v_{k,2},\rho) = b$. Using (20) we get

$$\gamma(v_{k,1};\rho)v'_{k,1} = \left(\alpha(v_{k,1}) - \rho\bar{\beta}\right)v'_{k,1} = \bar{\beta}\left(v_{k,1} - v_{k,0}\right),$$

and again, substituting in (17) and (19), we get

$$q'_k(\rho) = bv'_{k,2} - bv'_{k,0} - \frac{\bar{\beta}}{2}\left(v_{k,1} - v_{k,0}\right)^2 \quad \text{and} \quad \lambda'_k(\rho) = -\bar{\beta}(v_{k,2} - v_{k,0}) \tag{23}$$

Case FE: *The cycle starts with a full and ends with an empty period.* In this case, $x(v_{k,0},\rho) = b$ and $x(v_{k,1},\rho) = x(v_{k,2},\rho) = 0$, so that in (17) the term $x(v_{k,2},\rho)v'_{k,2} = 0$. Also, from (20) we arrive at

$$\gamma(v_{k,1};\rho)v'_{k,1} = \left(\alpha(v_{k,1}) - \rho\bar{\beta}\right)v'_{k,1} = \bar{\beta}\left(v_{k,1} - v_{k,0}\right),$$

and again, substituting in (17) and (19), we get

$$q'_k(\rho) = -bv'_{k,0} - \frac{\bar{\beta}}{2}\left(v_{k,1} - v_{k,0}\right)^2 \quad \text{and} \quad \lambda'_k(\rho) = -\bar{\beta}(v_{k,1} - v_{k,0}) \tag{24}$$

Theorem 2. *The sample derivatives of $Q(\rho)$ and $L(\rho)$ are given by*

$$Q'(\rho) = -\frac{\bar{\beta}}{2}\sum_{k=1}^{N_{\mathcal{C}}}\left(v_{k,1} - v_{k,0}\right)^2$$

$$L'(\rho) = -\sum_{k=1}^{N_{\mathcal{C}}}\left[(v_{k,2} - v_{k,0})(\mathbf{1}_{EF} + \mathbf{1}_{FF}) + (v_{k,1} - v_{k,0})\mathbf{1}_{FE}\right]$$

Proof: The theorem follows by combining (21)-(24). For $Q'(\rho)$ note also that $v_{k,2} = v_{k+1,0}$ and that every EF cycle is followed by either an FF or an FE cycle and every FF is followed by another FF or FE cycle. As a result all terms of the form $bv'_{k,2}$ are cancelled by $bv'_{k+1,0}$. ∎

Corollary 3. *Theorem 2 gives precisely the estimators obtained in [19].*

We do not present the estimators explicitly since they require the introduction of some new notation. In any case, the proof of this corollary follows along the lines of the proof of Corollary 2. In [19] it is also shown that the above estimators are *unbiased*. Furthermore, we point out that the implementation of the above estimators is extremely simple; they simply accumulate the time between certain events. Finally, before we leave the single-node single-class flow model we mention that [19] also derives IPA estimators with respect to parameters of the arrival process $\alpha(t;\theta)$.

4 Multiple Classes

In this section, we enrich the modeling framework introduced earlier by introducing multiple classes of flows which are merged into a First Come First Serve (FIFO) buffer. The various classes of flows are differentiated according to a *Threshold Policy* [25] which works as follows: When fluid from class m arrives, it is accepted in the buffer if the state $x(t;\theta) < T_m$, otherwise the fluid is rejected. Threshold T_m, $m,\ m = 1, ..., M$, is associated with the class m flow and we assume $0 = T_0 < T_1 < \cdots < T_m < \cdots < T_M$. In this way, class M has the highest priority and class 1 the lowest priority. This policy has been shown to provide good protection to the higher priority classes [25] and was proposed for the Differentiated Services (DS) architecture [26]. The inflow rate of class m at time t is denoted by $\alpha_m(t)$ and the corresponding loss rate by $\gamma_m(t;\theta)$. As in the previous section, the service rate is denoted by $\beta(t)$, and $x(t;\theta)$ is the buffer level at time t. For the purpose of our analysis, we choose any one of the thresholds, say, $T_{\overline{m}}$, as the one with respect to which we wish to carry out sensitivity analysis and denote this parameter by θ. We also assume that the processes $\{\alpha_m(t)\},\ m = 1, ..., M$, and $\{\beta(t)\}$ are independent of θ and **Assumption 1** holds. For notational economy we define:

$$A_m(t) \equiv \sum_{n=m}^{M} \alpha_n(t) - \beta(t), \quad m = 1, \cdots, M \tag{25}$$

and observe that $A_m(t) \geq A_{m+1}(t)$, $m = 1, ..., M-1$.

Fig. 3 depicts the SFM described above. Furthermore, we assume that the parameter θ is confined to a bounded (compact) interval $\Theta = (T_{\overline{m}-1}, T_{\overline{m}+1})$. In what follows, again we consider two performance metrics, the *Cumulative Workload*(or just *Work*) $Q(\theta)$ and the mth class *Loss Volume* $L_m(\theta)$, $m = 1, ..., M$ defined as follows:

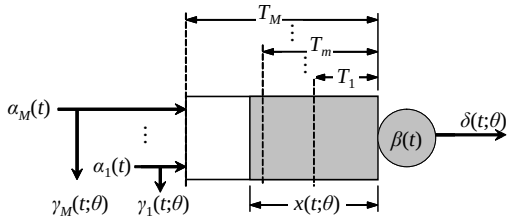

Fig. 3. M-Class Stochastic Fluid Model (SFM)

$$Q(\theta) = \int_0^T x(t;\theta)dt, \qquad L_m(\theta) = \int_0^T \gamma_m(t;\theta)dt. \tag{26}$$

For this system, we identify the following two event types. Event type e_1 is an event where the buffer content *leaves* the value $x(t;\theta) = T_m$, for some $m = 0, \cdots, M$, after it has maintained it for some finite length of time. This is an exogenous event since it is caused by a sign change of $A_m(t)$ in (25) for some $m = 1, ..., M$. Event type e_2 is defined to occur whenever the buffer content *reaches* or *crosses* the value $x(t;\theta) = T_m$, for any $m = 0, \cdots, M$. The interval between two consecutive events of type e_1 define a cycle (say $\mathcal{C}_k$). These events are assumed to occur at time instants $v_{k,0}$, $k = 1, \cdots, N_{\mathcal{C}}$ where $N_{\mathcal{C}}$ is the random number of such cycles in the interval $[0, T]$. During the kth cycle $\mathcal{C}_k$, we observe $R_k - 1$ endogenous (e_2) events, at time instants $v_{k,i}$ $i = 1, \cdots, R_k - 1$. (see Fig. 4 for examples of such events). Thus, the $\mathcal{C}_k$ cycle is divided into R_k periods

$$p_{k,i} \equiv [v_{k,i}, v_{k,i+1}), \quad i = 0, \cdots, R_{k-1}.$$

The corresponding open interval $(v_{k,i}, v_{k,i+1})$ is denoted by $p^o_{k,i}$. For the purpose of our analysis, we view each threshold as a *boundary*, thus any period $p_{k,i}$ where $x(t;\theta) = T_m$, $t \in p_{k,i}$, $m = 0, \cdots, M$, is considered as a *boundary period*. Otherwise, $p_{k,i}$ is a *non-boundary* period. We emphasize that each cycle ends with a boundary period p_{k,R_k-1} and thus the cycle definition is consistent with the definition of Section 2. A typical sample path for the case of $M = 3$ with three cycles $\mathcal{C}_i$, $\mathcal{C}_{i+1}$ and $\mathcal{C}_{i+2}$ is shown in Fig. 4. In the figure, note also that $v_{i,0} = v_{i-1,R_i}$, $i = 1, \cdots, N_{\mathcal{C}}$.

During a boundary period p_{k,R_k-1}, the buffer content dynamics are

$$\frac{dx(t;\theta)}{dt^+} = 0. \tag{27}$$

On the other hand, during a non-boundary period $p_{k,i}$, $i = 0, ..., R_k - 2$, if $T_{m-1} < x(t;\theta) < T_m$ for $t \in p^o_{k,i}$, the buffer content dynamics are

$$\frac{dx(t)}{dt^+} = A_m(t). \tag{28}$$

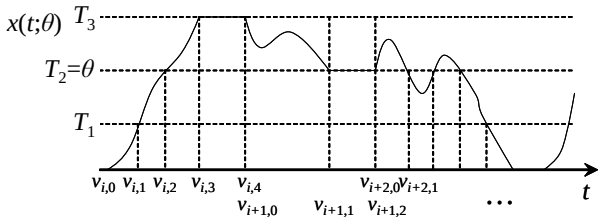

Fig. 4. Typical sample path segment ($M = 3$)

4.1 IPA with Respect to Thresholds

Our objective here is to estimate the sample derivatives $Q' = dQ(\theta)/d\theta$ and $L'_m = dL_m(\theta)/d\theta$, $m = 1, ..., M$. We proceed by first evaluating the sample derivatives $Q'(\theta)$ and $L'_m(\theta)$ in terms of event time derivatives $v'_{k,i} = dv_{k,i}/d\theta$, and then provide an algorithm for evaluating these event time derivatives based on observable quantities along a given sample path. Again for notational convenience, similar to our definition of $A_m(t)$ in (25), let us define

$$A_{m,k,i} \equiv \sum_{n=m}^{M} \alpha_n(v_{k,i}) - \beta(v_{k,i}) \tag{29}$$

Work IPA Derivative

Using the sample path partition into cycles, we write (3) as

$$Q(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} q_k(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} \int_{v_{k,0}}^{v_{k,R_k}} x(t;\theta)dt. \tag{30}$$

where, as earlier, $q_k(\theta) = \int_{v_{k,0}}^{v_{k,R_k}} x(t;\theta)dt$. Thus,

$$Q'(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} q'_k(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} \frac{d}{d\theta} \int_{v_{k,0}}^{v_{k,R_k}} x(t;\theta)dt = \sum_{k=1}^{N_{\mathcal{C}}} \int_{v_{k,0}}^{v_{k,R_k}} x'(t;\theta)dt \tag{31}$$

where we use the fact that $v_{k,0}$ and v_{k,R_k} are independent of θ since they correspond to exogenous events.

Theorem 3. *The sample derivative of $Q(\theta)$ with respect to θ is given by*

$$Q'(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} \sum_{j=0}^{R_k-1} x'_{k,j} \left(v_{k,j+1} - v_{k,j}\right)$$

where $x'_{k,R_k-1} = \mathbf{1}[x(v_{k,R_k-1};\theta) = \theta]$ and for $j = 0, \cdots, R_k - 2$

$$x'_{k,j} = \begin{cases} \mathbf{1}[x(v_{k,j};\theta) = \theta] - A_{m+1,k,j}v'_{k,j} & \text{if } \forall t \in p^o_{k,j}, T_m < x(t;\theta) < T_{m+1} \\ \mathbf{1}[x(v_{k,j};\theta) = \theta] - A_{m,k,j}v'_{k,j} & \text{if } \forall t \in p^o_{k,j}, T_{m-1} < x(t;\theta) < T_m \end{cases},$$

The proof follows easily from (31) by writing the expression of $x(t;\theta)$ in any interval $p_{k,j}$ and subsequently differentiating with respect to θ (see [23] for details). In [23] it is also shown that the estimator is unbiased. In order to evaluate $Q'(\theta)$ one simply needs to observe the net inflow rates $A_{m,k,i}$ at specific points in time and measure the intervals $(v_{k,j+1} - v_{k,j})$. In addition, one also needs the event time derivatives $v'_{k,j}$ which are determined in a subsequent subsection.

Class m Loss IPA Derivatives

Again, using the sample path partition into cycles, we may write $L_m(\theta)$ from (26) as follows:

$$L_m(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} \lambda_{m,k}(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} \int_{v_{k,0}}^{v_{k,R_k}} \gamma_m(t;\theta)dt \tag{32}$$

For the purpose of our analysis, a useful way of grouping periods $p_{k,i}$, $k = 1,\ldots,N_{\mathcal{C}}$ within a typical cycle is by defining sets associated with each class $m = 1,\cdots,M$ as follows:

Partial Loss Period Set U_m. For any $p_{k,i} \in U_m$, the buffer content is $x(t;\theta) = T_m$ for all $t \in p_{k,i}$, and class m traffic experiences partial loss. In particular, the traffic flows satisfy

$$A_m(t) > 0 \text{ and } A_{m+1}(t) < 0 \tag{33}$$

so that the processing capacity $\beta(t)$ can accommodate the cumulative incoming flow $\sum_{n=m+1}^{M} \alpha_n(t)$ due to classes $m+1,\cdots,M$, but not the flow $\sum_{n=m}^{M} \alpha_n(t)$ that includes the next lower priority class m. In this case, the system accepts only the portion of the class m traffic that can be accommodated and incurs a "partial" loss with rate $\gamma_m(t;\theta) = A_m(t)$. Formally, we define U_m as follows:

$$U_m := \{p_{k,i} : x(t;\theta) = T_m, \quad t \in p_{k,i}\}. \tag{34}$$

Note that the starting point $v_{k,i}$ of such a period corresponds to an endogenous event e_2, whereas the ending point $v_{k,i+1}$ corresponds to an exogenous event e_1 and is, therefore, locally independent of θ. Also note that the elements of U_m set are always the last interval of a cycle. In addition, the last interval of a cycle, p_{k,R_k-1}, if during it $x(t;\theta) \neq 0$, it must be an element of U_m for a some $m \in \{1,\cdots,M\}$.

Full Loss Period Set V_m. During such periods, the buffer content is $x(t;\theta) > T_m$ (excluding the starting point $v_{k,i}$) and *all* class m traffic is lost, i.e., $\gamma_m(t;\theta) = \alpha_m(t)$. Formally, we define V_m as follows:

$$V_m := \{p_{k,i} : x(t;\theta) > T_m, \quad t \in p^o_{k,i}\} \tag{35}$$

No Loss Period Set W_m. During such periods the buffer content is $x(t;\theta) < T_m$ (excluding the starting point $v_{k,i}$) and no class m loss occurs, i.e., $\gamma_m(t;\theta) = 0$. Formally, we define W_m as follows:

$$W_m := \{p_{k,i} : x(t) < T_m, \quad t \in p^o_{k,i}\} \tag{36}$$

Note that each of the sets above is locally independent of θ, and that for any m, $U_m \cup V_m \cup W_m = [0, T)$ with all sets being mutually exclusive.

Theorem 4. *The sample derivatives $L'_m(\theta)$, $m = 1, \cdots, M$ are given by*

$$L'_m(\theta) = \sum_{k=1}^{N_{\mathcal{C}}} \lambda'_{m,k}(\theta)$$

where

$$\lambda'_{k,m}(\theta) = \sum_{j=0}^{R_k-1} \left[\mathbf{1}[p_{k,j} \in V_m] \left(\alpha_m(v_{k,j+1})v'_{k,j+1} - \alpha_m(v_{k,j})v'_{k,j}\right) \right.$$
$$\left. -\mathbf{1}[p_{k,R_k-1} \in U_m] \cdot A_{m,k,R_k-1} v'_{k,R_k-1}\right].$$

Again the proof follows easily by differentiating (32) and writing the loss volume during each interval $p_{k,j}$. For details see [23] where it is also shown that the above estimators are unbiased. We also point out that as with $Q'(\theta)$, evaluating this estimator we only need some rates at specific points in time and the event time derivatives $v'_{k,j}$ which are determined next.

Event Time Derivatives

Theorems 3 and 4 provide estimators of the sample derivatives $Q'(\theta)$ and $L'(\theta)$ respectively. Both estimators require the event time derivatives $v'_{k,j}$, $j = 1, \cdots, R_k - 1$ and $k = 1, \cdots, N_{\mathcal{C}}$. As already discussed, the beginning and end points of $\mathcal{C}_k$ are independent of θ because they correspond to exogenous events (e_1), so $v'_{k,0} = v'_{k,R_k} = 0$. The following theorem provides an iterative algorithm for determining the event time derivative $v'_{k,i+1}$ given $v'_{k,i}$. To simplify the notation we also use

$$x_{k,i} = x(v_{i,k};\theta)$$

Theorem 5. *The event time derivative of any endogenous event occurring at time v_{i+1}, $i = 0, \cdots, R_k - 1$, is given by*

$$v'_{k,i+1} = F_{k,i} \cdot v'_{k,i} + G_{k,i},$$

where $F_{k,i}$ and $G_{k,i}$ and are given below

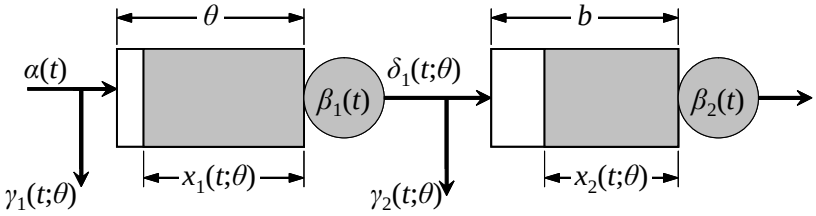

Fig. 5. Tandem two-node network

$$F_{k,i} = \begin{cases} S_{k,i}\frac{A_{m+1,k,i}}{A_{m+1,k,i+1}} & \text{if } x_{k,i} = T_m \text{ and } x_{k,i+1} = T_{m+1} \\ S_{k,i}\frac{A_{m,k,i}}{A_{m,k,i+1}} & \text{if } x_{k,i} = T_m \text{ and } x_{k,i+1} = T_{m-1} \\ \frac{A_{m+1,k,i}}{A_{m+1,k,i+1}} & \text{if } x_{k,i} = x_{k,i+1} = T_m \text{ and } x(t;\theta) > T_m, t \in p^o_{k,i} \\ \frac{A_{m,k,i}}{A_{m,k,i+1}} & \text{if } x_{k,i} = x_{k,i+1} = T_m \text{ and } x(t;\theta) < T_m, t \in p^o_{k,i} \end{cases}$$

$$G_{k,i} = \begin{cases} -\frac{1}{A_{m+1,k,i+1}} & \text{if } x_{k,i} = T_m = \theta \text{ and } x_{k,i+1} = T_{m+1} \\ -\frac{1}{A_{m,k,i+1}} & \text{if } x_{k,i} = T_m = \theta \text{ and } x_{k,i+1} = T_{m-1} \\ \frac{1}{A_{m,k,i+1}} & \text{if } x_{k,i} = T_{m-1} \text{ and } x_{k,i+1} = T_m = \theta \\ \frac{1}{A_{m+1,k,i+1}} & \text{if } x_{k,i} = T_{m+1} \text{ and } x_{k,i+1} = T_m = \theta \\ 0 & \text{otherwise} \end{cases}$$

where $S_{k,i} = 1$ *if* $x_{k,i+1} \neq \theta$ *and* $S_{k,i} = -1$ *if* $x_{k,i+1} = \theta$.

Recall that $A_{m,k,i}$ is given by (29). Also note that $G_{k,i} \neq 0$ only for the intervals i that either start or end at threshold θ. For the proof the interested reader is referred to [23]. We point out that unlike the estimators of the previous section, where we were able to determine a closed-form expression, these estimators are evaluated using the iterative algorithm of Theorem 5. An exception is the two-class case, where again it is possible to obtain a closed-form expression (see [17] and [23]).

5 Tandem Networks

In this section, we describe how perturbations in a buffer threshold propagate in a network. For simplicity, we limit ourselves to the two-node network of Fig. 5; for the general m-node network the reader is referred to [27]. The control parameter of interest is θ, the buffer size of node 1, and we are interested in the sample derivatives of the workload and loss volume of node 2; recall that the corresponding sample derivatives for node 1 were obtained in Section 3.

For notational convenience, let as focus on node 2 and use $x(t;\theta) = x_2(t;\theta)$ and $y(t;\theta) = x_1(t;\theta)$. Also, let $\gamma(t;\theta) = \gamma_2(t;\theta)$ and define the net inflow rate to node 2

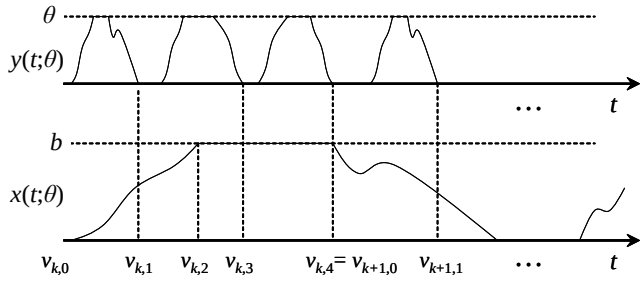

Fig. 6. A typical sample path for the tandem two-node network

$$A(t;\theta) = \delta_1(t;\theta) - \beta_2(t) \tag{37}$$

where, $\delta_1(t;\theta)$ is the outflow from node 1 and is equal to $\alpha(t)$ if $y(t;\theta) = 0$ or $\beta_1(t)$ if $y(t;\theta) > 0$. Using this notation, the objective functions of interest are

$$Q(\theta) = \int_0^T x(t;\theta)dt, \quad \text{and} \quad L(\theta) = \int_0^T \gamma(t;\theta)dt \tag{38}$$

A typical sample path of this system is shown in Fig. 6. Following the practice of the previous sections, we again partition the sample path of $x(t;\theta)$ into boundary and non-boundary periods and form cycles consisting of a non-boundary period and the following boundary period. The kth non-boundary period starts at $v_{k,0}$ with either one of the events $x(t;\theta)$ *ceases to be empty* or $x(t;\theta)$ *ceases to be full*, and ends at v_{k,r_k} with the events $x(t;\theta)$ *becomes either empty or full*. During this interval we also observe $r_k - 1$ buffer $y(t;\theta)$ *becomes empty* events. Similarly, the kth boundary period starts at v_{k,r_k} and ends at $v_{k,R_k} = v_{k+1,0}$ (the beginning of the next cycle). During the boundary period we observe $R_k - r_k - 1$ buffer $y(t;\theta)$ *becomes empty* events which in [27] are referred to as *active switchover points*; these are important because they are the *only* points that propagate the effect of perturbing θ downstream.

Theorem 6. *The workload and loss volume sample derivatives are given by*

$$Q'(\theta) = -\sum_{k=1}^{N_C} \sum_{j=1}^{r_k-1} (v_{k,r_k} - v_{k,j})\, \psi_{k,j} - \sum_{k=1}^{N_C} (v_{k,r_k} - v_{k,0})\, \phi_k$$

$$L'(\theta) = -\sum_{k=1}^{N_C} \sum_{\substack{j=1 \\ j\neq r_k}}^{R_k} \mathbf{1}[x_k]\, \psi_{k,j} + \sum_{k=1}^{N_C} \phi_k$$

where $\mathbf{1}[x_k] = \mathbf{1}[x(v_{k,r_k};\theta) = b]$ *and*

$$\psi_{k,j} = \left(A(v_{k,j}^+;\theta) - A(v_{k,j}^-;\theta)\right) v'_{k,j} = (\alpha(v_{k,j}) - \beta_1(v_{k,j}))\, v'_{k,j}$$

$$\phi_k = A(v_{k,0}^+)v'_{k,0}$$

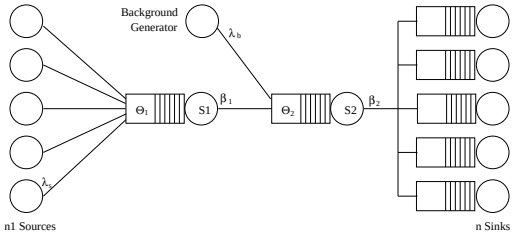

Fig. 7. System topology for the two-stage network

At this point, it is worth pointing out that $v'_{k,R_k} = v'_{k+1,0} = 0$ if the end (or beginning) of a cycle does not coincide with a buffer y *becomes empty* event; note that in this case, the end of a boundary period of x corresponds to an exogenous event. For the general case, when there are m nodes in series, refer to [27] where it is also shown that these estimators are unbiased (under Assumption 1). It is also worth pointing out that in order to evaluate the above estimators, one simply needs a number of timers that measure the intervals $(v_{k,r_k} - v_{k,j})$ for $j = 1, \cdots, r_k - 1$ and $k = 1, \cdots, N_{\mathcal{C}}$. In addition, one needs some rate information at specific points in time and such information is generally easy to get. Finally, the only remaining information is the event time derivatives $v'_{k,j}$. But these are precisely the event time derivatives evaluated in (9).

6 Simulation Results

For illustration purposes, in this section, we present some simulation results (also reported in [28]), where we use the estimators obtained in the earlier sections to control the buffer thresholds of a two-node network. The main objective is to optimize a cost function that consists of the weighted sum of loss and workload in the two-queue tandem system shown in Fig. 7. We emphasize that the setting of Fig. 7 may correspond to a "real system", so that it is best captured by a discrete-event model (not the fluid models that were considered so far). However, the simple form of the estimators obtained allows us to evaluate them using information readily available from the sample paths of discrete-event systems. Recall that all estimators consist only of some timers and counters. They simply count the number of cycles with overflow, or measure the interval between the occurrence of certain events (typically events that make a buffer become full or empty and events that make the buffer cease to be full or empty). Even though there is no guarantee that the estimators are still unbiased when evaluated using information from the discrete-event sample path, our simulation results indicate that such estimators can be used to solve practical optimization problems.

In the system of Fig. 7, intended to represent the operation of a communication network, the inflow process at the first queue consists of n_1 multiplexed

on–off data sources generating bursty traffic. When in the *on* state, each source generates a continuous data stream at the rate of α bits per second. These data streams are used to construct UDP packets which are forwarded to the buffer at the first queue and thence across the rest of the network. Each UDP packet consists of a 42-byte header (including UDP, IP, and IEEE 802.3 headers) and a 512 information (data) field, for a total of 554 bytes. The sources provide the content of the information field, and the header is prepended whenever that field becomes full. If the information field is not full at the time the state of a source changes from *on* to *off*, then the incomplete packet waits until the source changes back to the *on* state and completes the information field. In other words, all packets have 554 bytes. The *on* times and *off* times are i.i.d. random variables sampled from the exponential distribution with mean 0.1 seconds. The channel transporting packets from the first queue to the second queue has a capacity of β_1 bps. The inflow process to the second queue consists of the outflow process from the first queue and of traffic from the background generator. The background traffic consists of n_2 independent sources. Each one of these sources has the same statistical characteristics as the sources to the first queue. The outgoing channel from the second queue has a capacity of β_2 bps.

Note that the average bit rate from either one of the independent sources is $\alpha/2$ bps, since the expected durations of the off periods and the on periods are identical. Therefore, the expected bit rate of the aggregate flow to the first queue is $(n_1\alpha/2)\times(554/512)$, where the latter term accounts for the insertion of the headers. Consequently, the traffic intensity at the first queue, denoted by ρ_1, is given by

$$\rho_1 \;=\; n_1 \times \frac{\alpha}{2} \times \frac{554}{512} \times \frac{1}{\beta_1}. \tag{39}$$

Similarly, the traffic intensity of the second queue is denoted by ρ_2. All of the experiments were performed using the *Georgia Tech Network Simulator* (*GTNetS*) [29], modified to include the requisite IPA derivative calculations. In our simulation experiments we set $n_1 = n_2 = 100$, $\beta_1 = 10$ *Mbps*, and $\beta_2 = 20$ *Mbps*. For the simulation results, we set $\rho_1 = 0.95$ and calculated α according to (39).

Let $\boldsymbol{\theta} = [\theta_1, \theta_2]$ denote the two-dimensional parameter vector consisting of the buffer limits at the first and second queue respectively. The loss volumes and workloads at the two queues are denoted by $L_j(\boldsymbol{\theta})$ and $Q_j(\boldsymbol{\theta})$, $j = 1, 2$ (see (3), (4) and (38)). Let us define the cost function $J(\boldsymbol{\theta})$ as the weighted sum of the average loss rate and workload rate.

$$J(\boldsymbol{\theta}) = \frac{1}{T}\left[L_1(\boldsymbol{\theta}) + 10Q_1(\boldsymbol{\theta}) + L_2(\boldsymbol{\theta}) + 20Q_2(\boldsymbol{\theta})\right].$$

Recall that T is the observation interval over which the objective function is defined and it is set to $T = 1$ second. We seek to minimize $\mathbb{E}[J(\boldsymbol{\theta}]$ using a standard stochastic approximation technique (1) which defines a sequence of points $\boldsymbol{\theta}_n = [\theta_1^n, \theta_2^n]$. However, we substitute the gradient of $J(\boldsymbol{\theta})$,

$\mathbf{H}_n(\boldsymbol{\theta}_n; \mathbf{x}(0); \omega_n^{SFM})$ with $\mathbf{H}_n(\boldsymbol{\theta}_n; \mathbf{x}(0); \omega_n^{DES})$ to indicate that the gradient evaluation is done based on data observed from a discrete-event sample path. The required gradient is evaluated through the IPA algorithms described in the previous sections (specifically Theorems 1 and 6). In addition, although all our analysis is based on the assumption that all observed sample paths start with all queues at the empty state, we have nonetheless applied the IPA estimates at the nth iteration of the optimization algorithm using the ending state of the $(n-1)$th iteration. Furthermore, we adopt the step size sequence $\sigma_n = 10/n^{0.6}$. Finally, we used a simple heuristic to bound the displacement $\boldsymbol{\theta}_{n+1} - \boldsymbol{\theta}_n$ along each coordinate by modifying the vector $\mathbf{H}_n = [h_1^n, h_2^n]$ as follows. We first computed the partial derivatives $\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_i}$, $i = 1, 2$. If $|\sigma_n \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_i}| \leq 5$ then we set $h_i^n = \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_i}$, and if $|\sigma_n \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_i}| > 5$, then we set $h_i^n = 5\text{sgn}(\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_i})/\sigma_n$. The parameters θ_i^n $(i = 1, 2)$ were considered as real numbers, but the simulation runs were performed at the respective integer values closest to them.

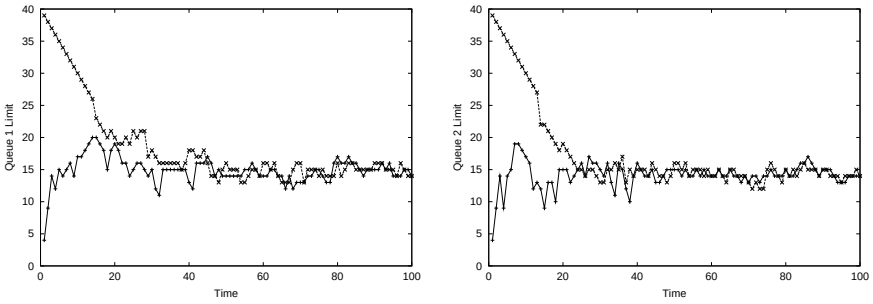

Fig. 8. Evolution of θ_1^n and θ_2^n

We ran the optimization algorithm twice, with two different initial parameters: first with $\boldsymbol{\theta}_1 = [5, 5]$, and then with $\boldsymbol{\theta}_n = [40, 40]$. In either case we ran the algorithm for 100 iterations (i.e., 100 seconds). For each experiment, we plotted the evolution of θ_1^n and θ_2^n as a function of iteration n, and show the results in Fig. 6 respectively. Each of the figures shows one trajectory for the $\boldsymbol{\theta}_n = [5, 5]$ initial condition, and a second one for the $\boldsymbol{\theta}_n = [40, 40]$ initial condition. The results indicate asymptotic convergence to approximately $\boldsymbol{\theta^*}_n = [15, 14]$ within approximately 20 seconds.

Finally, to add validity to these results, we plotted the graph of $J(\theta_1, \theta_2)$ as shown in Fig. 9. Each point on the plot is the average of 10 separate simulation experiments with $T = 100$ seconds, each with a different seed for the random number generators. However, each set of the 10 simulation experiments uses the same set of 10 random seeds as all other sets of experiments. This graph

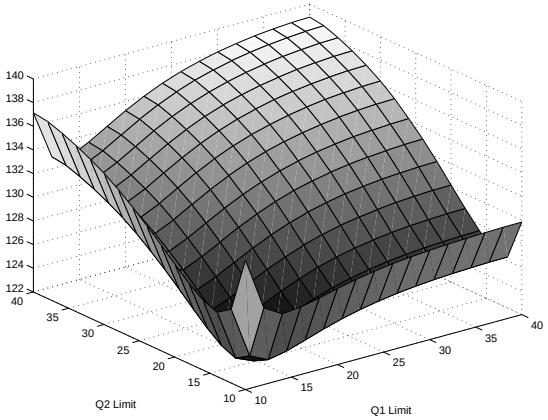

Fig. 9. Cost function $J(\theta_1, \theta_2)$

clearly corroborates the results obtained by the optimization runs, i.e., it shows that $\boldsymbol{\theta^*}_n = [15, 14]$ is indeed optimal.

7 Conclusions and Future Work

In this chapter we used the stochastic fluid modeling paradigm and derived IPA sensitivity estimates of some performance measures of interest with respect to the control parameters θ (or ρ). Subsequently, these estimators were evaluated from data observed in a discrete-event sample path and they were used in stochastic optimization schemes to solve non-linear stochastic optimization problems. For all problems considered in this chapter, there was no feedback involved, in other words, the inflow processes where independent of the control parameters. Such models are appropriate for User Datagram Protocol (UDP) traffic, however, they do not capture the Transport Control Protocol (TCP) traffic. The difficulty of TCP stems from the delayed feedback mechanisms that are embedded in this scheme. As a result, models with feedback are still being developed (see for example [30]). Closing, another open question is whether the IPA evaluation process using data observed from a discrete-event sample path still produces unbiased estimators. In general, this process does introduce some bias; however, our experience so far indicates that the estimators obtained are adequate for practical network management and optimization problems.

Acknowledgements

The work of the second and third authors is supported in part by the National Science Foundation under grants EEC-0088073 and DMI-0330171, by AFOSR under grant F49620-01-0056, and by ARO under grant DAAD19-01-0610.

References

1. W. E. Leland, M. S. Taqq, W. Willinger, and D. V. Wilson, "On the self-similar nature of ethernet traffic," in *ACM SIGCOMM*, pp. 183–193, 1993.
2. V. Paxson and S. Floyd, "Wide-area traffic: The failure of poisson modeling," *IEEE/ACM Transactions on Networking*, vol. 3, pp. 226–244, June 1995.
3. D. Anick, D. Mitra, and M. Sondhi, "Stochastic theory of a data-handling system with multiple sources," *The Bell System Technical Journal*, vol. 61, pp. 1871–1894, 1982.
4. H. Kobayashi and Q. Ren, "A mathematical theory for transient analysis of communications networks," *IEICE Transactions on Communications*, vol. E75-B, pp. 1266–1276, 1992.
5. R. Cruz, "A calculus for network delay, Part I: Network elements in isolation," *IEEE Transactions on Information Theory*, 1991.
6. G. Kesidis, A. Singh, D. Cheung, and W. Kwok, "Feasibility of fluid-driven simulation for ATM network," in *Proc. IEEE Globecom*, vol. 3, pp. 2013–2017, 1996.
7. K. Kumaran and D. Mitra, "Performance and fluid simulations of a novel shared buffer management system," in *Proceedings of IEEE INFOCOM*, March 1998.
8. B. Liu, Y. Guo, J. Kurose, D. Towsley, and W. Gong, "Fluid simulation of large scale networks: Issues and tradeoffs," in *Proceedings of the Intl. Conf. on Parallel and Distributed Processing Techniques and Applications*, June 1999. Las Vegas, Nevada.
9. A. Yan and W. Gong, "Fluid simulation for high-speed networks with flow-based routing," *IEEE Transactions on Information Theory*, vol. 45, pp. 1588–1599, 1999.
10. R. Akella and P. Kumar, "Optimal control of production rate in a failure prone manufacturing system," *IEEE Transactions on Automatic Control*, vol. 31, pp. 116–126, Feb 1986.
11. J. Perkins and R. Srikant, "The role of queue length information in congestion control and resource pricing," in *Proceedings IEEE Conference on Decision and Control*, pp. 2703–2708, Dec 1999.
12. J. Perkins and R. Srikant, "Failure-prone production systems with uncertain demand," *IEEE Transactions on Automatic Control*, vol. 46, pp. 441–449, 2001.
13. Y. Wardi and B. Melamed, "Variational bounds and sensitivity analysis of traffic processes in continuous flow models," *Discrete Event Dynamic Systems: Theory and Applications*, vol. 11, pp. 249–282, 2001.
14. B. Mohanty and C. Cassandras, "The effect of model uncertainty on some optimal routing problems," *Journal of Optimization Theory and Applications*, vol. 77, pp. 257–290, 1993.

15. S. Meyn, "Sequencing and routing in multiclass networks. Part I: Feedback regulation," in *Proceedings of the IEEE International Symposium on Information Theory*, pp. 4440–4445, 2000. To appear in SIAM J. Control and Optimization.
16. C. G. Cassandras, Y. Wardi, B. Melamed, G. Sun, and C. G. Panayiotou, "Perturbation analysis for on-line control and optimization of stochastic fluid models," *IEEE Transactions on Automatic Control*, vol. AC-47, no. 8, pp. 1234–1248, 2002.
17. C. Cassandras, G. Sun, C. Panayiotou, and Y. Wardi, "Perturbation analysis and control of two-class stochastic fluid models for communication networks," *IEEE Transactions on Automatic Control*, vol. 48, pp. 770–782, May 2003.
18. G. Sun, C. G. Cassandras, and C. G. Panayiotou, "Perturbation analysis of a multiclass stochastic fluid model with finite buffer capacity," in *Proceedings of 41st IEEE Conf. On Decision and Control*, pp. 2171–2176, 2002.
19. Y. Wardi, B. Melamed, C. Cassandras, and C. Panayiotou, "IPA gradient estimators in single-node stochastic fluid models," *Journal of Optimization Theory and Applications*, vol. 115, no. 2, pp. 369–406, 2002.
20. Y. C. Ho and X. Cao, *Perturbation Analysis of Discrete Event Dynamic Systems*. Dordrecht, Holland: Kluwer Academic Publishers, 1991.
21. C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Kluwer Academic Publishers, 1999.
22. H. J. Kushner and D. S. Clark, *Stochastic Approximation for Constrained and Unconstrained Systems*. Berlin, Germany: Springer-Verlag, 1978.
23. G. Sun, C. Cassandras, and C. Panayiotou, "Perturbation analysis of multiclass stochastic fluid models," *Journal of Discrete Event Dynamic Systems*, 2004. To Appear.
24. R. Y. Rubinstein and A. Shapiro, *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method*. New York, New York: John Wiley and Sons, 1993.
25. I. Cidon, R. Guérin, and A. Khamisy, "On protective buffer policies," *IEEE/ACM Transactions on Networking*, vol. 2, pp. 240–246, Jun 1994.
26. C. Panayiotou and C. Cassandras, "On-line predictive techniques for "differentiated services" networks," in *Proceedings IEEE Conference on Decision and Control*, pp. 4529–4534, Dec 2001.
27. G. Sun, C. Cassandras, Y. Wardi, and C. Panayiotou, "Perturbation analysis of stochastic flow networks," in *Proceedings of IEEE Conference on Decision and Control*, pp. 4831–4838, Dec 2003.
28. G. Sun, C. Cassandras, Y. Wardi, C. Panayiotou, and G. Riley, "Perturbation analysis and optimization of stochastic flow networks," 2003. Submitted.
29. G. F. Riley, "The Georgia Tech Network Simulator," in *Proceedings of Workshop on Models, Methods, and Tools for Reproducible Network Research (MoMeTools)*, Aug. 2003.
30. H. Yu and C. G. Cassandras, "Perturbation analysis of feedback-controlled stochastic flow systems," in *Proceedings of the IEEE Conference On Decision and Control*, pp. 6277–6282, 2003.