

Infinitesimal Perturbation Analysis for a Single Stochastic Fluid Model Node with a Class of Feedback Controlled Traffic

Christos G. Panayiotou

Abstract—In this paper we adopt the Stochastic Fluid Modeling framework for management and control of communication networks and attempt to explicitly model feedback controlled sources. Specifically, the inflow process consists of a feedback controlled source which is modelled as a hybrid automaton (it has both, time-driven as well as event-driven dynamics). This paper derives Infinitesimal Perturbation Analysis (IPA) derivative estimators for the buffer occupancy and throughput with respect to a node parameter (i.e., the buffer size). As also shown in earlier work [1], [2], [3], such estimators are used together with stochastic approximation techniques to dynamically determine the optimal operating point.

I. INTRODUCTION

Managing and operating large scale communication networks is a challenging task and it is only expected to get worse as networks grow larger and larger. The managing difficulties stem from the fact that modeling and analysis of large scale communication networks is an excessively difficult task. The natural modeling framework for packet-based communication networks is provided through queueing systems, however, the huge traffic volume that networks are supporting today makes such models highly impractical. If, on the other hand, we are to resort to analytical techniques from classical queueing theory, we find that traditional traffic models, largely based on Poisson processes, need to be replaced by more sophisticated stochastic processes that capture the bursty nature of realistic traffic. At the same time we need to account for the fact that the stochastic processes involved are time-varying, i.e., no stationarity assumption and, in addition, we need to explicitly model buffer overflow phenomena which typically defy tractable analytical derivations.

An alternative to queueing models is a hybrid modeling paradigm that is based on Stochastic Fluid Models (SFM) which have recently been shown to be especially useful for analyzing various kinds of high-speed networks (see [1], [4], [5], [6], [7] and references therein). Using this modeling framework a new approach for network management is being developed which is based on Infinitesimal Perturbation Analysis (IPA) [1], [2], [8]. In this approach, we estimate the gradient of the performance measure of interest (e.g., packet delay) 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. (i) The gradient estimation is done *on-line* thus the approach can be implemented on the real system; As the operating conditions change, it *continuously* seeks to optimize a generally time-varying performance metric. (ii) The gradient estimation process does not require any knowledge of the system's underlying stochastic processes. (iii) It turns out that the estimators consist of accumulators and timers and are generally easy to implement.

To date, the use of IPA for the control of a single class traffic in a single-node SFM has been studied in [1] where the buffer capacity is adjusted to optimize a performance measure of interest. In [2], [3] the approach is extended to include multiple classes of traffic with multiple thresholds while in [9] the approach is extended to a tandem network with a single traffic class. A common assumption in the models described above [1], [2], [3], [9] is that the packet arrival and service processes are independent of the control parameters, i.e., the various thresholds. In the context of the current computer networks, this assumption is valid for UDP (User Datagram Protocol) traffic however it is invalid for TCP (Transmission Control Protocol) traffic. In this paper, we extend the framework and study a class of feedback controlled sources of traffic. As explained in the next section, this class of sources is modelled as hybrid automaton [10] which falls in the hybrid TCP modeling framework developed in [11]. The first contribution of this paper is the derivation of IPA sensitivity estimates of performance measures such as average buffer content and throughput with respect to the buffer thresholds. Feedback controlled traffic is also studied in [12] however, the problem studied in [12] is in the context of manufacturing systems and the source model used is different. Other efforts to include different forms of feedback controlled traffic include [13]. Another contribution of this paper is that the proposed model explicitly addresses the propagation delay in providing feedback to the source. Such propagation delay is typical in current computer communication networks.

II. SYSTEM MODEL

The SFM setting is based on the fluid-flow world-view where “liquid molecules” flow in a continuous fashion. The basic SFM, used in [1] consists of a single-server

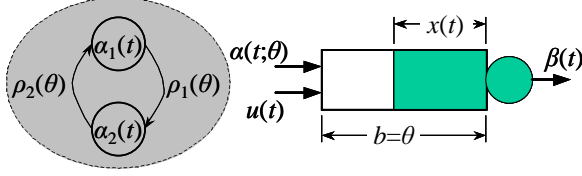


Fig. 1. System model

preceded by a fluid storage buffer with finite capacity b . The system is characterized by a set of stochastic processes, all defined on a common probability space (Ω, \mathcal{F}, P) . In [1] a source produces fluid at rate $\alpha(t)$ while the server processes the fluid at rate $\beta(t)$. Both, $\alpha(t)$ and $\beta(t)$ are time varying random functions independent of the buffer capacity. Given $\alpha(t)$, $\beta(t)$ and the buffer capacity b we derive three stochastic processes as follows:

- $\{\delta(t)\}$ the actual fluid discharge rate from the server,
- $\{x(t)\}$ the buffer occupancy or buffer content, i.e., the amount of fluid in the buffer,
- $\{\gamma(t)\}$ the overflow rate due to a full buffer.

As already mentioned, in [1] it is assumed that $\alpha(t)$ is *independent* of the buffer capacity $b = \theta$ which will constitute the parameter we will concentrate on for the purposes of IPA. In this paper we relax this assumption and allow the source's inflow rate to depend on the parameter θ as shown in Fig. 1. Specifically, we assume that the source can be in either of two states and in each state it can transmit at rates $\alpha_1(t)$ or $\alpha_2(t)$ respectively. Both, $\alpha_1(t)$ and $\alpha_2(t)$ are time varying random functions locally independent of θ . On the other hand, the state transitions depend on θ and as a result the resulting inflow process $\alpha(t; \theta)$ is a function of θ . The specific dependence on θ will be considered in a following subsection. Furthermore, as shown in Fig. 1 we assume a second uncontrolled traffic source with an inflow rate $u(t)$ which, in the context of communication networks may correspond to traffic like UDP; $u(t) < \infty$ is also a random time-varying function. The above processes evolve over a given time interval $[0, T]$ for a given fixed $0 < T < \infty$.

A. System Dynamics

To simplify the notation, we define the *net inflow* rate

$$A(t; \theta) = \alpha(t; \theta) + u(t) - \beta(t) \quad (1)$$

and similarly define

$$A_1(t) = \alpha_1(t) + u(t) - \beta(t) \text{ and } A_2(t) = \alpha_2(t) + u(t) - \beta(t).$$

The buffer content is defined by the following one-sided differential equation,

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

whose initial condition will be set to $x(0; \theta) = x_0$; for simplicity, we set $x_0 = 0$ throughout the paper. The outflow

rate $\delta(t; \theta)$ is defined by

$$\delta(t; \theta) = \begin{cases} \beta(t), & \text{if } x(t; \theta) > 0, \\ \alpha(t; \theta) + u(t), & \text{if } x(t; \theta) = 0, \end{cases} \quad (2)$$

and the overflow rate $\gamma(t; \theta)$ is given by,

$$\gamma(t; \theta) = \begin{cases} A(t; \theta), & \text{if } x(t; \theta) = b, \\ 0, & \text{if } x(t; \theta) < b. \end{cases} \quad (3)$$

For completeness we assume that losses of the two types of traffic $\alpha(t; \theta)$ and $u(t)$ are proportional to the inflow rates of each traffic. This SFM can be viewed as a dynamical system whose input consists of the three processes $\{\alpha(t; \theta)\}$, $\{u(t)\}$, and $\{\beta(t)\}$, its state is comprised of the buffer content process, and its output includes the outflow and overflow processes. This system is also viewed as a Discrete Event System where we identify the following events.

- e_1 : buffer *ceases to be empty*,
- e_2 : buffer *becomes full*,
- e_3 : buffer *ceases to be full*,
- e_4 : buffer *becomes empty*,
- e_5 : sign of $A(t; \theta)$ changes from positive to negative,
- e_6 : sign of $A(t; \theta)$ changes from negative to positive.

Some events are shown in Fig. 2 which shows a typical sample path of the system; for example v_0 corresponds to an e_1 , v_1 to an e_2 , v_2 to an e_3 and v_S to an e_4 .

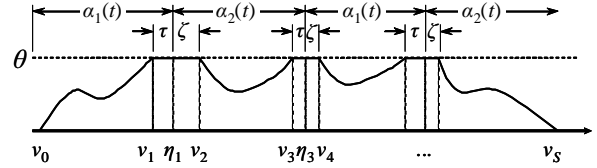


Fig. 2. Typical buffering period of the system sample path

B. State Transition Mechanism of the Hybrid Automaton

In this section we examine the state transition mechanism of the source process $\alpha(t, \theta)$. We assume that initially it starts transmitting according to $\alpha_1(t)$ until the buffer becomes full (say at time v_1 as in Fig. 2). At this point the node will send a signal to the source to change its transmission rate. The signal will arrive at the source after a generally random delay τ and will cause a transition from state 1 to state 2 (point η_1 in Fig. 2). Thus the new transmission rate will become $\alpha_2(t)$. At this point, assuming the source was effectively controlled, one would have expected the buffer content $x(t)$ to cease to be full. However, as indicated in Fig. 2, this does not happen until ζ time units later, where ζ is another random variable. This delay is used to model the propagation delay from the source to the buffer. In addition, ζ is needed because it is possible that at the time of the state transition, the net inflow to the buffer might still be greater than the server capacity (i.e., $A(t; \theta) > \beta(t)$) due to the uncontrolled flow $u(t)$. The source will continue transmitting at a rate $\alpha_2(t)$ until the next time the buffer will become full. At this point,

again the node will send a signal to the source to change its transmission rate. The signal will arrive at the source after a random delay τ and will cause a transition from state 2 to state 1 and so on.

Remark 1: The function $\alpha_i(t)$ $i = 1, 2$ can be any general, random, and time varying function, as long as it is finite. For effective congestion control however, one might expect that at the transition time, $\alpha_i(t)$ starts small and is an increasing functions of time t . This is consistent with TCP's slow start and congestion avoidance states.

For the purposes of this paper, the random delay τ will be considered as linearly dependent on θ , i.e.,

$$\tau = c\theta + \tilde{d} \quad (4)$$

where $c \geq 0$ constant, and \tilde{d} is a random variable independent of θ . Note that this model allows for $c = 0$ which makes τ independent of θ . Furthermore, the random delay ζ will be assumed independent of θ .

Remark 2: In the context of communication networks the propagation delay may or may not depend on the buffer capacity θ , depending on the protocol. For example, in protocols with backward explicit congestion notification, the delay is independent of θ . On the other hand, in general TCP traffic or forward explicit congestion notification (e.g., [14]), the source detects congestion after a round trip time. This is proportional to θ because the packets that follow the dropped packet (i.e., the ones that signal congestion to the source) have to go through a full buffer.

Other approaches that apply IPA on feedback controlled traffic include [12], [13] however they either use different forms of feedback or do not account for the delayed reaction of the feedback controlled source. The IPA approach is described in the next section but first we define the performance measures of interest that one might want to optimize through gradient based techniques and thus use the IPA estimates.

C. Performance Measures

In this paper we are concerned with two sample performance measures of interest, the average queue length $Q(T; \theta)$ and, the throughput $W(T; \theta)$, both defined over a finite interval $[0, T]$ as shown below.

$$Q(T; \theta) = \frac{1}{T} \int_0^T x(t; \theta) dt \quad (5)$$

$$W(T; \theta) = \frac{1}{T} \int_0^T \delta(t; \theta) dt \quad (6)$$

We point out that from $Q(T; \theta)$ it is possible to get delay related measures using Little's Law (see [15]). Furthermore, it is worth pointing out that unlike the problem formulation in earlier work [1], [2] (with uncontrolled source models), the problem considered in this paper does not include the loss volume metric. Rather, in this paper we have substituted the loss volume with the throughput metric. The justification for this change is based on the fact that

in feedback controlled sources (such as TCP sources) lost packets are retransmitted. Furthermore, it is unclear whether it is effective (or desirable) to control packet losses from feedback controlled sources by changing the buffer size¹.

Having no buffer, implies minimum packet delay however, this seriously restricts the system's throughput. To trade off between the two measures we define an objective function which is a linear combination of the two. Of course one is interested in minimizing the average delay while maximizing the expected throughput, thus we define the following objective function

$$\min J(T; \theta) \equiv \mathbb{E}[Q(T; \theta)] - h\mathbb{E}[W(T; \theta)] \quad (7)$$

where $h \geq 0$ is an appropriate scaling constant.

III. INFINITESIMAL PERTURBATION ANALYSIS (IPA)

As in earlier work, we divide the system sample path in buffering periods (BP) and Empty Periods (EP) (a typical BP is shown in Fig. 2). Thus the sample functions of (5) and (6) can be rewritten as

$$Q(T; \theta) = \frac{1}{T} \sum_{k=1}^{N_T} q_k = \frac{1}{T} \sum_{k=1}^{N_T} \int_{v_{k,0}}^{v_{k,S_k}} x(t; \theta) dt \quad (8)$$

$$W(T; \theta) = \frac{1}{T} \sum_{k=1}^{N_T} w_k = \frac{1}{T} \sum_{k=1}^{N_T} \int_{v_{k-1,S_{k-1}}}^{v_{k,S_k}} \delta(t; \theta) dt \quad (9)$$

where N_T is the random number of BPs in the interval $[0, T]$, $v_{k,0}$ denotes the beginning of the the k th BP while v_{k,S_k} denotes its end time. For the analysis that follows we are only considering a single BP and in order to simplify the notation we drop the index k . We also use the notation v_{-1} to indicate the ending time of the *previous* BP. Fig. 2 shows a typical buffering period which is divided into intervals $p_i = [v_i, v_{i+1})$, $i = 0, \dots, S-1$. Before we proceed with the derivation of the IPA derivatives we make the following mild assumption that guarantees the existence of derivatives.

Assumption 1:

- $\alpha_1(t) < \infty$, $\alpha_2(t) < \infty$, $u(t) < \infty$, and $\beta(t) < \infty$ for all $t \in [0, T]$
- For every $\theta \in \Theta$, w.p. 1, no two events may occur at the same time
- W.p. 1, there exists no interval $(v_i, v_i + \tau)$, $\tau > 0$, such that $x(t) = \theta$ for all $t \in (v_i(\theta), v_i(\theta) + \tau)$, and $A_1(t; \theta) = 0$ or $A_2(t; \theta) = 0$.

All three parts of *Assumption 1* are mild technical conditions. Regarding part *c*, we point out that if it is violated, one-sided derivatives may still be used, however we make the assumption to simplify our analysis.

In addition, to simplify the notation we define the processes $G_i(t)$ and $\bar{G}_i(t)$, where, $G_1(t) = A_1(t)$ if the previous BP ended when the source was in state 1, and $G_1(t) = A_2(t)$ if the previous BP ended when the source

¹Ignoring any background traffic, every time the buffer becomes full, it stays full for $\tau + \zeta$ time units. Thus packet losses are proportional to the length of that interval which is an increasing function of θ . In other words, increasing the buffer size increases packet losses.

was in state 2. $\bar{G}_1(t) = A_1(t)$ if $G_1(t) = A_2(t)$ and $\bar{G}_1(t) = A_2(t)$ if $G_1(t) = A_1(t)$. Subsequently, for all $i = 2, \dots, \frac{S+1}{2}$ the $G_i(t)$ and $\bar{G}_i(t)$ will be determined recursively

$$G_i(t) = \bar{G}_{i-1}(t) \quad \text{and} \quad \bar{G}_i(t) = G_{i-1}(t) \quad (10)$$

Clearly, $G_i(t)$ and $\bar{G}_i(t)$, $i = 1, \dots, \frac{S+1}{2}$ are all independent of θ . Also, for notational economy we define

$$\Delta G_i(t) = G_i(t) - \bar{G}_i(t) \quad (11)$$

A. IPA for the Buffer Content

Over a single buffering period, the buffering content is given by

$$q = \int_{v_0}^{v_1} x(t) dt + \sum_{i=1}^{\frac{S-1}{2}} \left[(v_{2i} - v_{2i-1})\theta + \int_{v_{2i}}^{v_{2i+1}} x(t; \theta) dt \right] \quad (12)$$

where we point out that $x(t)$ over the interval $[v_0, v_1]$ is independent of the buffer size θ . Next we take derivatives one term at a time, so we have

$$\frac{d}{d\theta} \int_{v_0}^{v_1} x(t) dt = x(v_1)v'_1 = \theta v'_1 \quad (13)$$

where the prime notation (\cdot') denotes the derivative with respect to θ . To derive (13) we have used the fact that v_0 and $x(t)$ during this interval are locally independent from θ , and, $x(v_1) = \theta$.

$$\begin{aligned} \frac{d}{d\theta} \sum_{i=1}^{\frac{S-1}{2}} [(v_{2i} - v_{2i-1})\theta] \\ = \sum_{i=1}^{\frac{S-1}{2}} [(v_{2i} - v_{2i-1}) + (v'_{2i} - v'_{2i-1})\theta] \end{aligned} \quad (14)$$

$$\begin{aligned} \frac{d}{d\theta} \sum_{i=1}^{\frac{S-3}{2}} \left[\int_{v_{2i}}^{v_{2i+1}} x(t; \theta) dt \right] \\ = \sum_{i=1}^{\frac{S-3}{2}} \left[(v'_{2i+1} - v'_{2i})\theta + \int_{v_{2i}}^{v_{2i+1}} x'(t; \theta) dt \right] \end{aligned} \quad (15)$$

and for the last non full interval,

$$\frac{d}{d\theta} \int_{v_{S-1}}^{v_S} x(t; \theta) dt = -v'_{S-1}\theta + \int_{v_{S-1}}^{v_S} x'(t; \theta) dt \quad (16)$$

Combining equations (13) - (16) we observe that all terms $v'_i\theta$ cancel out. Therefore

$$\frac{dq}{d\theta} = \sum_{i=1}^{\frac{S-1}{2}} \left[(v_{2i} - v_{2i-1}) + \int_{v_{2i}}^{v_{2i+1}} x'(t; \theta) dt \right] \quad (17)$$

To determine the $x'(t; \theta)$ term that appears in (17) we recognize that in any interval (v_{2i}, v_{2i+1}) $i = 1, \dots, \frac{S-1}{2}$

$$x(t; \theta) = \theta + \int_{v_{2i}}^t G_{i+1}(r) dr$$

Thus, $x'(t; \theta)$ is given by

$$x'(t; \theta) = 1 - G_{i+1}(v_{2i})v'_{2i}$$

and therefore

$$\int_{v_{2i}}^{v_{2i+1}} x'(t; \theta) dt = [1 - G_{i+1}(v_{2i})v'_{2i}] (v_{2i+1} - v_{2i}) \quad (18)$$

Combining (18) with (17) we get

$$\frac{dq}{d\theta} = (v_S - v_1) - \sum_{i=1}^{\frac{S-1}{2}} G_{i+1}(v_{2i}) (v_{2i+1} - v_{2i}) v'_{2i}. \quad (19)$$

Equation (19) shows that the buffer content derivative consists of a timer that counts the interval between the first overflow and the end of the buffering period $(v_S - v_1)$. This is precisely the result obtained for the uncontrolled sources (see [1]). Subsequently, there is also a ‘‘correction’’ term which accounts for the feedback. Its contribution can be evaluated given the rate $A_1(t)$ and $A_2(t)$ and the time derivatives v'_i $i = 1, \dots, S$. These rates can be estimated or can be evaluated using analytical techniques (see for example [11]). Furthermore, the time instances v_i , $i = 1, \dots, S$ are generally easy to observe from the queue sample path at the switch; these correspond to instances where packets are lost or instances where the buffer ceases to be full. Time instances η_{2i-1} , $i = 1, \dots, \frac{S-1}{2}$ are a little trickier to obtain since they are not directly observable. However, depending on the specific protocol used, it may be possible to infer them. In the worse case, the node can probe the network and get an estimate of τ . We don't expect that accurate estimate of τ is critical to the performance of the estimator, but we will investigate this further in the future. Thus if the switch has estimates of $A_1(t)$, $A_2(t)$ and an estimate of τ , the only thing required for evaluating (19) are the time derivatives v'_i and η'_i . These are evaluated recursively as we show in Section III-C.

B. IPA for Throughput

As seen in equations (2) and (9) for the evaluation of throughput, we also need to consider the empty period that precedes the busy period. During this period the fluid discharge of the server is equal to the fluid inflow, i.e., $G_1(t) + \beta(t)^2$. During the BP, the fluid discharge is $\beta(t)$, thus during a cycle of an empty period followed by a busy period, the throughput is given by

$$w = \int_{v_{-1}}^{v_0} (G_1(t) + \beta(t)) dt + \int_{v_0}^{v_S} \beta(t) dt$$

where, as mentioned earlier, we denote the end point of the previous BP (i.e., $v_{k-1, S_{k-1}}$) by v_{-1} . Differentiating we get

$$\frac{dw}{d\theta} = -G_1(v_{-1})v'_{-1} - \beta(v_{-1})v'_{-1} + \beta(v_S)v'_S \quad (20)$$

²Note that $G_i(t)$ corresponds to the *net* inflow.

Substituting these terms in (8) then all $\beta(v_i)v'_i$ terms cancel out. So the throughput sample derivative is given by the sum of terms of the form

$$\frac{dw}{d\theta} = -G_1(v_{-1})v'_{-1} \quad (21)$$

The throughput derivative is simply a collection of terms which are the product between the net inflow at the instant when the buffer empties times the derivative of that time instant. Next, we derive an iterative algorithm for evaluating these event time derivatives.

C. Time Derivatives

In this section we determine the event time derivatives v'_i and η'_i , $i = 1, 2, \dots$. First, note that

$$\int_{v_0}^{v_1} G_1(t)dt = \theta.$$

Differentiating both sides we get

$$G_1(v_1)v'_1 = 1 \Rightarrow v'_1 = \frac{1}{G_1(v_1)} \quad (22)$$

where we have used the fact that v_0 and $G_1(t)$ are independent of θ . Next, we investigate how time perturbations propagate during any period “non-full” period $[v_{2i}, v_{2i+1})$, $i = 1, \dots, \frac{S-3}{2}$. For all such periods (except of the last one) $x(v_{2i}) = \theta$, $x(v_{2i+1}) = \theta$, and $x(t) < \theta$ for all $v_{2i} < t < v_{2i+1}$, therefore

$$\int_{v_{2i}}^{v_{2i+1}} G_{i+1}(t)dt = 0.$$

Thus differentiating with respect to θ we get

$$G_{i+1}(v_{2i+1})v'_{2i+1} - G_{i+1}(v_{2i})v'_{2i} = 0$$

Rearranging terms we get

$$v'_{2i+1} = \frac{G_{i+1}(v_{2i})}{G_{i+1}(v_{2i+1})}v'_{2i} \quad (23)$$

$i = 1, \dots, \frac{S-3}{2}$. Finally,

$$\int_{v_{S-1}}^{v_S} G_{\frac{S+1}{2}}(t)dt = -\theta.$$

Thus, after differentiating with respect to θ and rearranging terms we get

$$v'_S = -\frac{1 - G_{\frac{S+1}{2}}(v_{S-1})v'_{S-1}}{G_{\frac{S+1}{2}}(v_S)} \quad (24)$$

Finally, we investigate how perturbations propagate from the beginning of a full period to the end of the full period, i.e., from v_{2i-1} to v_{2i} , $i = 1, \dots, \frac{S-1}{2}$. For this, we identify two cases.

- C1: The time of the occurrence of the e_3 event (i.e., v_{2i} , $i = 1, \dots, \frac{S-1}{2}$) is locally independent of θ^3 .

³Note that this would correspond to a case where the e_3 event ($A(t; \theta)$ turns from positive to negative) is caused by a change of the instantaneous rate of the uncontrolled traffic $u(t)$.

In this case,

$$v'_{2i} = 0 \text{ for all } i = 1, \dots, \frac{S-1}{2}. \quad (25)$$

- C2: The event e_3 depends on θ . For the purposes of this paper we will consider the following linear dependence⁴:

$$\begin{aligned} v_{2i} &= v_{2i-1} + \tau + \zeta \\ &= v_{2i-1} + c\theta + \tilde{\zeta} \end{aligned} \quad (26)$$

where $\tilde{\zeta} = \tilde{d} + \zeta$ is a random variable independent of θ . Differentiating, we get

$$v'_{2i} = v'_{2i-1} + c \quad (27)$$

Equations (22)-(27) describe recursive algorithms for determining the time derivatives v'_i , $i = 1, \dots, S$. Furthermore, the derivative of η_{2i-1} , $i = 1, \dots, \frac{S-1}{2}$ is given by

$$\eta'_{2i-1} = v'_{2i} \quad (28)$$

This follows immediately since (as mentioned earlier) $\eta_{2i-1} + \zeta = v_{2i}$, $i = 1, \dots, \frac{S-1}{2}$ and ζ is independent of θ . For the two cases described above, we can also summarize the algorithms.

- C1: All occurrences of e_3 events are independent of θ , then

$$v'_i = \begin{cases} \frac{1}{G_1(v_1)} & \text{if } i = 1 \\ 0 & \text{for } i = 2, \dots, S-1 \\ \frac{-1}{G_{\frac{S+1}{2}}(v_S)} & \text{if } i = S. \end{cases} \quad (29)$$

Note that $v'_S > 0$ since $G_{\frac{S+1}{2}}(v_S) < 0$ because at v_S the buffer empties. This is precisely the result obtained in [1].

- C2: All occurrences of e_3 event have a linear dependence with respect to θ and the previous e_2 event.

$$v'_i = \begin{cases} \frac{1}{G_1(v_1)} & \text{if } i = 1, \\ v'_{i-1} + c & \text{if } i \text{ is even,} \\ \frac{G_{\frac{i-1}{2}}(v_{i-1})}{G_{\frac{i-1}{2}}(v_i)}v'_{i-1} & \text{if } i > 1 \text{ is odd} \end{cases} \quad (30)$$

In a simulation environment, it is generally easy to identify which of the two cases above applies and thus use the right formula. For online implementations however, this might not be so easy, since one would have to infer which case applies. One approach would be to just assume that only one of the cases always applies. For examples, in problems with heavy uncontrolled UDP traffic, the moment that the event *buffer cease to be full* occurs is likely to be independent of the buffer size θ and thus for such instances one can use only case 1. Another approach is to just randomize (according to observed probability mass function). In any case the exact implementation of this for online applications is still under investigation.

⁴Non-linear dependence is also possible but it is out of the scope of this paper.

IV. SIMULATION RESULTS

In this section we present some simulation results that indicate the benefit of this approach. The model used consists of a controlled source with two states (see Fig. 1). Once the source switches to any of the two states, it starts transmitting according to a Poisson process with rate 1 packet per second. Every 0.2 seconds its transmission rate rate is increased by 1, i.e.,

$$\alpha_1(t - t_k) = \alpha_2(t - t_k) = 1 + 5(t - t_k)$$

packets per second, where t_k indicates the time of the k th transition from one state to the other. All packets are assumed to have fixed size equal to 1K bytes. Furthermore, we assume that the transmission rate of the link is constant at 0.5M bytes per second and that there is no background traffic, i.e., $u(t) = 0$ for all t . Finally, we assume the $\tau = 10$ ms and $\zeta = 0$, both deterministic.

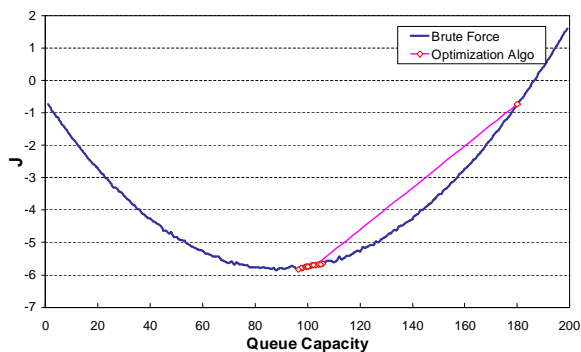


Fig. 3. Simulation results

The “Brute Force” curve in Figure 3 shows an estimate of the cost function of (7) with $h = 0.2$. These estimates are obtained by running long simulations for $\theta = 1, \dots, 200$ (10 hours for each sample point). Subsequently, we use the algorithm

$$\theta_{k+1} = \theta_k - \sigma_k \hat{J}'_k$$

to obtain the optimal buffer threshold. Where σ_k , $k = 1, 2, \dots$ is the step size sequence but for simplicity we used $\sigma_k = 1$ for all k . \hat{J}'_k is the sample derivative of the cost function obtained through IPA. Each sample derivative was obtained after observing a sample path of length $T = 1000$ seconds, however, we point out that significantly shorter observation intervals can also be used at the expense of some oscillations around the optimal point. As indicated in the simulation results, within a *single* iteration, the controller was able to bring the system very close to the optimality region.

V. CONCLUSIONS AND FUTURE WORK

In this paper we propose a stochastic fluid model with feedback controlled traffic and derive IPA derivative estimates of metrics like buffer content and throughput with respect to the buffer threshold. This approach is intended

to extend the control framework proposed in [1] (i.e., use SFM-based IPA together with gradient-based stochastic approximation techniques for on-line control of queueing networks) to networks with TCP traffic. A number of issues are still open and are investigated. The first issue is the unbiasedness of the derived estimators. This can be done by showing that the performance measures of interest are Lipschitz continuous functions of the parameter θ . This task has been proven more difficult than earlier work because of the introduction of the round-trip time delay⁵. The second issue deals with finding the most appropriate TCP hybrid model for the various versions of TCP and verify it. Finally, some implementation issues need to be resolved (e.g., figure out ways of determining the inflow rates).

REFERENCES

- [1] 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.
- [2] 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, no. 5, pp. 770–782, May 2003.
- [3] G. Sun, C. Cassandras, and C. Panayiotou, “Perturbation analysis of multiclass stochastic fluid models,” *Journal of Discrete Event Dynamic Systems*, 2004, accepted.
- [4] G. Kesidis, A. Singh, D. Cheung, and W. Kwok, “Feasibility of fluid-driven simulation for ATM network,” in *Proc. IEEE Globecom*, vol. 3, 1996, pp. 2013–2017.
- [5] K. Kumaran and D. Mitra, “Performance and fluid simulations of a novel shared buffer management system,” in *Proceedings of IEEE INFOCOM*, March 1998.
- [6] 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.
- [7] 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.
- [8] 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.
- [9] G. Sun, C. Cassandras, Y. Wardi, and C. Panayiotou, “Perturbation analysis of stochastic flow networks,” in *Proceedings of IEEE Conference on Decision and Control*, Dec 2003, pp. 4831–4838.
- [10] T. A. Henzinger, “The theory of hybrid automata,” in *Proceedings of the IEEE Symposium on Logic in Computer Science, LICS '96*, July 1996, pp. 278–292.
- [11] S. Bohacek, J. Hespanha, J. Lee, and K. Obraczka, “Analysis of a tcp hybrid model,” in *Proc. of 39th Annual Allerton Conference on Communication, Control, and Computing*, Oct 2001.
- [12] H. Yu and C. G. Cassandras, “Perturbation analysis for production control and optimization of manufacturing systems,” *Automatica*, 2002, submitted.
- [13] —, “Perturbation analysis of feedback-controlled stochastic flow systems,” in *Proceedings of the IEEE Conference On Decision and Control*, 2003, pp. 6277–6282.
- [14] K. Ramakrishnan and S. Floyd, “The addition of explicit congestion notification (ENC) to IP,” *IETF RFC 3168*, Sep 2001, <http://www.ietf.org/rfc/rfc3168.txt>.
- [15] 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.

⁵Assuming that RTT=0, then unbiasedness follows easily from the results in [12]