Chapter 1

A NEW PARADIGM FOR ON-LINE MANAGEMENT OF COMMUNICATION NETWORKS WITH MULTIPLICATIVE FEEDBACK CONTROL

Haining Yu
Department of Manufacturing Engineering, Boston University
fernyu@bu.edu

Christos G. Cassandras
Department of Manufacturing Engineering, Boston University
Corresponding author
cgc@bu.edu

Abstract We describe the use of Stochastic Flow Models (SFMs) for control and optimization (rather than performance analysis) of computer networks. After reviewing earlier work applying Infinitesimal Perturbation Analysis (IPA) to SFMs without feedback or with additive feedback, we consider systems operating with a multiplicative feedback control mechanism. Using IPA, we derive gradient estimators for loss and throughput related performance metrics with respect to a *feedback gain* parameter and show their unbiasedness. We also illustrate the use of these estimators in network control by combining them with standard gradient–based stochastic approximation schemes and providing several simulation-based examples.

1. Introduction

A natural modelling framework for computer networks is provided by Discrete Event Systems (DES), most notably through queueing theory, e.g., Kleinrock (1975). However, it has become increasingly difficult for traditional queueing theory to handle the complexity of today's computer networks. First of all, the enormous traffic volume in today's

Internet (which is still growing) makes packet-by-packet queueing analysis infeasible. Moreover, the discovery of self-similar patterns in the Internet traffic distribution (see Leland et al., (1993)) and the resulting inadequacies of Poisson traffic models (see Paxson and Floyd (1995)) further complicate queueing analysis. Consequently, performance analysis techniques that do not depend on detailed traffic distributional information are highly desirable. Fluid models have thus 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 by Anick et al., (1982) and later proposed by Kobayashi and Ren (1992) for the analysis of multiplexed data streams and by Cruz (1991) for network performance analysis, fluid models have been shown to be especially useful for simulating various kinds of high speed networks (see Kesidis et al., (1996), Kumaran and Mitra (1998), Liu et al., (1999), Yan and Gong (1999)). Stochastic Flow Models (SFM) have the extra feature that the flow rates are treated as general *stochastic* processes, which distinguishes itself from the approach adopted in Akella and Kumer (1986) and other work, e.g., Perkins and Srikant (1999), Perkins and Srikant (2001).

While the aggregation property of SFMs brings efficiency to *performance analysis*, the resulting accuracy depends on traffic conditions, the structure of the underlying network, and the nature of the performance metrics of interest. On the other hand, SFMs often capture the critical features of the underlying "real" network, which is useful in solving *control and optimization* problems. In control and optimization, e.g., Kelly et al., (1998) and Low (2000), estimating the gradients of given performance metrics with respect to key parameters becomes an essential task. Perturbation Analysis (PA) methods (see Ho and Cao (1991), Cassandras and Lafortune (1999)) are therefore suitable, if appropriately applied to a SFM as an abstraction of an underlying network component or a network, as in recent work by Wardi et al., (2002), Liu and Gong (1999), Cassandras et al., (2002), and Cassandras et al., (2003). In a single node with threshold-based buffer control, Infinitesimal Perturbation Analysis (IPA) has been shown to yield simple sensitivity estimators for loss and workload metrics with respect to threshold parameters;

see Cassandras et al., (2002). In the multiclass case studied in Cassandras et al., (2003), the estimators generally depend on traffic rate information, but not on the stochastic characteristics of the arrival and service processes involved. In addition, the estimators obtained are unbiased under very weak structural assumptions on the defining traffic processes. As a result, they can be evaluated *based on data observed on a sample path of the actual (discrete-event) network* and combined with gradient-based optimization schemes as shown in Cassandras et al., (2002) and Cassandras et al., (2003). This makes it possible to adjust parameters on line in order to adapt to rapidly changing network situations. On-line management is appealing in today's computer networks and will become even more important as high speed network technologies, such as Gigabyte Ethernet and optical networks 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.

An important feature in today's Internet management is the presence of *feedback* mechanisms. For example, in Random Early Detection (see Floyd and Jacobson 1993), an IP router may send congestion signals to TCP flows by dropping packets and a TCP flow should adjust its window size (and therefore its sending rate) according to feedback signals (for example, acknowledgement packets sent back from a destination node; see Jacobson (1988)). However, queueing networks have been studied largely based on the assumption that the system state, typically queue length information, has no effect on arrival and service processes, i.e., in the absence of feedback. Unfortunately, the presence of feedback significantly complicates analysis, and makes it extremely difficult to derive closed-form expressions of performance metrics such as average queue length or mean waiting time (unless stringent assumptions are made; see Takacs (1963), Foley and Disney (1983), Avignon and Disney (1977/78), Wortman et al., (1991)), let alone developing analytical schemes for performance optimization. It is equally difficult to extend the theory of PA for discrete-event queueing systems in the presence of feedback. The importance of incorporating feedback to networks as well as their SFM counterparts, and the effectiveness of IPA methods applied to SFMs to date motivates the study of SFMs with *multiplicative* feedback. We define $\alpha(t)$ as the maximal external incoming flow rate for a node in the network and introduce a feedback mechanism by setting the inflow rate to $c \cdot \alpha(t)$ when the buffer content $x(t)$ is greater than a certain intermediate threshold ϕ. It is worth noticing that this form of feedback has been widely adopted in today's Internet, i.e., in the Random Early Detection (RED) (see Floyd and Jacobson 1993) and other algorithms.

The rest of the chapter is organized as follows. Section 2 briefly reviews earlier work applying IPA to SFMs. Section 3 presents the SFM framework with multiplicative feedback. In Section 4 we carry out IPA and derive explicit sensitivity estimators for loss and throughput related metrics. We also prove unbiasedness of these estimators. In Section 5 we present some numerical examples to illustrate the use of IPA estimation in on-line queueing system control. Conclusions and future research directions are given in Section 6.

2. Review of IPA in SFMs

In this section we briefly review earlier results incorporating IPA to SFMs. Consider a network node where buffer control at the packet level takes place using a simple threshold-based policy: when a packet arrives and the queue length $x(t)$ is below a given level b, it is accepted; otherwise, it is rejected. This can be modeled as a queueing system. Next, we adopt a simple SFM for the system, treating packets as "fluid." The buffer content at time t is again denoted by $x(t)$ and it is limited to θ, which may be viewed as the capacity or as a threshold parameter used for buffer control. When the buffer level reaches θ the system starts to overflow. In the underlying DES, both $x(t)$ and b are integers; in the SFM, $x(t)$ and θ are treated as real numbers. As we will explain later, analyzing the SFM provides useful information for solving control and optimization problems defined on the underlying network node. Figure 1.1 shows a typical network node on the left with buffer control and its SFM counterpart on the right. In the SFM, the maximal processing rate of the server is generally time-varying and denoted by $\beta(t)$. The incoming rate, also generally time-varying, is denoted by $\alpha(t)$. We also use $\delta(t)$ and $\gamma(t)$ to denote the outflow flow rate and the overflow rate due to excessive incoming fluid at a full buffer respectively.

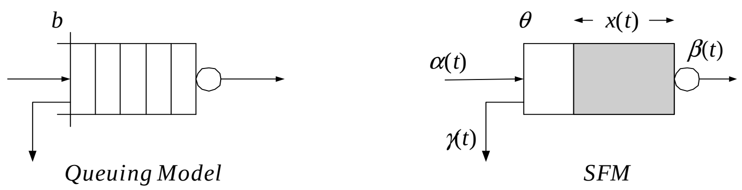

Figure 1.1. A network node with threshold-based buffer control and its SFM counterpart

Over a time interval $[0, T]$, the buffer content $x(t;\theta)$ is determined by the following one-sided differential equation,

$$\frac{dx(t;\theta)}{dt^+} = \begin{cases} 0, & \text{if } x(t;\theta) = 0 \text{ and } \alpha(t) - \beta(t) \leq 0, \\ 0, & \text{if } x(t;\theta) = \theta \text{ and } \alpha(t) - \beta(t) \geq 0, \\ \alpha(t) - \beta(t), & \text{otherwise} \end{cases} \quad (1.1)$$

The overflow rate $\gamma(t;\theta)$ is given by

$$\gamma(t;\theta) = \begin{cases} \max\{\alpha(t) - \beta(t), 0\}, & \text{if } x(t;\theta) = \theta, \\ 0, & \text{if } x(t;\theta) < \theta. \end{cases} \quad (1.2)$$

A typical state trajectory over $[0, T]$ can be decomposed into two kinds of intervals: *empty periods* and *buffering periods*. Empty Periods (EP) are intervals during which the buffer is empty, while Buffering Periods (BP) are intervals during which the buffer is nonempty. We define EPs to always be closed intervals, whereas BPs are open intervals unless containing one of the boundary points 0 or T.

We consider two performance metrics, the *Loss Volume* $L_T(\theta)$ and the *Cumulative Workload* (or just *Work*) $Q_T(\theta)$, both defined on the interval $[0, T]$ as:

$$L_T(\theta) = \int_0^T \gamma(t;\theta)dt, \quad (1.3)$$

and

$$Q_T(\theta) = \int_0^T x(t;\theta)dt. \quad (1.4)$$

Let us denote the kth BP as $\mathcal{B}_k$. Define $\Phi(\theta)$ as the index set for all BPs with at least one overflow period. For every $k \in \Phi(\theta)$, there is a (random) number $M_k \geq 1$ of *overflow periods* in $\mathcal{B}_k$, i.e., intervals during which the buffer is full and $\alpha(t) - \beta(t) > 0$. Let us denote these overflow periods by $\mathcal{F}_{k,m}$, $m = 1, \ldots, M_k$, in increasing order and express them as $\mathcal{F}_{k,m} = [u_{k,m}(\theta), v_{k,m}]$, $k = 1, \ldots, K$. Observe that the starting time $u_{k,m}(\theta)$ generally depends on θ, whereas the ending time $v_{k,m}$ is locally independent of θ, since it corresponds to a change of sign in $\alpha(t) - \beta(t)$ in (1.1), which has been assumed independent of θ.

Through Infinitesimal Perturbation Analysis (IPA), the following sample derivatives can be obtained, as shown in Cassandras et al., (2002):

Proposition 2.1. For every $\theta \in \Theta$,

$$L_T^{'}(\theta) = -|\Phi(\theta)|. \quad (1.5)$$

and

$$Q_T^{'}(\theta) = \sum_{k \in \Phi(\theta)} [\eta_k(\theta) - u_{k,1}(\theta)]. \quad (1.6)$$

Under certain technical conditions (see Cassandras et al., (2002)), it is also proved that

Proposition 2.2. The IPA estimators $L_T'(\theta)$ and $Q_T'(\theta)$ are unbiased, i.e.,

$$\frac{\partial E[L_T(\theta)]}{\partial \theta} = E\left[L_T'(\theta)\right] \text{ and } \frac{\partial E[Q_T(\theta)]}{\partial \theta} = E\left[Q_T'(\theta)\right]$$

These IPA estimators are extremely simple to implement not only in a SFM, but in the actual network component as well: (1.5) is merely a counter of all BPs observed in $[0, T]$ in which at least one overflow event takes place. The estimator is *nonparametric* in the sense that no knowledge of the traffic or processing rates is required, nor does (1.5) depend on the nature of the random processes involved. In (1.6), the contribution of a BP, $\mathcal{B}_k$, to the sample derivative $Q_T'(\theta)$ is the length of the interval defined by the first point at which the buffer becomes full and the end of the BP. Once again, as in (1.5), the IPA derivative $Q_T'(\theta)$ is nonparametric, since it requires only the recording of times at which the buffer becomes full (i.e., $u_{k,1}(\theta)$) and empty (i.e., $\eta_k(\theta)$) for any BP which has at least one overflow period. In other words, (1.5) and (1.6) may be directly obtained from a single sample path of the network node, and the final values of the estimators are independent of the SFM.

The analysis above can be extended to a network with multiple nodes. In Sun et al., (2003), a tandem network is studied, where the output of a node becomes the input to a downstream node, and the dynamics of each node follow those of the single-node system described above. Figure 1.2 shows such a tandem network. By similar techniques as in Cassandras et al., (2002), IPA analysis can be carried out and the unbiasedness of IPA estimators can be proved for loss and work related metrics with respect to the parameters $\theta, b_2, \ldots, b_m$.

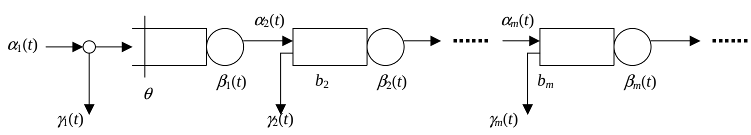

Figure 1.2. SFM of a tandem network

Figure 1.3 illustrates another extension by introducing feedback. In this case, $\alpha(t)$ is not the inflow rate. The actual incoming flow is the output of a traffic shaper. A *traffic shaper* modifies some incoming process $\alpha(t)$, according to system information (i.e., queue content information $x(t)$), and creates the actual incoming flow to the node. We denote the actual inflow rate (which is also the output of the traffic shaper) by

$u(t)$. If we regard queue content as state information, $x(t)$ being input to the above traffic shaper implies a feedback controller of the form: $u(t) = u(\alpha(t), x(t))$. Because $u(t)$ depends on $x(t)$ and $x(t)$ depends on the buffer capacity θ, the inflow rate is not independent of θ. However in what follows we will simply use $u(t)$ for notational simplicity unless the dependence needs to be stressed. The system dynamics are:

$$\frac{dx(t;\theta)}{dt^+} = \begin{cases} 0, & \text{if } x(t;\theta) = 0 \text{ and } u(t) - \beta(t) \leq 0, \\ 0, & \text{if } x(t;\theta) = \theta \text{ and } u(t) - \beta(t) \geq 0, \\ u(t) - \beta(t), & \text{otherwise} \end{cases} \quad (1.7)$$

The outflow rate $\delta(t)$ is given by

$$\delta(t) = \begin{cases} \beta(t) & \text{if } x > 0 \\ \min\{u(t), \beta(t)\} & \text{if } x = 0 \end{cases} \quad (1.8)$$

The overflow rate $\gamma(t)$ is given by

$$\gamma(t) = \begin{cases} \max\{u(t) - \delta(t), 0\} & \text{if } x = \theta \\ 0 & \text{if } x < \theta \end{cases} \quad (1.9)$$

Similarly, $\delta(t)$ and $\gamma(t)$ are functions of $\alpha(t), x(t)$ and θ. But we do not explicitly indicate the dependence unless it is necessary to do so.

In Yu and Cassandras (2003), an *additive feedback* mechanism is studied by setting the inflow rate $u(t)$ to be

$$u(t) = \alpha(t) - p(x(t)),$$

where $p(x)$ is a *feedback function.* Using similar techniques as in Cassan-

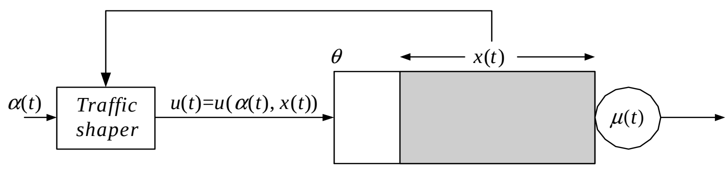

Figure 1.3. A SFM with feedback

dras et al., (2002), we can derive the IPA sample derivatives $L_T'(\theta)$ and $Q_T'(\theta)$ and prove their unbiasedness under modest technical assumptions. Moreover, in the case of linear feedback, i.e., $p(x) = cx$, the estimators again turn out to be nonparametric; for details, see Yu and Cassandras (2003).

The feedback mechanism in Yu and Cassandras (2003) implies that state information, i.e., buffer content, is instantaneously available to the controller. This is reasonable for situations such as manufacturing systems, but unlikely to hold in high-speed distributed environments such as communication networks. This stringent requirement, together with a natural interest in feedback policies which are readily applicable to real-world networks, leads to the problem of deriving IPA gradient estimators for SFMs with *multiplicative* feedback mechanisms. Consider a single-node SFM with threshold-based buffer control as in Cassandras et al., (2002). Once again, we define $\alpha(t)$ as the maximal external incoming flow rate and introduce a feedback mechanism by setting the inflow rate to $c \cdot \alpha(t)$ when the buffer content $x(t)$ is greater than a certain intermediate threshold ϕ. Compared with Yu and Cassandras (2003), the current mechanism has two main advantages: (i) System information is needed only when the buffer content reaches or leaves the threshold ϕ, while in Yu and Cassandras (2003) it has to be continuously available; as a result, the cost of communicating state information is greatly reduced, and (ii) The multiplicative feedback mechanism can be easily implemented in an actual network, for example via probabilistic dropping. Similar to our previous work, applying IPA and deriving the SFM-based sensitivity information of certain performance metrics with respect to key parameters is still our primary interest. However, the following differences makes the problem more challenging: First, our interest switches from θ, which decides the feedback range to c, which decides the feedback gain. Secondly, the feedback only applies in part of the range, i.e. there is no feedback when $x < \phi$. While these features make the implementation of this feedback mechanism simple, they also complicate the IPA estimation.

3. A SFM with Multiplicative Feedback

The SFM for a typical network node that we consider consists of a server with a finite buffer as shown in Fig. 1.3. In the remainder of this chapter, we study the following traffic shaper with *multiplicative feedback*:

$$u(t) = \begin{cases} c\alpha(t) & \text{if } \phi < x \leq \theta \\ \alpha(t) & \text{if } 0 \leq x < \phi \end{cases} \tag{1.10}$$

where $\alpha(t)$ is the maximal inflow rate, c is the feedback gain parameter and $\phi < \theta$ is an intermediate threshold. We assume $0 < c \leq 1$, thus ensuring that the effect of feedback is more pronounced when $x > \phi$. When the buffer level is below ϕ, the whole flow is accepted into the system; when the buffer level is above ϕ, part of the flow may be rejected before entering the system. Thus, ϕ decides the feedback *range* and c decides the feedback *gain*. As discussed before, the inflow rate $u(t)$ is a function of $x(t)$ and c. Since the queue length $x(t)$ is a function of ϕ and θ, $u(t)$ depends on ϕ and θ too. But in what follows we will simply denote it by $u(t)$ for notational simplicity unless the dependence needs to be stressed. Note that from an implementation standpoint (1.10) is a policy in which packets arriving at a node after its queue content exceeds a level ϕ are dropped with probability $1 - c$. The policy can be readily extended to one with multiple thresholds $\phi_1, \ldots, \phi_n$ and corresponding gains $c_1, \ldots, c_n$ to resemble the RED algorithm Floyd and Jacobson 1993 adopted as part of congestion control in the Internet. Thus, a byproduct of the analysis that follows is to develop means for determining state-dependent packet dropping probabilities that optimize a performance metric of choice.

The only requirement imposed by the feedback mechanism in (1.10) is that the source be notified of the events: "$x(t)$ reaches ϕ" or "$x(t)$ leaves ϕ". It is also assumed that the stochastic processes $\{\alpha(t)\}$ and $\{\beta(t)\}$ are independent of the buffer level $x(t)$ and of the parameters c, ϕ or θ. Further, it is assumed that the rate processes are bounded in the sense that there exist $\alpha_{\max}$ and $\beta_{\max}$ such that w.p. 1 $\alpha(t) \leq \alpha_{\max} < \infty$ and $\beta(t) \leq \beta_{\max} < \infty$. Finally, we assume that the real-valued parameter c is confined to a closed and bounded (compact) interval $\mathcal{C}$ and that $c > 0$ for all $c \in \mathcal{C}$. Now we can see that the dynamics of the buffer content are given by

$$\frac{dx(t)}{dt^+} = \begin{cases} \max\{u(t) - \delta(t), 0\} & \text{when } x(t) = 0 \\ u(t) - \delta(t) & \text{when } 0 < x(t) < \theta \\ \min\{u(t) - \delta(t), 0\} & \text{when } x(t) = \theta \end{cases} \tag{1.11}$$

where $\delta(t)$ is the outflow rate defined in (1.8). Note that the above dynamics are not yet complete, because the case $x(t) = \phi$ in (1.10) is not specified. In order to fully specify it, let us take a closer look at all possible cases when $x(t) = \phi$:

Case 1. $\beta(t) < c\alpha(t)$: The buffer level at t^+ becomes $x(t^+) > \phi$;

Case 2. $\alpha(t) < \beta(t)$: The buffer level at t^+ becomes $x(t^+) < \phi$;

Case 3 $c\alpha(\tau) \leq \beta(\tau) \leq \alpha(\tau)$ for all τ in an interval $[t, t + \varepsilon)$ for some $\varepsilon > 0$: There are two further cases to consider: (i) If we set $u(\tau) = c\alpha(\tau)$, it follows that

$$\left.\frac{dx}{dt}\right|_{t=\tau} = c\alpha(\tau) - \beta(\tau) \leq 0$$

and the buffer content immediately starts decreasing. Therefore $x(\tau^+) < \phi$ and the actual incoming rate becomes $u(\tau^+) = \alpha(\tau^+)$. Thus,

$$\left.\frac{dx}{dt}\right|_{t=\tau^+} = \alpha(\tau^+) - \beta(\tau^+) \geq 0$$

and the buffer content starts increasing again. This process repeats, resulting in a "chattering" behavior. (ii) If, on the other hand, we set $u(\tau) = \alpha(\tau)$, it follows that $\frac{dx}{dt}\big|_{t=\tau^+} = \alpha(\tau^+) - \beta(\tau^+) \geq 0$. Then, upon crossing ϕ, the actual input rate must switch to $c\alpha(\tau^+)$ which gives $c\alpha(\tau^+) - \beta(\tau^+) \leq 0$. This implies that the buffer content immediately decreases below ϕ and a similar chattering phenomenon occurs.

The chattering behavior above is due to the nature of the SFM and does not occur in the actual DES where buffer levels are maintained for finite periods of time; in the present SFM, it is readily prevented by setting $u(\tau) = \beta(\tau)$ and $\frac{dx(t)}{dt} = 0$ for all $\tau \geq t$ such that the buffer content is maintained at ϕ. Note that $u(t)$ is a function of of $x(t)$ and c, i.e., $u(t) = u(t, x(t); c)$. Now we can complete the dynamics by modifying (1.10) as follows:

$$u(t, x(t); c) = \begin{cases} \alpha(t) & \text{when } 0 < x < \phi \\ c\alpha(t) & \text{when } x(t) = \phi \text{ and } \beta(t) < c\alpha(t) \\ \beta(t) & \text{when } x(t) = \phi \text{ and } c\alpha(t) \leq \beta(t) \leq \alpha(t) \\ \alpha(t) & \text{when } x(t) = \phi \text{ and } \alpha(t) < \beta(t) \\ c\alpha(t) & \text{when } \phi < x \leq \theta \end{cases} \tag{1.12}$$

with the initial condition $x(0; c) = 0$.

Our objective is to obtain sensitivity information of some performance metrics with respect to key parameters. We limit ourselves to considering c as the controllable parameter of interest. For a finite *time horizon* $[0, T]$ during which c is fixed, we define the *throughput* as:

$$H_T = \frac{1}{T}\int_0^T \delta(t)dt \tag{1.13}$$

and the *loss rate* as:

$$\begin{aligned} L_T =& \frac{1}{T}\int_0^T \mathbf{1}[x(t) = \theta](u(t) - \delta(t))dt \\ =& \frac{1}{T}\int_0^T \mathbf{1}[x(t) = \theta](c\alpha(t) - \beta(t))dt \end{aligned} \tag{1.14}$$

where $\mathbf{1}[\cdot]$ is the usual indicator function. A typical optimization problem is to determine c^* that maximizes a cost function of the form

$$J_T(\theta) = E[H_T(c)] - \lambda \cdot E[L_T(c)] \tag{1.15}$$

where λ generally reflects the trade-off between maintaining proper throughput and incurring high loss. Care must also be taken in defining the previous expectations over a finite time horizon, since they generally depend on initial conditions; we shall assume that the queue is empty at time 0.

Note that we do not make any stationarity assumption here since the performance metrics are defined over a finite time interval. Moreover, the finite-horizon formulation is suitable for the "moving finite horizon" type of network performance problems where one trades off short-term quasi-stationary behavior against long-term changes possibly caused by user behavior.

In order to accomplish this optimization task, we rely on estimates of $dE[H_T(c)]/dc$ and $dE[L_T(c)]/dc$ provided by the sample derivatives $dH_T(c)/dc$ and $dL_T(c)/dc$. Accordingly, the main objective of the following sections is the derivation of $dH_T(c)/dc$ and $dL_T(c)/dc$, which we will pursue through IPA techniques. For any sample performance metric $\mathcal{L}(\theta)$ and a generic parameter θ, the IPA gradient estimation technique computes $d\mathcal{L}(\theta)/d\theta$ along an observed sample path. If the IPA-based estimate $d\mathcal{L}(\theta)/d\theta$ satisfies $dE[\mathcal{L}(\theta)]/d\theta = E[d\mathcal{L}/d\theta]$, it is unbiased. Unbiasedness is the principal condition for making the application of IPA practical, since it enables the use of the IPA sample derivative in stochastic gradient-based algorithms. A comprehensive discussion of IPA and its applications can be found in Ho and Cao (1991), Glasserman (1991) and Cassandras and Lafortune (1999).

4. Infinitesimal Perturbation Analysis

In this section we tackle the performance optimization problem raised in the last section. After introducing the notion of sample path decomposition in Section 4.1, we present our main results, namely, the IPA gradient estimates and their unbiasedness in Section 4.2. As we will see, the IPA gradient estimates rely on *event time sample derivatives*, which will be derived in Section 4.3. Some critical properties for the proof of unbiasedness will be established in Section 4.4.

4.1 Sample Path Decomposition and Event Definition

As already mentioned, our objective is to estimate the derivatives $dE[H_T(c)]/dc$ and $dE[L_T(c)]/dc$ through the sample derivatives $dH_T(c)/dc$ and $dL_T(c)/dc$, which are commonly referred to as IPA estimators. In the process, however, it will be necessary to identify events of interest and decompose the sample path first.

For a fixed c, the interval $[0,T]$ is divided into alternating boundary periods and non-boundary periods. A *Boundary Period* (BP) is defined as a time interval during which $x(t)=\theta$ or $x(t)=0$, and a *Non-Boundary Period* (NBP) is defined as a time interval during which $0<x(t)<\theta$. BPs are further classified as *Empty Periods* (EP) and *Full Periods* (FP). An EP is an interval such that $x(t)=0$; a FP is an interval such that $x(t)=\theta$. We assume that there are N NBPs in the interval $[0,T]$, where N is a random number. We index these NBPs by $n=1,\ldots,N$ and express them as $[\eta_n,\zeta_n)$. Figure 1.4 shows a typical sample path of the SFM.

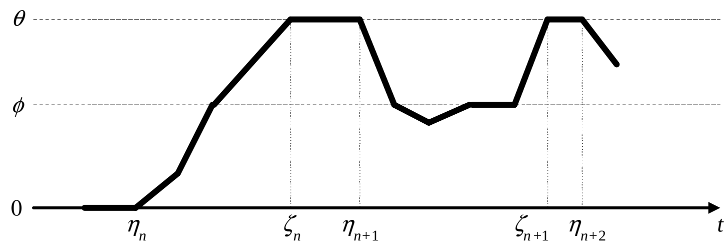

Figure 1.4. A typical sample path

We define the following random index sets:

$$\Psi_F(c)=\{n:\ x(t)=\theta \text{ for all } t\in[\zeta_{n-1},\eta_n),\ n=1,\ldots,N\} \tag{1.16}$$

$$\Psi_E(c)=\{n:\ x(t)=0 \text{ for all } t\in[\zeta_{n-1},\eta_n),\ n=1,\ldots,N\} \tag{1.17}$$

so that if $n\in\Psi_F$, the nth BP (which immediately precedes the nth NBP) is a FP; if $n\notin\Psi_F$, the nth BP (which immediately precedes the nth NBP) is an EP.

Next, we identify events of interest in the SFM : (i) A jump in $\alpha(t)$ or $\beta(t)$ is termed an *exogenous* event, reflecting the fact that its occurrence time is independent of the controllable parameter c, and (ii) The buffer content $x(t)$ reaches any one of the critical values $0,\phi$ or θ; this is termed an *endogenous* event to reflect the fact that its occurrence time generally depends on c. Note that the combination of these events and the continuous dynamics in (1.11) gives rise to a stochastic hybrid system model of the underlying discrete event system of Fig. 1.3.

Finally, we further decompose the sample path according to the events defined above. Let us consider a typical NBP $[\eta_n,\zeta_n)$ as shown in Fig. 1.5. Let $\pi_{n,i}$ denote times when $x(t)$ reaches or leaves $0,\phi$ or θ in this NBP, where $i=0,1,\ldots,I_n-1$, in which I_n is the number of such events in $[\eta_n,\zeta_n)$. Note that the starting point of the NBP is $\eta_n=\pi_{n,0}$. To maintain notational consistency we also set $\zeta_n=\pi_{n,I_n}$ even though this

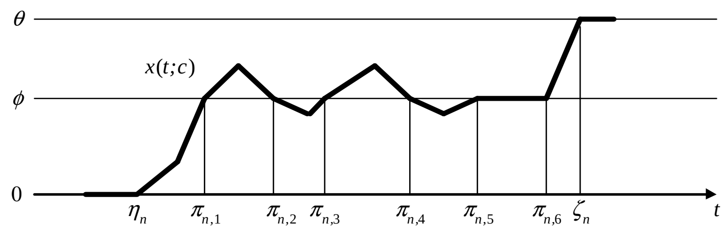

Figure 1.5. A typical NBP

point is not included in $[\eta_n, \zeta_n)$. We can now see that a sample path is decomposed into five sets of intervals that we shall refer to as the *modes* of the SFM: (i) Mode 0 is the set M_0 of all EPs contained in the sample path, (ii) Mode 1 is the set M_1 of intervals $[\pi_{n,i}, \pi_{n,i+1})$ such that $x(\pi_{n,i}) = 0$ or ϕ and $0 < x(t) < \phi$ for all $t \in (\pi_{n,i}, \pi_{n,i+1})$, $n = 1, \ldots, N$, (iii) Mode 2 is the set M_2 of intervals $[\pi_{n,i}, \pi_{n,i+1})$ such that $x(t) = \phi$ for all $t \in [\pi_{n,i}, \pi_{n,i+1})$, $n = 1, \ldots, N$, (iv) Mode 3 is the set M_3 of intervals $[\pi_{n,i}, \pi_{n,i+1})$ such that $x(\pi_{n,i}) = \phi$ or θ and $\phi < x(t) < \theta$ for all $t \in (\pi_{n,i}, \pi_{n,i+1})$, $n = 1, \ldots, N$ and (v) Mode 4 is the set M_4 of all FPs contained in the sample path. Note that the events occurring at times $\pi_{n,i}$ are all endogenous for $i = 1, \ldots, I_n$ and we should express them as $\pi_{n,i}(c)$ to stress this fact; for notational economy, however, we will only write $\pi_{n,i}$. Finally, recall that for $i = 0$, we have $\pi_{n,0} = \eta_n$, corresponding to an exogenous event starting the nth NBP. As shown in Fig. 1.5, the NBP $[\eta_n, \zeta_n)$ is decomposed into $I_n = 7$ intervals, including three M_1 intervals $[\pi_{n,0}, \pi_{n,1}), [\pi_{n,2}, \pi_{n,3}), [\pi_{n,4}, \pi_{n,5})$, three M_3 intervals $[\pi_{n,1}, \pi_{n,2}), [\pi_{n,3}, \pi_{n,4}), [\pi_{n,6}, \pi_{n,7})$, and one M_2 interval $[\pi_{n,5}, \pi_{n,6})$.

4.2 IPA Gradient Estimates for Performance Metrics and Their Unbiasedness

In this section we present the IPA gradient estimates for performance metrics and prove their unbiasedness.

THEOREM 1.1 *The IPA estimator of* $dE\left[H_T(c)\right]/dc$ *is:*

$$\frac{\partial H_T}{\partial c} = \frac{1}{T} \sum_{n \in \Psi_E} B(\zeta_{n-1}) \frac{\partial \zeta_{n-1}}{\partial c} \tag{1.18}$$

in which $\frac{\partial \zeta_{n-1}}{\partial c}$ *is given by Lemma 1.10.*

Proof. See Appendix. ■

Theorem 1.2 *The IPA estimator of $dE\left[L_T(c)\right]/dc$ is:*

$$\frac{dL_T(c)}{dc} = \frac{1}{T}\sum_{n\in\Psi_F}\left\{-A(\zeta_{n-1})\frac{\partial\zeta_{n-1}}{\partial c} + \frac{1}{c}H(\zeta_{n-1},\eta_n)\right\} + \frac{L_T}{c} \tag{1.19}$$

in which $\frac{\partial\zeta_{n-1}}{\partial c}$ *is given by Lemma 1.10.*

Proof. See Appendix. ■

In addition, we define the Suppression Traffic Volume to be the average volume which is denied admission before entering the system and denote it by R_T:

$$R_T = \frac{1}{T}\int_0^T \mathbf{1}\left[x(t)\geq\phi\right]\left[\alpha(t)-u(t)\right]dt$$

Theorem 1.3 *The IPA estimator of $dE\left[R_T(c)\right]/dc$ is:*

$$\frac{\partial R_T}{\partial c} = \frac{1}{T}\left\{-\sum_{i\in M_2}\left\{\left[\alpha(\pi_i)-\beta(\pi_i)\right]\frac{\partial\pi_i}{\partial c}\right\}\right.$$
$$\left.+\sum_{i\in M_3\cup M_4}\left[(1-c)\alpha(\pi_{i+1})\frac{\partial\pi_{i+1}}{\partial c}-(1-c)\alpha(\pi_i)\frac{\partial\pi_i}{\partial c}-\int_{\pi_i}^{\pi_{i+1}}\alpha(t)dt\right]\right\} \tag{1.20}$$

in which $\frac{\partial\pi_i}{\partial c}$ *is given by lemmas 1.5-1.9 in the following section.*

Proof. See Appendix. ■

Note that in our previous work on SFM-based IPA (see Cassandras et al., (2002), Sun et al., (2003), Yu and Cassandras (2003)), only raw data from a network node are required for IPA estimation, such as detecting a FP or an EP. However, the IPA estimators (1.18), (1.19) and (1.20) rely on flow rates and mode identification, which are all defined in a SFM context, making it less obvious to find their analogs in an actual node. Specifically, when the buffer level reaches ϕ and $c\alpha(t)\leq\beta(t)\leq\alpha(t)$, the SFM enters Mode 2 and the buffer level should stay at ϕ until this condition no longer applies. However, in the underlying DES, the buffer level will oscillate around ϕ instead and we must carefully define such a chattering interval so that it corresponds to Mode 2 of the SFM. As a result, errors in the recursive calculation of the event time sample derivatives $\partial\pi_i/\partial c$ may be introduced. To minimize the effect of such errors, we make use of the estimators whose form involves $\partial\pi_i/\partial c$ the least. For example, the IPA estimator for the throughout sensitivity,

$dH_T(c)/dc$, can either be directly evaluated by (1.18) or derived indirectly through estimators of $dL_T(c)/dc$ and $dR_T(c)/dc$ as follows. Recall the flow balance equation

$$H_T(c) + L_T(c) + R_T(c) = \bar{\alpha}$$

where $\bar{\alpha}$, the time average of the defining process $\alpha(t)$ over $[0, T]$, is independent of c. The above equation then gives

$$\frac{\partial H_T}{\partial c} = -\frac{\partial L_T}{\partial c} - \frac{\partial R_T}{\partial c} \tag{1.21}$$

The two evaluations (1.18) and (1.21) are equivalent in the SFM. But because of the discrepancy between DES and SFM, they may yield different results when applied to an actual network node. We select the latter estimation option for the following reason. The direct estimation of dH_T/dc in (1.18) entirely relies on the evaluation of event time sample derivatives $\partial\pi_i/\partial c$. As mentioned above, evaluating these from actual network data may introduce errors. On the other hand, the second and last terms in (1.19) can be directly observed from a DES sample path by counting the number of departures and number of packets dropped when the buffer is full; only the first term still involves event time derivatives. Similarly, the last term of dR_T/dc in (1.20) can be evaluated directly from actual network data. Recall that the inflow rate is $u(t) = c\alpha(t)$ when $x > \phi$. Thus, $\int_{\pi_i}^{\pi_{i+1}} c\alpha(t)dt$ is the inflow volume when $x > \phi$ and the last term in (1.20) can be obtained from the incoming packet volume divided by c when the buffer level is above or equal to ϕ. In this way, raw network data can be partially used in this indirect estimation approach.

Next we establish the unbiasedness of the IPA estimators (1.18), (1.19) and (1.20). First we make the following assumption:

ASSUMPTION 1 *For every c, w.p.1, no two events (either exogenous or endogenous) occur at the same time.*

This assumption precludes a situation where the queue content reaches one of the critical threshold values $0, \phi$ or θ at the same time π_i as an exogenous event which might cause it to leave the threshold; this would prevent the existence of the event time sample derivative $\partial\pi_i/\partial c$ which will be derived in the sequel (however, one could still carry out perturbation analysis with one-sided derivatives as in Cassandras et al., (2002)). Moreover, by Assumption 1, N, the number of NBPs in the sample path, is locally independent of c (since no two events may occur simultaneously, and the occurrence of exogenous events does not depend on c, there exists a neighborhood of c within which, w.p.1, the number of NBPs in $[0, T]$ is constant). Hence, the random index set Ψ_F is also locally independent of c. Similarly, the decomposition of the sample path into modes is also locally independent of c.

Normally, the unbiasedness of an IPA derivative $d\mathcal{L}(\theta)/d\theta$ for some performance metric $\mathcal{L}(\theta)$ is ensured by the following two conditions (see Rubinstein and Shapiro (1993), Lemma A2, p.70): (i) For every $\theta \in \tilde{\Theta}$, the sample derivative exists w.p.1, and (ii) W.p.1, the random function $\mathcal{L}(\theta)$ is Lipschitz continuous throughout $\tilde{\Theta}$, and the (generally random) Lipschitz constant has a finite first moment. Based on the monotonicity properties that will be established in Section 4.4, we can readily verify the two conditions and consequently establish the unbiasedness of the IPA estimators in the following theorem.

THEOREM 1.4 *Under Assumption 1, the IPA estimators (1.18), (1.19) and (1.20) are unbiased, i.e.,*

$$\frac{\partial E[L_T(c)]}{\partial c} = E\left[\frac{\partial L_T(c)}{\partial c}\right], \quad \frac{\partial E[H_T(c)]}{\partial c} = E\left[\frac{\partial H_T(c)}{\partial c}\right]$$

and

$$\frac{\partial E[R_T(c)]}{\partial c} = E\left[\frac{\partial R_T(c)}{\partial c}\right]$$

Proof. See Appendix. ■

4.3 Event Time Sample Derivatives

In this section, we derive sample derivatives for event times, which are necessary in the process of obtaining IPA estimators for performance metrics.

First we make the following additional assumptions:

ASSUMPTION 2 *$\alpha(t)$ and $\beta(t)$ are piecewise constant functions that can take a finite number of values.*

This assumption can be regarded as an approximation of general time-varying processes. As we will see later, we do not set any upper bound on the numbers of values that $\alpha(t)$ and $\beta(t)$ can possibly take, essentially allowing the piecewise constant process to approximate a general time-varying process as close as possible. The assumption is brought in mostly for ease of analysis. Due to this assumption and recalling the dynamics in (1.11), $x(t)$ has to be a piecewise linear function of time t, as shown in Fig. 1.4.

Assumption 3 *W.p.1, there exists an arbitrarily small positive constant ϵ such that for all t, $|\alpha(t) - \beta(t)| \geq \epsilon > 0$ and for a fixed c:*

$$|c\alpha(t) - \beta(t)| \geq \epsilon > 0$$

Combining the above two assumptions, we obtain for every pair of possible values of $\alpha(t)$ indexed by i and $\beta(t)$ indexed by j:

$$|c\alpha_i - \beta_j| \geq \epsilon$$

which is equivalent to

$$c\alpha_i - \beta_j \geq \epsilon \text{ or } c\alpha_i - \beta_j \leq -\epsilon$$

Therefore we obtain

$$c \geq \frac{\beta_j + \epsilon}{\alpha_i} \text{ or } c \leq \frac{\beta_j - \epsilon}{\alpha_i}$$

which implies an "invalid interval" $\left(\frac{\beta_j - \epsilon}{\alpha_i}, \frac{\beta_j + \epsilon}{\alpha_i}\right)$ for c. According to Assumption 2, there are a finite number of such invalid intervals. We shall also refer to a *valid interval* as the maximal interval between two adjacent invalid intervals. In what follows, we shall concentrate on a typical NBP $[\eta_n, \zeta_n(c))$ and drop the index n from the event times $\pi_{n,i}$ in order to simplify notation.

Assumptions 2-3 are needed to ensure the existence of the sample derivatives $\partial\pi_i/\partial c$, but they can be significantly weakened by simply assuming that w.p. 1 an event such that $c\alpha(t) - \beta(t)$ changes sign cannot coincide with any endogenous event (e.g., $x(t)$ reaches the level θ). This weaker condition introduces some technical complications in the derivations that follow which we will choose to avoid here by restricting ourselves to piecewise constant rate processes satisfying the last two assumptions.

In the rest of this section, we derive the sample derivative $\partial\pi_i/\partial c$ through a series of lemmas which cover all possible values that $x(\pi_i; c)$ can take in an interval $[\pi_i, \pi_{i+1})$.

Lemma 1.5 *Under Assumptions 4-6, if a FP ends at time η_n, i.e., $x(\eta_n; c) = \theta$, then*

$$\frac{\partial \eta_n}{\partial c} = 0$$

Proof. See Appendix. ■

LEMMA 1.6 *Under Assumptions 4-6, if an EP ends at time η_n, i.e., $x(\eta_n; c) = 0$, then*

$$\frac{\partial \eta_n}{\partial c} = 0$$

Proof. See Appendix. ■

The above two lemmas show that an event time perturbation is always eliminated after a NBP ends. The following lemma further asserts that the same is true after a finite interval during which $x(t; c) = \phi$.

LEMMA 1.7 *Under Assumptions 4-6, if an M_2 interval ends at time π_i, $x(\pi_i; c) = \phi$, then*

$$\frac{\partial \pi_i}{\partial c} = 0$$

Proof. See Appendix. ■

Next, we define the following shorthand notation:

$$A(t) = c\alpha(t) - \beta(t) \quad \text{and} \quad B(t) = \alpha(t) - \beta(t)$$

LEMMA 1.8 *Under Assumptions 4-6, if $[\pi_i, \pi_{i+1}) \in M_3$, then*

$$\frac{\partial \pi_{i+1}}{\partial c} = \frac{A(\pi_i^+)}{A(\pi_{i+1}^-)} \cdot \frac{\partial \pi_i}{\partial c} - \frac{1}{A(\pi_{i+1}^-)} \int_{\pi_i}^{\pi_{i+1}} \alpha(\tau) d\tau \tag{1.22}$$

Proof. See Appendix. ■

Define

$$H(\pi_i, t) = \int_{\pi_i}^{t} \delta(t) dt \tag{1.23}$$

as the node throughput during time interval $[\pi_i, t)$. In addition, we have the following flow balance equation:

$$\int_{\pi_i}^{\pi_{i+1}} c\alpha(t) dt - H(\pi_i, \pi_{i+1}) = x(\pi_{i+1}; c) - x(\pi_i; c)$$

which gives

$$\int_{\pi_i}^{\pi_{i+1}} \alpha(t) dt = \frac{x(\pi_{i+1}; c) - x(\pi_i; c) + H(\pi_i, \pi_{i+1})}{c}. \tag{1.24}$$

Combining Lemma 1.8 and (1.24) gives

$$\frac{\partial \pi_{i+1}}{\partial c} = \frac{A(\pi_i^+)}{A(\pi_{i+1}^-)} \cdot \frac{\partial \pi_i}{\partial c} - \frac{x(\pi_{i+1}; c) - x(\pi_i; c) + H(\pi_i, \pi_{i+1})}{cA(\pi_{i+1}^-)} \tag{1.25}$$

where

$$[x(\pi_{i+1}; c) - x(\pi_i; c)] \in \{\phi - \theta, 0, \theta - \phi\}.$$

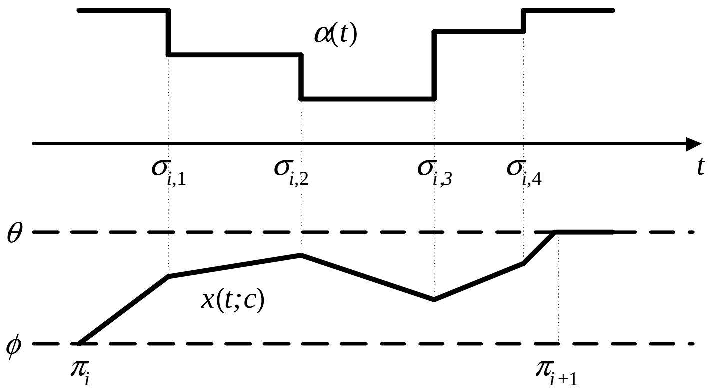

Figure 1.6. The Decomposition of an M_3 Interval

According to Assumption 2, $\alpha(t)$ and $\beta(t)$ are piecewise constant functions. The interval $[\pi_i, \pi_{i+1})$ can be then decomposed by exogenous events occurring when $\alpha(t)$ jumps from one value to another.

As shown in Fig. 1.6, we use $\sigma_{i,k}, k = 1, \ldots S_i$ to denote the kth such exogenous event and let $\pi_i = \sigma_{i,0}$ and $\pi_{i+1} = \sigma_{i,S_i+1}$ in order to maintain notational consistency. Moreover we define the value of $\alpha(t)$ in interval $[\sigma_{i,k}, \sigma_{i,k+1})$ as $\alpha_{i,k}$. It follows that

$$\int_{\pi_i}^{\pi_{i+1}} \alpha(t)dt = \sum_{k=0}^{S_i} \alpha_{i,k} \left(\sigma_{i,k+1} - \sigma_{i,k}\right)$$

If we use the following shorthand

$$b_{i,k} = \sigma_{i,k+1} - \sigma_{i,k} \text{ for all } i, k, \tag{1.26}$$

to define the length of an interval between two exogenous $\alpha(t)$ jump events, we get $\int_{\pi_i}^{\pi_{i+1}} \alpha(t)dt = \sum_{k=0}^{S_i} \alpha_{i,k} b_{i,k}$. Then, (1.22) becomes

$$\frac{\partial \pi_{i+1}}{\partial c} = \frac{A(\pi_i^+)}{A(\pi_{i+1}^-)} \cdot \frac{\partial \pi_i}{\partial c} - \frac{\sum_{k=0}^{S_i} \alpha_{i,k} b_{i,k}}{A(\pi_{i+1}^-)} \tag{1.27}$$

Similar to the work in Yu and Cassandras (2003), our ultimate purpose is to apply the IPA estimators (which we will derive in next section based on event time sample derivatives) to an actual underlying DES. The three expressions (1.22), (1.25) and (1.27) provide alternative ways to evaluate the event time sample derivative which are equivalent in the SFM context. In the discrete-event setting, however, some information required for IPA estimation may be more difficult to obtain than other. For example, (1.27) depends on the evaluation of $\alpha_{i,k}$, the maximal incoming rate and $b_{i,k}$, the length of intervals between two $\alpha(t)$ jump events. This information may be difficult to acquire or measure if the source is remote. On the other hand, (1.25) requires a throughput evaluation during the time interval $[\pi_i, \pi_{i+1})$, which may be much easier to obtain, i.e., in an actual network node, it can be done by simply counting processed packets. In summary, we want to remind readers that different forms of IPA estimators exist and that one should select the appropriate one based on implementation considerations. We also point out that (1.22) can be further simplified when the service rate $\beta(t) = \beta$ is constant:

$$\frac{\partial \pi_{i+1}}{\partial c} = \frac{c\alpha(\pi_i^+) - \beta}{c\alpha(\pi_{i+1}^-) - \beta} \cdot \frac{\partial \pi_i}{\partial c} - \frac{[x(\pi_{i+1}; c) - x(\pi_i; c)] + \beta(\pi_{i+1} - \pi_i)}{c^2\alpha(\pi_{i+1}^-) - c\beta}$$

In this case, only $\pi_{i+1} - \pi_i$, the length of the Mode 3 interval has to be evaluated.

LEMMA 1.9 *If* $[\pi_i, \pi_{i+1}) \in M_1$,

$$\frac{\partial \pi_{i+1}}{\partial c} = \frac{B(\pi_i^+)}{B(\pi_{i+1}^-)} \cdot \frac{\partial \pi_i}{\partial c} \tag{1.28}$$

Proof. See Appendix. ■

The combination of Lemmas 1.5 through 1.9 provides a linear recursive relationship for obtaining the event time sample derivative $\partial \pi_i / \partial c$, and the coefficients involved are based on information directly available from a sample path of the SFM and the throughput given in (1.23). Moreover, $\frac{\partial \zeta_n}{\partial c}$, the event time sample derivatives for the end of a NBP $[\eta_n, \zeta_n)$, can also be derived by combining the above lemmas. Recall that $\zeta_n = \pi_{n,I_n}$. Using the previous lemmas, we can also obtain a recursive expression for $\frac{\partial \zeta_n}{\partial c}$ as follows:

LEMMA 1.10 *For a NBP* $[\eta_n, \zeta_n)$,

$$\frac{\partial \zeta_n}{\partial c} = \begin{cases} \frac{B(\pi_{n,I_n-1}^+)}{B(\pi_{n,I_n}^-)} \cdot \frac{\partial \pi_{n,I_n-1}}{\partial c} & \textit{if } x(\zeta_n; c) = 0 \\ -\frac{H(\pi_{n,I_n-1}, \pi_{n,I_n})}{cA(\pi_{n,I_n}^-)} & \textit{if } x(\zeta_n; c) = \theta \textit{ and } x(\pi_{n,I_n-1}; c) = \theta \\ \frac{A(\pi_{n,I_n-1}^+)}{A(\pi_{n,I_n}^-)} \cdot \frac{\partial \pi_{n,I_n-1}}{\partial c} - \frac{\theta - \phi + H(\pi_{n,I_n-1}, \pi_{n,I_n})}{cA(\pi_{n,I_n}^-)} & \textit{if } x(\zeta_n; c) = \theta \textit{ and } x(\pi_{n,I_n-1}; c) = \phi \end{cases} \tag{1.29}$$

Proof. See Appendix. ■

To summarize, event time sensitivities $\frac{\partial \pi_i}{\partial c}$ are triggered only in Mode 3 through the second term in (1.22). The sensitivities are subsequently reset to zero after a Mode 0 (EP), Mode 2, or Mode 4 (FP) interval. With the help of the five lemmas derived above, we are now able to derivative IPA estimators for various performance metrics in the following section.

4.4 Monotonicity and Lipschitz Continuity of $x(t; c)$ with respect to c

In this section we establish some monotonicity properties that are critical in the proof of unbiasedness (Theorem 1.4). As mentioned before, the buffer content is a function of c, i.e., $x(t) = x(t; c)$. In this section we establish the monotonicity and Lipschitz continuity of the function $x(t; c)$ with respect to the parameter c. As we will show later, this property is critical in proving unbiasedness of IPA estimators. We first establish this result for SFMs with the general feedback scheme introduced in Section 2, and verify its applicability to the specific multiplicative feedback mechanism of Section 3.

Consider a SFM with feedback defined in Section 2 where c is a generic controllable parameter of the traffic shaper. We assume that c and θ are independent of each other. As mentioned before, $u(t)$, the actual inflow rate, $\delta(t)$, the outflow rate and $\gamma(t)$ are all functions of some defining process $\alpha(t)$, queue content $x(t)$, thresholds θ, ϕ, and c, the parameter of interest. But for notational simplicity we will suppress these dependencies unless it is necessary.

We make the following assumptions on the dependence of $u(t)$ on $x(t; c)$ and c. Based on these assumptions we can establish the monotonicity and Lipschitz continuity of $x(t; c)$ for SFMs with general negative feedback. We will subsequently verify them for the multiplicative feedback mechanism introduced in Section 3.

Assumption 4 *For any fixed t and c, $u(t, x; c)$ is a monotonically nonincreasing function of x, i.e., when $x_1 > x_2$, $u(t, x_1; c) \leq u(t, x_2; c)$ for all t and c.*

Assumption 5 *For any fixed t and x, $u(t, x; c)$ is a monotonically nondecreasing function of c.*

Assumption 6 *$u(t, x; c)$ is Lipschitz continuous with respect to c, i.e.*

$$|u(t, x; c + \Delta c) - u(t, x; c)| \leq K\Delta c$$

in which K is the Lipschitz constant.

We assume that for all c, $x(t;c)$ is a continuous function with respect to t, and $x(t;c) = 0$ when $t = 0$. For notational simplicity, we use SFM_N to denote the state trajectory of the nominal system under parameter c and SFM_P to denote its perturbed counterpart under parameter $c+\Delta c$. Throughout this section, for a function $f(\cdot)$ we use $f'(\cdot)$ to represent $f(c+\Delta c)$, the corresponding function in SFM_P, while $f(\cdot)$ represents $f(c)$, the corresponding function in SFM_N. Thus, the buffer level is denoted by $x(t)$ in SFM_N and $x'(t)$ in SFM_P.

Define *buffer level perturbations* as

$$\Delta x(t) = x(t; c + \Delta c) - x(t; c) = x'(t) - x(t)$$

with respect to a perturbation Δc. The following lemma establishes monotonicity:

LEMMA 1.11 *Under Assumption 5, for any* $\Delta c > 0$,

$$\Delta x(t) \geq 0 \text{ For all } t \geq 0 \tag{1.30}$$

Proof. See Appendix. ■

In order to establish the Lipschitz continuity, first we present the following lemma leading to Theorem 1.13:

LEMMA 1.12 *Define*

$$\Delta\delta(t) = \delta(t, x'(t); c + \Delta c) - \delta(t, x(t); c) = \delta'(t) - \delta(t)$$

and

$$\Delta\gamma(t) = \gamma(t, x'(t); c + \Delta c) - \gamma(t, x(t); c) = \gamma'(t) - \gamma(t).$$

Then under Assumption 5, for any $\Delta c > 0$,

$$\Delta\delta(t) \geq 0, \ \Delta\gamma(t) \geq 0$$

Proof. See Appendix. ■

THEOREM 1.13 *Under Assumptions 4-6,*

$$\Delta x(t) \leq KT\Delta c$$

Proof. See Appendix. ■

It is also easy to verify the above results for $\Delta c < 0$.

In order to establish that the general results in this section cover the feedback mechanism defined in (1.12), i.e., to prove that Lemma 1.11 and Theorem 1.13 hold for $u(t)$ in (1.12), we need to verify Assumptions 4-6.

For Assumption 4, assume $x_1 < x_2$. There are three possible cases: (i) if $x_1 < \phi$, $u(t, x_1; c) = \alpha(t) \geq u(t, x_2; c)$; (ii) if $x_1 = \phi < x_2$, $u(t, x_2; c) = c\alpha(t) \leq u(t, x_1; c)$; (iii) if $\phi < x_1 < x_2$, $u(t, x_1; c) = u(t, x_2; c) = c\alpha(t)$. Therefore, for any x_1, x_2, if $x_1 < x_2$, $u(t; x_1; c) \geq u(t; x_2; c)$ and the assumption is verified.

For Assumptions 5 and 6, there are again three possible cases:

Case 1. if $x < \phi$, $u(t, x; c) = u(t, x; c + \Delta c) = \alpha(t)$, which gives $u(t, x; c + \Delta c) - u(t, x; c) = 0$;

Case 2. if $x > \phi$, $u(t, x; c) = c\alpha(t)$, $u(t, x; c + \Delta c) = (c + \Delta c)\alpha(t)$, which gives $u(t, x; c + \Delta c) - u(t, x; c) = \alpha(t) \cdot \Delta c \geq 0$;

Case 3. if $x = \phi$, since $c\alpha(t) \leq (c + \Delta c)\alpha(t) \leq \alpha(t)$, there are four cases to consider regarding the relative values of $c\alpha(t), (c+\Delta c)\alpha(t), \alpha(t)$ and $\beta(t)$:

Case 3.1. $c\alpha(t) \leq (c + \Delta c)\alpha(t) \leq \alpha(t) \leq \beta(t)$, $u(t, x; c) = u(t, x; c + \Delta c) = \alpha(t)$. It follows that $u(t, x; c + \Delta c) - u(c; t, x) = 0$;

Case 3.2. $c\alpha(t) \leq (c + \Delta c)\alpha(t) \leq \beta(t) \leq \alpha(t)$, $u(t, x; c) = u(t, x; c + \Delta c) = \beta(t)$. It follows that $u(t, x; c + \Delta c) - u(c; t, x) = 0$;

Case 3.3. $c\alpha(t) \leq \beta(t) \leq (c + \Delta c)\alpha(t) \leq \alpha(t)$, $u(t, x; c) = \beta(t)$ and $u(c + \Delta c; t, x'(t)) = (c + \Delta c)\alpha(t)$, so that $u(t, x; c + \Delta c) \geq u(t, x; c)$. Moreover, it follows that

$$\begin{aligned} u(t, x; c + \Delta c) - u(t, x; c) =& (c + \Delta c)\alpha(t) - \beta(t) \\ \leq& (c + \Delta c)\alpha(t) - c\alpha(t) \\ =& \alpha(t) \cdot \Delta c \end{aligned}$$

Case 3.4. $\beta(t) \leq c\alpha(t) \leq (c + \Delta c)\alpha(t) \leq \alpha(t)$, $u(t, x; c) = u(t, x; c + \Delta c) = c\alpha(t)$. It follows that $u(t, x; c + \Delta c) - u(t, x; c) = 0$.

Combining all of the above cases verifies Assumption 5. In order to verify Assumption 6, note that $u(t, x; c+\Delta c) - u(t, x; c) \leq \alpha(t) \cdot \Delta c$ from the above cases. Recalling that we have assumed the process $\{\alpha(t)\}$ to be such that w.p. 1 $\alpha(t) \leq \alpha_{\max} < \infty$, it follows that

$$u(t, x; c + \Delta c) - u(t, x; c) \leq \alpha(t) \cdot \Delta c \leq \alpha_{\max} \cdot \Delta c \tag{1.31}$$

Hence Assumption 6 is also verified with the Lipschitz constant $K = \alpha_{\max}$. Therefore, Lemma 1.11 and Theorem 1.13 hold for the feedback mechanism (1.10).

5. Optimization Examples

In this section we present some numerical examples to illustrate how the IPA estimators we have developed are used in optimization problems. As suggested before, the solution to an optimization problem defined for an actual queueing system may be approximated by the solution to the same problem based on a SFM of the system. Let us now consider the feedback-based buffer control problem defined in Section 3 with cost function (1.15):

$$J_T^{DES}(c) = E[H_T^{DES}(c)] - \lambda \cdot E[L_T^{DES}(c)]$$

The optimal value of c which maximizes $J_T^{DES}(c)$ above may be determined through a standard stochastic approximation algorithm (details on such algorithms, including conditions required for convergence to an optimum may be found, for instance, in Kushner and Yin (1997)):

$$c_{n+1} = c_n + \nu_n \mathcal{H}_n(c_n, \omega_n^{DES}), \qquad n = 0, 1, \dots \tag{1.32}$$

where $\mathcal{H}_n(c_n, \omega_n^{DES})$ is an estimate of dJ_T/dc evaluated at $c = c_n$ and $\{\nu_n\}$ is a step size sequence. $\mathcal{H}_n(\cdot)$, the form of the estimator, comes from SFM-based IPA analysis, i.e., from (1.18) and (1.19), but the data input to the estimator are based on a DES sample path denoted by ω_n^{DES}. Obviously, the resulting gradient estimator $\mathcal{H}_n(c_n, \omega_n^{DES})$ is now an approximation leading to a sub-optimal solution of the above optimization problem.

Note that, after a control update, the state must be reset to zero, in accordance with our convention that all performance metrics are defined over an interval with an initially empty buffer. In the case of off-line control (as in the numerical examples we present), this simply amounts to simulating the system after resetting its state to 0. In the more interesting case of on-line control, we proceed as follows. Suppose that the nth iteration ends at time τ_n and the state is $x(c_n; \tau_n)$ [in general, $x(c_n; \tau_n) > 0$]. At this point, the threshold is updated and its new value is c_{n+1}. Let τ_n^0 be the next time that the buffer is empty, i.e., $x(c_{n+1}; \tau_n^0) = 0$. At this point, the $(n+1)$th iteration starts and the next gradient estimate is obtained over the interval $[\tau_n^0, \tau_n^0 + T]$, so that $\tau_{n+1} = \tau_n^0 + T$ and the process repeats. The implication is that over the interval no estimation is carried out while the controller waits for the system to be reset to its proper initial state; therefore, sample path information available over $[\tau_n, \tau_n^0]$ is effectively wasted as far as gradient estimation is concerned.

Figure 1.7 shows examples of the application of (1.32) to a network node modeled as in Fig. 1.3 under two different parameter settings (scenarios). The service rate $\beta(t)$ remains constant throughout the simulation but $\alpha(t)$ is piecewise constant: it remains constant for an exponentially distributed period of time and when it switches the next value of $\alpha(t)$ is generated according to a transition probability matrix. For simplicity, we assume that all elements of the transition probability matrix are equal and the only feasible value of these elements is $q = 1/m$, in

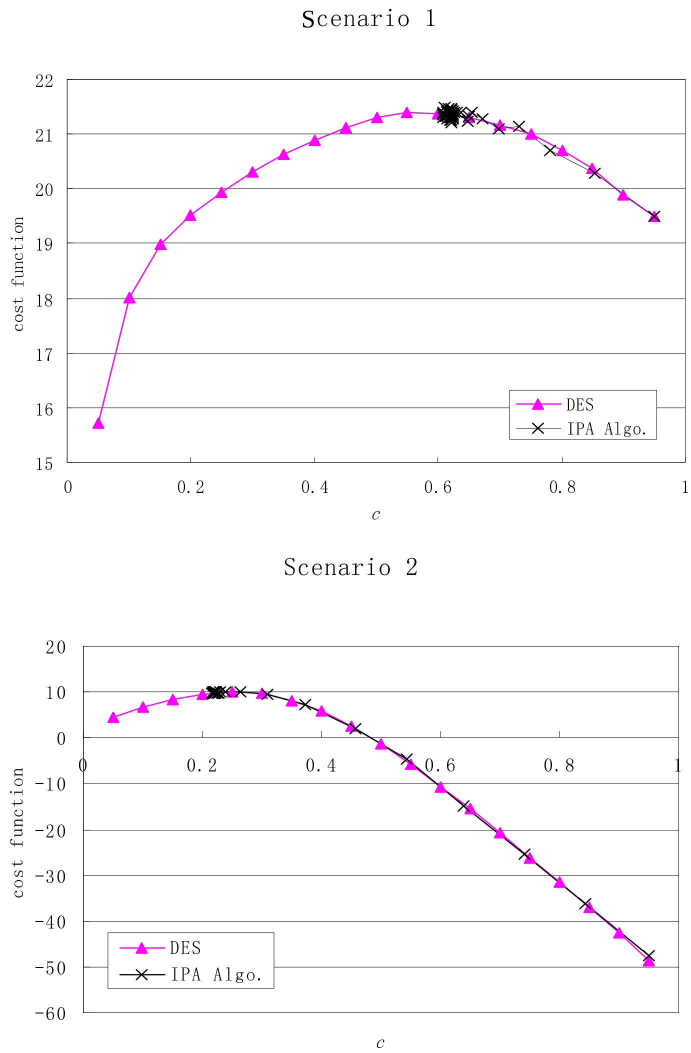

Figure 1.7. Numerical results for SFM-based gradient optimization of an actual network node

which m is the number of values $\alpha(t)$ can take. For different scenarios, $\alpha(t)$ value sets, the value of β, the initial value of feedback gain c_0 and overflow penalty λ also vary. Table 1 summarizes the settings for both scenarios. Also shown in the table are c_0, the initial feedback gain value, and c^*, the value obtained through (1.32). In Fig. 1.7, the curve "DES"

Scenario	θ	ϕ	λ	$q = \frac{1}{m}$	α value set	β	c_0	c^*
1	15	5	2	0.125	100,28,27,24,21,20,14,9	30	0.95	0.62
2	15	2	2	0.25	150,60,30,8	15	0.95	0.21

Table 1.1. Summary of parameter settings for two scenarios

denotes the cost function $J_T(c)$ obtained through exhaustive simulation for different (discrete) values of c with $T = 100000$; the curve "IPA Algo." represents the optimization process (1.32) with the simulation time horizon for each step of (1.32) set to $T' = 10000$, and with constant step size $\nu = 0.01$. As shown in Fig. 1.7, the gradient-based algorithm (1.32) converges to the neighborhood of the optimal feedback gain.

6. Conclusions and Future Directions

SFMs have recently been used to capture the dynamics of complex stochastic discrete event systems, such as computer networks, and to implement control and optimization methods based on gradient estimates of performance metrics obtained through IPA. In Yu and Cassandras (2003) we showed that IPA can be used in SFMs with additive feedback and here we have further explored the effect of feedback by considering a single-node SFM with a controllable inflow rate as a multiplicative function of the state (i.e., queue level) feedback parameterized by a feedback gain c and a threshold ϕ (capturing a quantization in the state feedback). We have developed IPA estimators for the loss volume and average workload with respect to the feedback gain parameter c and shown their unbiasedness, despite the complications brought about by the presence of feedback. This scheme bypasses the need for continuous state information seen in additive mechanisms and involves only knowledge of a single event representing the queue level crossing the threshold ϕ. Moreover, even if this state information is not instantaneously supplied, the delays involved are naturally built into the IPA estimator, based on which appropriate control parameters can be selected.

This work opens up a variety of possible extensions. First, looking at the feedback mechanism (1.12), note that c represents the feedback gain and ϕ represents the range. Instead of controlling c or ϕ separately (along the lines of previous work in Yu and Cassandras (2002)), it may be more effective to control the (c, ϕ) pair jointly. Next, noticing that probabilistic dropping/marking mechanisms are widely adopted in computer networks (e.g., in Random Early Detection or Random Early Marking), it is appealing to apply IPA specifically to these algorithms.

Finally, of obvious interest is the application of our SFM-based IPA estimators to an actual underlying DES such as the internet, i.e., to determine the value of c that minimizes a weighted sum of loss volume and average workload, as we have done in Cassandras et al., (2002) and Yu and Cassandras (2003). As mentioned earlier, one advantage of IPA is that the estimators depend only on data directly observable along a sample path of the actual DES (not just the SFM which is an abstraction of the system); see, for example, Cassandras et al., (2002) and Yu and Cassandras (2003). Here, however, we have seen that this direct connection to the DES no longer holds because the estimators rely on the identification of "modes" whose definition does not always have a direct correspondence to a DES. As a result, in order to successfully apply the SFM-based IPA estimators to an actual DES, we need to carefully select and interpret an appropriate abstraction of the underlying DES.

Acknowledgments

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

Appendix

Proof of Theorem 1.1. Recall the definition of throughput:

$$H_T = \frac{1}{T}\int_0^T \delta(t)dt.$$

Using (1.8), we can rewrite this equation as follows:

$$\begin{aligned} H_T =& \frac{1}{T}\left\{\sum_{i\in M_0}\int_{\pi_i}^{\pi_{i+1}} \alpha(t)dt + \sum_{i\notin M_0}\int_{\pi_i}^{\pi_{i+1}} \beta(t)dt\right\} \\ =& \frac{1}{T}\left\{\sum_{i\in M_0}\int_{\pi_i}^{\pi_{i+1}} \left[\alpha(t)-\beta(t)\right]dt + \sum_{i\notin M_0}\int_{\pi_i}^{\pi_{i+1}} \beta(t)dt + \sum_{i\in M_0}\int_{\pi_i}^{\pi_{i+1}} \beta(t)dt\right\} \\ =& \frac{1}{T}\left\{\sum_{i\in M_0}\int_{\pi_i}^{\pi_{i+1}} \left[\alpha(t)-\beta(t)\right]dt + \int_0^T \beta(t)dt\right\} \end{aligned}$$

Differentiating with respect to c we obtain:

$$\frac{\partial H_T}{\partial c} = \frac{1}{T}\sum_{i\in M_0}\left\{\left[\alpha(\pi_{i+1})-\beta(\pi_{i+1})\right]\frac{\partial \pi_{i+1}}{\partial c} + \left[\beta(\pi_i)-\alpha(\pi_i)\right]\frac{\partial \pi_i}{\partial c}\right\}$$

Note that, if $[\pi_i, \pi_{i+1}) \in M_0$, π_i is the start of an EP. Hence it is also the end of some NBP, i.e., $\pi_i = \zeta_{n-1}$ and $\pi_{i+1} = \eta_n$ for some $n \in \Psi_E$. Combining this with Lemma

1.6, we get $\partial \pi_{i+1}/\partial c = 0$. The above equation then becomes:

$$\begin{aligned}\frac{\partial H_T}{\partial c} =& \frac{1}{T}\sum_{i\in M_0}\left[\beta(\pi_i)-\alpha(\pi_i)\right]\frac{\partial \pi_i}{\partial c}\\ =& \frac{1}{T}\sum_{n\in\Psi_E}\left[\beta(\zeta_{n-1})-\alpha(\zeta_{n-1})\right]\frac{\partial \zeta_{n-1}}{\partial c}\end{aligned}$$

■

Proof of Theorem 1.2. If $[\zeta_{n-1},\eta_n)$ is a FP, we have $x(t)=\theta$ for all $t\in[\zeta_{n-1},\eta_n)$. Recalling that Ψ_F is locally independent of c, it follows from (1.14) that

$$\frac{dL_T(c)}{dc} = \frac{1}{T}\sum_{n\in\Psi_F}\left\{\left[c\alpha(\eta_n)-\beta(\eta_n)\right]\frac{\partial\eta_n}{\partial c}-\left[c\alpha(\zeta_{n-1})-\beta(\zeta_{n-1})\right]\frac{\partial\zeta_{n-1}}{\partial c}+\int_{\zeta_{n-1}}^{\eta_n}\alpha(t)dt\right\}$$

Combining the above equation with Lemma 1.5 it follows that

$$\frac{dL_T(c)}{dc} = \frac{1}{T}\sum_{n\in\Psi_F}\left\{-\left[c\alpha(\zeta_{n-1})-\beta(\zeta_{n-1})\right]\frac{\partial\zeta_{n-1}}{\partial c}+\int_{\zeta_{n-1}}^{\eta_n}\alpha(t)dt\right\}$$

Moreover, we have the flow balance equation

$$c\int_{\zeta_{n-1}}^{\eta_n}\alpha(t)dt=\int_{\zeta_{n-1}}^{\eta_n}\beta(t)dt+L_n+x(\eta_n;c)-x(\zeta_{n-1};c)$$

in which L_n is the lost volume because of overflow in the FP $[\zeta_{n-1},\eta_n)$. Noticing that $x(\eta_n;c)=x(\zeta_{n-1};c)=\theta$, we obtain:

$$\frac{dL_T(c)}{dc} = \frac{1}{T}\sum_{n\in\Psi_F}\left\{-\left[c\alpha(\zeta_{n-1})-\beta(\zeta_{n-1})\right]\frac{\partial\zeta_{n-1}}{\partial c}+\frac{1}{c}\left[\int_{\zeta_{n-1}}^{\eta_n}\beta(t)dt+L_n\right]\right\}$$

Since

$$\sum_{n\in\Psi_F}L_n = T\cdot L_T,$$

we obtain:

$$\begin{aligned}\frac{dL_T(c)}{dc} =& \frac{1}{T}\sum_{n\in\Psi_F}\left\{-\left[c\alpha(\zeta_{n-1})-\beta(\zeta_{n-1})\right]\frac{\partial\zeta_{n-1}}{\partial c}+\frac{1}{c}\int_{\zeta_{n-1}}^{\eta_n}\beta(t)dt\right\}+\frac{L_T}{c}\\ =& \frac{1}{T}\sum_{n\in\Psi_F}\left\{-\left[c\alpha(\zeta_{n-1})-\beta(\zeta_{n-1})\right]\frac{\partial\zeta_{n-1}}{\partial c}+\frac{1}{c}H(\zeta_{n-1},\eta_n)\right\}+\frac{L_T}{c}\end{aligned}$$

■

Proof of Theorem 1.3. According to (1.11) and (1.12), when the system is in Mode 2, the suppressed flow rate is $\alpha(t)-\beta(t)$; when the system is in Mode 3 or 4, the suppressed flow rate is $(1-c)\,\alpha(t)$. Therefore

$$R_T=\frac{1}{T}\left\{\sum_{i\in M_2}\int_{\pi_i}^{\pi_{i+1}}\left[\alpha(t)-\beta(t)\right]dt+\sum_{i\in M_3\cup M_4}\int_{\pi_i}^{\pi_{i+1}}(1-c)\,\alpha(t)dt\right\}$$

Differentiating with respect to c we obtain:

$$\frac{\partial R_T}{\partial c} = \frac{1}{T}\left\{\sum_{i\in M_2}\left\{[\alpha(\pi_{i+1})-\beta(\pi_{i+1})]\frac{\partial \pi_{i+1}}{\partial c} - [\alpha(\pi_i)-\beta(\pi_i)]\frac{\partial \pi_i}{\partial c}\right\} + \sum_{i\in M_3\cup M_4}\left[(1-c)\alpha(\pi_{i+1})\frac{\partial \pi_{i+1}}{\partial c} - (1-c)\alpha(\pi_i)\frac{\partial \pi_i}{\partial c} - \int_{\pi_i}^{\pi_{i+1}}\alpha(t)dt\right]\right\} \tag{1.A.1}$$

According to Lemma 1.7, $\frac{\partial \pi_{i+1}}{\partial c} = 0$ if $i \in M_2$. Thus we get (1.20). ■

Proof of Theorem 1.4. We prove the unbiasedness of the IPA derivatives by establishing that the unbiasedness Conditions (i) and (ii) are satisfied for the random functions $L_T(c)$, $H_T(c)$ and $R_T(c)$. Condition (i) is in force by Assumptions 4-6. Regarding Condition (ii), we have the following flow balance equations for SFM_N and SFM_P respectively:

$$x(T) - x(0) = \int_0^T u(t)dt - \int_0^T \delta(t)d\tau - \int_0^T \gamma(t)dt$$

and

$$x'(T) - x'(0) = \int_0^T u'(t)dt - \int_0^T \delta'(t)dt - \int_0^T \gamma'(t)dt$$

Combining the above equations and recalling the assumption that $x'(0) = x(0) = 0$, we obtain:

$$\int_0^T \Delta u(t)dt = \Delta x(T) + \int_0^T \Delta\delta(t)dt + \int_0^T \Delta\gamma(t)dt \tag{1.A.2}$$

According to Lemma 1.11, $\Delta x(T) \geq 0$. According to Lemma 1.12, $\Delta\delta(t) \geq 0$, $\Delta\gamma(t) \geq 0$. Therefore,

$$\int_0^T \Delta u(t)dt \geq 0.$$

Moreover,

$$\int_0^T \Delta\delta(t)dt \leq \int_0^T \Delta u(t)dt, \quad \int_0^T \Delta\gamma(t)dt \leq \int_0^T \Delta u(t)dt \tag{1.A.3}$$

Recall that

$$\Delta u(t) = u(t, x'(t); c + \Delta c) - u(t, x(t); c)$$

According to Assumption 5, $u(t, x'(t); c + \Delta c) \leq u(t, x(t); c + \Delta c)$. Thus,

$$\Delta u(t) \leq u(t, x(t); c + \Delta c) - u(t, x(t); c)$$

According to (1.31), $u(t, x(t); c + \Delta c) - u(t, x(t); c) \leq \alpha_{\max} \cdot \Delta c$. Hence,

$$\Delta u(t) \leq \alpha_{\max} \cdot \Delta c$$

which gives

$$\int_0^T \Delta u(t)dt \leq \alpha_{\max} T \cdot \Delta c$$

Therefore, from (1.A.3) we get

$$\int_0^T \Delta\gamma(t)dt \leq \int_0^T \Delta u(t)dt \leq \alpha_{\max} T \cdot \Delta c$$

and

$$\int_0^T \Delta\delta(t)dt \leq \int_0^T \Delta u(t)dt \leq \alpha_{\max} T \cdot \Delta c$$

In other words, both $L_T(c)$ and $H_T(c)$ are Lipschitz continuous. For $R_T(c)$, recall the flow balance equation

$$\bar{\alpha} = L_T(c) + H_T(c) + R_T(c),$$

where

$$\bar{\alpha} = \frac{1}{T}\int_0^T \alpha(t)dt,$$

is the time average of $\alpha(t)$, independent of c. Hence

$$R_T(c) = \bar{\alpha} - L_T(c) - H_T(c)$$

is also Lipschitz continuous. This completes the proof. ■

Proof of Lemma 1.5. If $x(t)$ decreases from θ at time η_n, this defines the start of a NBP. From (1.11) and (1.12) we must have $c\alpha(\eta_n^-)-\beta(\eta_n^-) \geq 0$ and $c\alpha(\eta_n^+)-\beta(\eta_n^+) < 0$. From Assumption 3 we know that $c\alpha(\eta_n^-) - \beta(\eta_n^-) > \epsilon$, $c\alpha(\eta_n^+) - \beta(\eta_n^+) < \epsilon$. Recalling Assumption 2, we conclude that a jump in $\alpha(t)$ or $\beta(t)$ occurs at time η_n. Since $\alpha(t)$ or $\beta(t)$ are independent of c, Assumption 1 implies that there exists a neighborhood of c within which a change of c does not affect η_n. This implies that η_n is locally independent of c and the result follows. ■

Proof of Lemma 1.6. The proof is similar to that of the previous lemma, with $\alpha(\eta_n^-) - \beta(\eta_n^-) \geq 0$ and $\alpha(\eta_n^+) - \beta(\eta_n^+) < 0$. ■

Proof of Lemma 1.7. At the end of an M_2 interval, $x(t)$ may either increase or decrease from ϕ. On one hand, $x(t)$ increasing from ϕ at time π_i defines the start of a M_3 interval. Specifically, from (1.11) and (1.12) we have $c\alpha(\pi_i^-) \leq \beta(\pi_i^-) \leq \alpha(\pi_i^-)$ and $c\alpha(\pi_i^+) > \beta(\pi_i^+)$. Since $\alpha(t)$ and $\beta(t)$ are independent of c, we conclude that an exogenous event occurs at time π_i. Moreover, from Assumption 3 we know that there exists a neighborhood of c within which a change of c does not affect π_i. This implies that π_i is locally independent of c. On the other hand, $x(t)$ decreasing from ϕ at time π_i defines the start of a M_1 interval. Specifically from (1.11) we have $c\alpha(\pi_i^-) \leq \beta(\pi_i^-) \leq \alpha(\pi_i^-)$ and $c\alpha(\pi_i^+) > \beta(\pi_i^+)$, which implies that π_i is the occurrence time of an exogenous event and therefore locally independent of c. The result follows when we combine the above arguments. ■

Proof of Lemma 1.8. If $[\pi_i, \pi_{i+1}) \in M_3$, from (1.11) and (1.12) we have

$$\int_{\pi_i}^{\pi_{i+1}} [c\alpha(t) - \beta(t)]\, dt = x(\pi_{i+1}; c) - x(\pi_i; c) \tag{1.A.4}$$

Note that $x(\pi_i; c)$ and $x(\pi_{i+1}; c)$ can only take values from the set $\{\theta, \phi\}$. Therefore, differentiating with respect to c we obtain:

$$\left[c\alpha(\pi_{i+1}^-) - \beta(\pi_{i+1}^-)\right] \frac{\partial \pi_{i+1}}{\partial c} - \left[c\alpha(\pi_i^+) - \beta(\pi_i^+)\right] \frac{\partial \pi_i}{\partial c} + \int_{\pi_i}^{\pi_{i+1}} \alpha(t)dt = 0$$

which gives

$$\frac{\partial \pi_{i+1}}{\partial c} = \frac{c\alpha(\pi_i^+) - \beta(\pi_i^+)}{c\alpha(\pi_{i+1}^-) - \beta(\pi_{i+1}^-)} \frac{\partial \pi_i}{\partial c} - \frac{1}{c\alpha(\pi_{i+1}^-) - \beta(\pi_{i+1}^-)} \int_{\pi_i}^{\pi_{i+1}} \alpha(t)dt$$

■

Proof of Lemma 1.9. If $[\pi_i, \pi_{i+1}) \in M_1$, from (1.11) and (1.12) we obtain:

$$x(\pi_{i+1}; c) - x(\pi_i; c) = \int_{\pi_i}^{\pi_{i+1}} [\alpha(t) - \beta(t)]\, dt$$

Note that $x(\pi_i; c)$ and $x(\pi_{i+1}; c)$ can only take values from the set $\{0, \phi\}$. Therefore, differentiating with respect to c we obtain:

$$\left[\alpha(\pi_{i+1}^-) - \beta(\pi_{i+1}^-)\right] \frac{\partial \pi_{i+1}}{\partial c} - \left[\alpha(\pi_i^+) - \beta(\pi_i^+)\right] \frac{\partial \pi_i}{\partial c} = 0$$

which gives

$$\frac{\partial \pi_{i+1}}{\partial c} = \frac{\alpha(\pi_i^+) - \beta(\pi_i^+)}{\alpha(\pi_{i+1}^-) - \beta(\pi_{i+1}^-)} \cdot \frac{\partial \pi_i}{\partial c}$$

■

Proof of Lemma 1.10. Recall that $\zeta_n = \pi_{n,I_n}$ is the end of a NBP and $x(\zeta_n; c) = \theta$ or 0. Therefore $[\pi_{n,I_n-1}, \pi_{n,I_n})$, the last interval in the NBP, is either an M_1 or an M_3 interval.

Case 1: If it is an M_1 interval, according to Lemma 1.9 we obtain:

$$\frac{\partial \zeta_n}{\partial c} = \frac{B(\pi_{n,I_n-1}^+)}{B(\pi_{n,I_n}^-)} \cdot \frac{\partial \pi_{n,I_n-1}}{\partial c}$$

Case 2: If it is an M_3 interval, according to Lemma 1.8 we obtain:

$$\frac{\partial \zeta_n}{\partial c} = \frac{A(\pi_{n,I_n-1}^+)}{A(\pi_{n,I_n}^-)} \cdot \frac{\partial \pi_{n,I_n-1}}{\partial c} - \frac{x(\pi_{n,I_n}) - x(\pi_{n,I_n-1}) + H(\pi_{n,I_n-1}, \pi_{n,I_n})}{cA(\pi_{n,I_n}^-)}$$

Since $x(\zeta_n; c) = \theta$, the above equation becomes

$$\frac{\partial \zeta_n}{\partial c} = \frac{A(\pi_{n,I_n-1}^+)}{A(\pi_{n,I_n}^-)} \cdot \frac{\partial \pi_{n,I_n-1}}{\partial c} - \frac{\theta - x(\pi_{n,I_n-1}) + H(\pi_{n,I_n-1}, \pi_{n,I_n})}{cA(\pi_{n,I_n}^-)} \tag{1.A.5}$$

Moreover because $[\pi_{n,I_n-1}, \pi_{n,I_n})$ is an M_3 interval, $x(\pi_{n,I_n-1}; c) = \theta$ or ϕ. If $x(\pi_{n,I_n-1}; c) = \theta = x(\pi_{n,I_n}; c)$, the interval $[\pi_{n,I_n-1}, \pi_{n,I_n})$ forms a NBP itself and from Lemma 1.8 we have:

$$\frac{\partial \zeta_n}{\partial c} = \frac{A(\pi_{n,I_n-1}^+)}{A(\pi_{n,I_n}^-)} \cdot \frac{\partial \pi_{n,I_n-1}}{\partial c} - \frac{H(\pi_{n,I_n-1}, \pi_{n,I_n})}{cA(\pi_{n,I_n}^-)}.$$

In addition, from Lemmas 1.5 and 1.6 it follows that

$$\frac{\partial \pi_{n,I_n-1}}{\partial c} = \frac{\partial \eta_n}{\partial c} = 0.$$

We then obtain:

$$\frac{\partial \zeta_n}{\partial c} = -\frac{H(\pi_{n,I_n-1}, \pi_{n,I_n})}{cA(\pi_{n,I_n}^-)} \tag{1.A.6}$$

Similarly, if $x(\pi_{n,I_n-1}; c) = \phi$, it follows that

$$\frac{\partial \zeta_n}{\partial c} = \frac{A(\pi_{n,I_n-1}^+)}{A(\pi_{n,I_n}^-)} \cdot \frac{\partial \pi_{n,I_n-1}}{\partial c} - \frac{\theta - \phi + H(\pi_{n,I_n-1}, \pi_{n,I_n})}{cA(\pi_{n,I_n}^-)} \tag{1.A.7}$$

Combining (1.A.5), (1.A.6) and (1.A.7) we get (1.29). ■

Proof of Lemma 1.11. The proof has two steps. First we show that, if $x'(t) = x(t)$,

$$\frac{dx'(t^+)}{dt} \geq \frac{dx(t^+)}{dt} \tag{1.A.8}$$

Before proceeding we point out that according to Assumption 5, $u(t,x;c)$ is a monotonically nondecreasing function of c for any fixed t and x, i.e., $u(t,x;c) \leq u(t,x;c+\Delta c)$.

Assume $x'(t) = x(t) = x_0$. The value of x_0 can be classified as follows:

Case 1. $x_0 = 0$: In view of Assumption 5, $u(t,x_0;c) \leq u(t,x_0;c+\Delta c)$. Thus, there are three cases to consider regarding the relative values of $u(t,0;c), u(t,0;c+\Delta c)$ and $\beta(t)$.

Case 1.1. $u(t,0;c) \leq u(t,0;c+\Delta c) \leq \beta(t)$: The buffer content will remain empty for both sample paths:

$$\frac{\partial x'(t^+)}{\partial t} = \frac{\partial x(t^+)}{\partial t} = 0$$

Case 1.2. $u(t,0;c) \leq \beta(t) \leq u(t,0;c+\Delta c)$: According to (1.11),

$$\frac{\partial x'(t^+)}{\partial t} = u(t,0;c+\Delta c) - \beta(t) \geq \frac{\partial x(t^+)}{\partial t} = 0$$

Case 1.3. $\beta(t) \leq u(t,0;c) \leq u(t,0;c+\Delta c)$: According to (1.11),

$$\frac{\partial x'(t^+)}{\partial t} = u(t,0;c+\Delta c) - \beta(t) \geq u(t,0;c) - \beta(t) = \frac{\partial x(t^+)}{\partial t}$$

Case 2. $0 < x_0 < \theta$: In this case, (1.11) gives

$$\frac{\partial x(t^+)}{\partial t} = u(t,x_0;c) - \beta(t)$$

and

$$\frac{\partial x'(t^+)}{\partial t} = u(t,x_0;c+\Delta c) - \beta(t)$$

which implies

$$\frac{\partial x'(t^+)}{\partial t} \geq \frac{\partial x(t^+)}{\partial t}$$

according to Assumption 5.

Case 3. $x_0 = \theta$: There are three sub-cases to consider:

Case 3.1. $u(t,\theta;c) \leq u(t,\theta;c+\Delta c) \leq \beta(t)$: According to (1.11) and Assumption 5,

$$\frac{\partial x'(t^+)}{\partial t} = u(t,0;c+\Delta c) - \beta(t) \geq u(t,0;c) - \beta(t) = \frac{\partial x(t^+)}{\partial t}$$

Case 3.2. $u(t,\theta;c) \leq \beta(t) \leq u(t,\theta;c+\Delta c)$: According to (1.11), SFM_P will stay at θ so that

$$\frac{\partial x'(t^+)}{\partial t} = 0$$

SFM_N will drop from θ so that

$$\frac{\partial x(t^+)}{\partial t} = u(t,\theta;c) - \beta(t) < 0$$

Combining the above two equations gives

$$\frac{\partial x'(t^+)}{\partial t} \geq \frac{\partial x(t^+)}{\partial t}$$

Case 3.3. $\beta(t) \leq u(t, \theta; c) \leq u(t, \theta; c + \Delta c)$: According to (1.11), both sample paths will stay at θ, i.e.,

$$\frac{\partial x'(t^+)}{\partial t} = \frac{\partial x(t^+)}{\partial t} = 0$$

Combining all of the above cases, we conclude that

$$\frac{\partial x'(t^+)}{\partial t} \geq \frac{\partial x(t^+)}{\partial t}$$

which implies that SFM_P will never go below SFM_N for any $t > \tau$ when $x(\tau) = x'(\tau)$. Moreover when $t = 0$, $x'(0) = x(0) = 0$. It follows that $\Delta x(t) \geq 0$, for all $t \geq 0$. ■

Proof of Lemma 1.12. For $\Delta\delta(t)$, regarding all possible value combinations of $x(t)$ and $x'(t)$ we have the following cases:

Case 1. $x'(t) > x(t) > 0$: According to (1.8), $\delta'(t) = \delta(t) = \beta(t)$;

Case 2. $x'(t) > 0$, $x(t) = 0$: According to (1.8), $\delta(t) = \min(u(t), \beta(t)) \leq \beta(t) = \delta'(t)$;

Case 3. $x'(t) = 0$, $x(t) > 0$: According to Lemma 1.11, this case is impossible;

Case 4. $x'(t) = 0$, $x(t) = 0$: According to (1.8), $\delta(t) = \min(u(t), \beta(t))$, $\delta'(t) = \min(u'(t), \beta(t))$. Moreover Assumption 5 gives $u(t) = u(t, 0; c) \leq u(t, 0; c + \Delta c) = u'(t)$, so $\delta(t) \leq \delta'(t)$.

Combining the above four cases gives $\Delta\delta(t) \geq 0$.

Similarly for $\Delta\gamma(t)$, regarding all the possible value combinations of $x(t)$ and $x'(t)$ we have the following cases:

Case 1. $x(t) \leq x'(t) < \theta$: According to (1.9), $\gamma(t) = \gamma'(t) = 0$;

Case 2. $x(t) < x'(t) = \theta$: According to (1.9), $\gamma(t) = 0 \leq \gamma'(t)$;

Case 3. $x(t) = x'(t) = \theta$: According to (1.9), $\gamma(t) = \max(u(t) - \beta(t), 0)$ and $\gamma'(t) = \max(u'(c + \Delta c; t, \theta) - \beta(t), 0)$. Moreover, Assumption 5 gives $u(t) = u(t, 0; c) \leq u'(t) = u(t, 0; c + \Delta c)$, so $\gamma(t) \leq \gamma'(t)$.

Case 4. $x(t) = \theta, x'(t) < \theta$: This case is impossible according to Lemma 1.11.

Combining the above cases gives $\Delta\gamma(t) \geq 0$. ■

Proof of Theorem 1.13. We have the following flow balance equations for SFM_N and SFM_P respectively:

$$x(t) - x(0) = \int_0^t u'(\tau)d\tau - \int_0^t \delta(\tau)d\tau - \int_0^t \gamma(\tau)d\tau, \text{ for all } t \geq 0$$

and

$$x'(t) - x'(0) = \int_0^t u'(\tau)d\tau - \int_0^t \delta'(\tau)d\tau - \int_0^t \gamma'(\tau)d\tau, \text{ for all } t \geq 0$$

Combining the above equations and recalling that $x'(0) = x(0) = 0$, we obtain:

$$\Delta x(t) = \int_0^t \Delta u(\tau)d\tau - \int_0^t \Delta\delta(\tau)d\tau - \int_0^t \Delta\gamma(\tau)d\tau \tag{1.A.9}$$

Recalling Lemma 1.12, $\Delta\delta(\tau) \geq 0$ and $\Delta\gamma(\tau) \geq 0$ for all τ, $0 < \tau < t$. Then, (1.A.9) implies

$$\Delta x(t) \leq \int_0^t \Delta u(\tau)d\tau \tag{1.A.10}$$

On the other hand,

$$\Delta u(\tau) = u(\tau, x'(\tau); c + \Delta c) - u(\tau, x(\tau); c) \qquad (1.A.11)$$

Lemma 1.11 gives $x'(\tau) \geq x(\tau)$, from which we obtain

$$u(\tau, x'(\tau); c + \Delta c) \leq u(\tau, x(\tau); c + \Delta c)$$

according to Assumption 4. Combining the above inequality with (1.A.11) gives

$$\Delta u(\tau) \leq u(\tau, x(\tau); c + \Delta c) - u(\tau, x(\tau); c) \qquad (1.A.12)$$

According to Assumption 6,

$$u(\tau, x(\tau); c + \Delta c) - u(\tau, x(\tau); c) \leq K\Delta c \qquad (1.A.13)$$

Combining (1.A.12) and (1.A.13) we get:

$$\Delta u(\tau) \leq K\Delta c \qquad (1.A.14)$$

Thus, from (1.A.10) and (1.A.14) we obtain:

$$\Delta x(t) \leq \int_0^t \Delta u(\tau) d\tau \leq Kt\Delta c \leq KT\Delta c \qquad \text{for all } t, \quad 0 \leq t \leq T$$

■

References

L. Kleinrock, *Queueing Systems*, vol. I: Theory. Wile-Interscience, 1975.

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.

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.

D. Anick, D. Mitra, and M. Sondhi, “Stochastic theory of a data-handling system with multiple sources,” *The Bell System Technical Journal*, vol. Vol. 61, pp. 1871–1894, 1982.

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.

R. Cruz, “A calculus for network delay, Part I: Network elements in isolation,” *IEEE Transactions on Information Theory*, 1991.

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.

K. Kumaran and D. Mitra, “Performance and fluid simulations of a novel shared buffer management system,” in *Proceedings of IEEE INFOCOM*, March 1998.

B. Liu, Y. Guo, J. Kurose, D. Towsley, and W. Gong, "Fluid simulation of large scale networks: Issues and tradeoffs," in *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications*, June 1999. Las Vegas, Nevada.

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.

R. Akella and P.R.Kumar, "Optimal control of production rate in a failure prone manufacturing system," *IEEE Trans. Automatic Control*, vol. AC-31, pp. 116–126, Feb. 1986.

J. Perkins and R. Srikant, "The role of queue length information in congestion control and resource pricing," *Proc. of the 38th Conference on Decision and Control*, Dec. 1999.

J. Perkins and R. Srikant, "Failure-prone production systems with uncertain demand," *IEEE Trans. Automatic Control*, vol. 46, pp. 441–449, 2001.

F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: Shadow price, proportional fairness and stability," *Journal of the Operational Research Society*, 1998.

S. Low, "A duality model of tcp and queue management algorithms," in *Proceedings of ITC Specialist Seminar on IP Traffic Measurement, Modeling and Management*, 2000.

Y. Ho and X. Cao, *Perturbation Analysis of Discrete Event Dynamic Systems*. Boston, Massachusetts: Kluwer Academic Publishers, 1991.

C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Kluwer Academic Publishers, 1999.

Y. Wardi, B. Melamed, C. G. Cassandras, and C. G. Panayiotou, "IPA gradient estimators in single-node stochastic fluid models," *Journal of Optimization Theory and Applications*, vol. 115, no. 2, pp. 369–406, 2002.

Y. Liu and W. Gong, "Perturbation analysis for stochastic fluid queueing systems," in *Proc. 38th IEEE Conf. Dec. and Ctrl*, pp. 4440–4445, 1999.

C. G. Cassandras, Y. Wardi, B. Melamed, G. Sun, and C. G. Panayiotou, "Perturbation analysis for online control and optimization of stochastic fluid models," *IEEE Trans. on Automatic Control*, vol. 47, no. 8, pp. 1234-1248, 2002.

C. G. Cassandras, G. Sun, C. G. Panayiotou, and Y. Wardi, "Perturbation Analysis and Control of Two-Class Stochastic Fluid Models for Communication Networks", *IEEE Trans. on Automatic Control*, vol. 48, no. 5, pp. 770-782, 2003.

S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, pp. 397–413, 1993.

V. Jacobson, "Congestion avoidance and control," *Computer Communication Review*, vol. 18, pp. 314–329, August 1988.

L.Takacs, "A single-server queue with feedback," *Bell System Tech. J.*, vol. 42, pp. 505–519, 1963.

R.D.Foley and R.L.Disney, "Queues with delayed feedback," *Adv. In Appl. Probab.*, vol. 15, no. 1, pp. 162–182, 1983.

G. R. D. Avignon and R.L.Disney, "Queues with instantaneous feedback," *Management Sci.*, vol. 24, pp. 168–180, 1977/78.

M.A.Wortman, R.L.Disney, and P. Kiessler, "The M/GI/1 bernoulli feedback queue with vacations," *Queueing Systems Theory Appl.*, vol. 9, no. 4, pp. 353–363, 1991.

G. Sun, C. G. Cassandras, Y. Wardi, and C. G. Panayiotou, "Perturbation analysis of stochastic flow networks," in *Proceedings of 42nd IEEE Conf. On Decision and Control*, pp. 4831–4836, Dec. 2003.

H. Yu and C. G. Cassandras, "Perturbation analysis of feedback-controlled stochastic flow systems," in *Proceedings of 42nd IEEE Conf. On Decision and Control*, pp. 6277-6282, Dec. 2003.

P. Glasserman, *Gradient Estimation via Perturbation Analysis.* Kluwer Academic Pub., 1991.

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.

H. Kushner and G. Yin, *Stochastic Approximation Algorithms and Applications.* New York, NY: Springer-Verlag, 1997.

H. Yu and C. G. Cassandras, "Perturbation analysis for production control and optimization of manufacturing systems," *Automatica*, vol. 40, pp. 945-956, 2004.