Sensitivity Analysis for Discrete-Time Randomized Service Priority Queues*

George Kesidis[1], Takis Konstantopoulos[2], Michael Zazanis[3]

1. Elec. & Comp. Eng. Dept, University of Waterloo, Waterloo, ON, Canada, N2L 3G1
2. Dept. of Elec. & Comp. Eng., University of Texas, Austin TX 78712
3. Dept. of Industrial Eng., University of Massachusetts, Amherst MA 01003

Abstract

We consider a collection of queues operating in parallel and sharing a common server via a "randomized" service priority sharing rule. We assume that the service times of the customers (cells) are all constant and identical. This paper is concerned with estimating the sensitivity of the tail of the distribution of the typical sojourn time through these queues to the fraction of server bandwidth given to them by randomized service sharing. Two approaches are considered: "smoothed" perturbation analysis and stochastic intensity-based estimators.

1 Introduction

In future high-speed, integrated networks, different kinds of traffic will have different performance requirements from the network. For instance, a voice call may require a small end-to-end delay (delay sensitive) and may tolerate significant packet loss[1]. A data call, however, may require an extremely small packet loss probability (loss sensitive) and may tolerate significant delay. A highly compressed, real-time video call and network signaling traffic may be both loss and delay sensitive. Finally, a "casual" electronic mail message may be neither loss nor delay sensitive (see [9]). To achieve these different requirements, some kind of sharing rule for the buffers and servers in the network could be implemented so that certain calls receive priority for service and/or buffers. In this paper, we focus on the buffer design wherein each traffic class occupies a separate buffer and the buffers share a server

*Supported by NSERC of Canada and NSF Research Initiation Award NCR-9211343

[1] Voice is actually an isochronous traffic stream requiring small delay *jitter*, but by reducing the maximum end-to-end delay, jitter can be eliminated by a small buffer at the destination.

via the "randomized" service priority sharing rule [10]. We also assume that the cells all require service times that are constant and identical. In the following, we will use the words "queue" and "buffer" interchangeably.

We now describe the operation of the randomized service sharing on a group of K queues with independent, stationary and ergodic sources. In "simple" (or "state - independent") randomized service, the server is assigned to a queue; when it finishes serving a single cell, the server is then assigned to possibly another queue. If the server is assigned to an empty queue, it waits there for an amount of time equal to the service time of one cell; so, simple randomized service is not work conserving. The assignment process is determined by i.i.d. random variables $\{\gamma_n\}$ that are uniformly distributed over $[0,1]$. The interval $[0,1]$ is partitioned into K smaller intervals $\{A_i\}_{i=1}^{K}$ so that if $\gamma_n \in A_i$, then the server will be assigned to the i^{th} queue in the n^{th} service epoch.

Note that the generation of the sequence $\{\gamma_n\}$ can be done in parallel with the operation of the buffers. Also note that randomized service sharing is a way to share the bandwidth of the output link without the use of frames [5] (thus saving the overhead of framing). Finally, simple randomized service lends itself to circuit-switched style admission control in the sense that an effective bandwidth result has been obtained in [2] for it. All of these qualities make randomized service sharing appealing to ATM networks [3].

Consider the i^{th} queue sharing the link bandwidth. Let $\theta = P(\gamma_n \in A_i)$ be the fraction of service bandwidth this queue receives on average. In this paper, we are concerned with estimating the sensitivity

$$\eta \quad := \quad \frac{d}{d\theta} P(T(\theta) \geq B) \tag{1}$$

where $T(\theta)$ is the sojourn time of a typical cell in this particular queue. (To ease the notation we omit the index i.) The choice of θ clearly depends on performance requirements of the traffic using this queue. For reliable network design, the choice of θ should depend on η as well. Possibly η could also be used by the network to dynamically change the parameter θ in reaction to observed excessive accumulation of voice packets.

This note is organized as follows. In Section 2, we describe the smoothed perturbation analysis estimator of the sensitivity in equation (1). In Section 3, we describe a recursive update formula for the estimate of the sensitivity to be used in a simulation. In Section 4, we present a method based on the updating of the stochastic intensity of the arrival process. In Section 5, simulation results are described. Finally, conclusions are drawn in Section 6.

2 Smoothed Perturbation Analysis

Consider a "G/D/RS" queue; i.e., a queue with discrete-time stationary and ergodic arrivals, each arrival (called a "cell") requires one unit of time of service, and this queue shares the server with other queues according to the randomized service priority discipline. Let θ be the fraction of service bandwidth this queue receives on average. We want to find

$$\frac{d}{d\theta}P(T(\theta) \geq B)$$

where $T(\theta)$ is the sojourn time of a typical cell.

Assume that this queue is simulated for N cell departures and to the n^{th} arrival ($n = 1, ..., N$) associate an infinite sequence of i.i.d. uniform $[0, 1]$ random variables $\{\xi_i^n\}_{i=1}^{\infty}$. The amount of "virtual service" required by the n^{th} cell is

$$\sigma_n(\theta) = \inf\{i : \xi_i^n \leq \theta\}. \tag{2}$$

Note that $\sigma_n(\theta)$ has a geometric distribution. Define the random vector

$$\sigma(\theta) := (\sigma_1(\theta), \ldots, \sigma_N(\theta)).$$

Now consider two simulations of the G/D/RS queue: one using the parameter θ and the other using $\theta - \Delta\theta$ where $0 < \Delta\theta << 1$. Only the θ-simulation will actually be conducted. Below we describe how to estimate $dP(T(\theta) \geq B)/d\theta$ given the results of the θ-simulation alone.

Using the notation of Suri [11], for the θ-simulation of length N cell departures define:

$$\begin{aligned}
BP_j &= \text{the } j^{th} \text{ busy period} \\
t_j &= \text{the starting time of } BP_j \\
&= \text{the } j^{th} \text{ arrival time of a cell to an empty queue} \\
s_j &= \text{the ending time of } BP_j \\
&= \text{the } j^{th} \text{ departure time of a cell leaving the queue empty} \\
M &= \text{the number of } BP\text{'s for the } N \text{ arrivals simulated} \\
k_{j+1} &= \text{the index of the last cell of } BP_j \text{ with } k_0 = 0 \\
\delta_j &= t_j - s_{j-1} \\
\Delta_j^i &= \begin{cases} \sum_{k=j+1}^{i} \delta_k & \text{for } i > j \\ 0 & \text{for } i = j \end{cases} .
\end{aligned}$$

Let $\mathbf{A} = \{x \in \mathbf{R}^N \mid x = k\mathbf{e}_j,\ k = 0,1,2,...,\ j = 1,2,3,...\}$ where $\mathbf{e}_j$ is the unit vector whose j^{th} entry is 1 and all other entries are 0. Note that:

$$\begin{aligned}
P(\sigma(\theta - \Delta\theta) - \sigma(\theta) = 0 \mid \sigma(\theta)) &= (1 - \Delta\theta/\theta)^N \ = \ 1 + o(\Delta\theta), \\
P(\sigma(\theta - \Delta\theta) - \sigma(\theta) = k\mathbf{e}_j \mid \sigma(\theta)) &= (\theta - \Delta\theta)(1 - \theta + \Delta\theta)^{k-1}\frac{\Delta\theta}{\theta}(1 - \frac{\Delta\theta}{\theta})^{N-1} \\
&= (1-\theta)^{k-1}\Delta\theta + o(\Delta\theta), \\
P(\sigma(\theta - \Delta\theta) - \sigma(\theta) \notin \mathbf{A} \mid \sigma(\theta)) &= o(\Delta\theta).
\end{aligned}$$

Thus to find an expression for an estimate of

$$\begin{aligned}
\frac{d}{d\theta}P(T(\theta) \geq B) &= \lim_{\Delta\theta \downarrow 0} \frac{P(T(\theta) \geq B) - P(T(\theta - \Delta\theta) \geq B)}{\Delta\theta} \\
&= \lim_{\Delta\theta \downarrow 0} \frac{E\left(E\left[\mathbf{1}\{T(\theta) \geq B\} - \mathbf{1}\{T(\theta - \Delta\theta) \geq B\} \mid \sigma(\theta), \text{Arrivals}\right]\right)}{\Delta\theta},
\end{aligned}$$

we need only consider the case where a single cell's virtual service extends when $\theta \to \theta - \Delta\theta$. Note that $E(\mathbf{1}\{T(\theta) \geq B\} \mid \sigma(\theta), \text{Arrivals}) = E(\mathbf{1}\{T(\theta) \geq B\}) = P(T(\theta) \geq B)$.

By an argument similar to that in [11], section VI.B (the "smoothed perturbation analysis" of Gong and Ho [4]) an estimate of $dP(T(\theta) \geq B)/d\theta$ is

$$\hat{\eta}(N) \ := \ -\frac{1}{N}\sum_{j=1}^{M}\sum_{n=k_j+1}^{k_{j+1}}\sum_{a=1}^{\infty}(1-\theta)^{a-1}R(j,n,a) \tag{3}$$

where

$$R(j,n,a) \ := \ \sum_{b=j}^{M}\sum_{m=\max\{n,k_b+1\}}^{k_{b+1}} \mathbf{1}\{B - [a - \Delta_j^b]^+ \leq T_m < B\} \tag{4}$$

where T_m is the sojourn time of the m^{th} cell in the θ-simulation.

Explanation of equations (3) and (4):
Cell n residing in BP_j has its virtual service time extended by a units when $\theta \to \theta - \Delta\theta$. Consequently, some busy periods, beginning with BP_j, may coalesce. The result is that the sojourn times of some cells will increase when $\theta \to \theta - \Delta\theta$:
for $m = n, ..., k_{j+1}$, $T_m(\theta - \Delta\theta) - T_m(\theta) = a$,
for $m = k_{j+1} + 1, ..., k_{j+2}$, $T_m(\theta - \Delta\theta) - T_m(\theta) = [a - \delta_{j+1}]^+ = [a - \Delta_j^{j+1}]^+$,
for $m = k_{j+2} + 1, ..., k_{j+3}$, $T_m(\theta - \Delta\theta) - T_m(\theta) = [a - \Delta_j^{j+2}]^+$,
etc.
Equation (4) then follows from

$$\begin{aligned}
\mathbf{1}\{T_m(\theta) \geq B\} - \mathbf{1}\{T_m(\theta - \Delta\theta) \geq B\} &= \mathbf{1}\{T_m(\theta) \geq B\} - \mathbf{1}\{T_m(\theta) + [a - \Delta_j^b]^+ \geq B\} \\
&= -\mathbf{1}\{B - [a - \Delta_j^b]^+ \leq T_m(\theta) < B\}
\end{aligned}$$

3 Recursive Update Formula for the Estimate

In this section we find a simple formula for $\delta(N+1) := \hat{\eta}(N+1) - \hat{\eta}(N)$. This formula can be used to update our estimate of the sensitivity after every cell departure of the θ-simulation.

In the following we assume that the random quantities k_j and M are evaluated for first N departed cells. Consider two cases for the cell $N+1$:

Case 1) Cell $N+1$ is a member of BP_M. In this case we get that

$$\begin{aligned}\delta(N+1) &= -\sum_{a=1}^{\infty}(1-\theta)^{a-1}\mathbf{1}\{B-a \le T_{N+1} < B\} \\ &\quad -\sum_{j=1}^{M}\sum_{n=k_j+1}^{k_{j+1}}\sum_{a=1}^{\infty}(1-\theta)^{a-1}\mathbf{1}\{B-[a-\Delta_j^M]^+ \le T_{N+1} < B\} \\ &= -\frac{\mathbf{1}\{T_{N+1}<B\}}{\theta}\left((1-\theta)^{B-T_{N+1}-1} + \sum_{j=1}^{M}(k_{j+1}-k_j)(1-\theta)^{B-T_{N+1}+\Delta_j^M-1}\right).\end{aligned}$$

Note that the first term in the equations above is due to the virtual service time of cell $N+1$ increasing by a. The second term is due to the virtual service time of one of the first N cells increasing by a causing the queueing time of cell $N+1$ to increase.

Case 2) Cell $N+1$ is not a member of BP_M; i.e., cell $N+1$ begins the new busy period BP_{M+1}. In this case we get that

$$\begin{aligned}\delta(N+1) &= -\sum_{j=1}^{M+1}\sum_{n=k_j+1}^{k_{j+1}}\sum_{a=1}^{\infty}(1-\theta)^{a-1}\mathbf{1}\{B-[a-\Delta_j^{M+1}]^+ \le T_{N+1} < B\} \\ &= -\frac{\mathbf{1}\{T_{N+1}<B\}}{\theta}\left(\sum_{j=1}^{M+1}(k_{j+1}-k_j)(1-\theta)^{B-T_{N+1}+\Delta_j^{M+1}-1}\right).\end{aligned}$$

Note that $k_{M+2} - k_{M+1} = 1$ and the two cases yield very similar expressions.

4 An Alternative SPA Estimator

An alternative estimator can be obtained if in the above SPA analysis the perturbation $\Delta\theta$ is taken in the opposite direction. Denote by $W_t(\theta)$ the workload process when the parameter value is equal to θ (the *nominal* sample path. From (2) it follows that $\sigma(\theta + \Delta\theta) \le \sigma(\theta)$ w.p.1 and hence that the "lucky customers" of the process in the nominal path will remain "lucky" in the perturbed path as well. P^0 designates the Palm probability w.r.t. the arrival process while P^* Palm probability w.r.t. the lucky arrivals with parameter value θ. The cycle formula between these two measures gives

$$\frac{1}{\Delta\theta}[P^0(T_0(\theta+\Delta\theta) > B) - P^0(T_0(\theta) > B)] = \frac{1}{E^*Q} E^*[\sum_{i=0}^{Q-1} 1(T_i(\theta+\Delta\theta) > B) - 1(T_i(\theta) > B)], \tag{5}$$

where Q is the number of customers in the first busy period of the nominal sample path. In general terms the difference between this approach and that of §2 is that we now need only worry about a single busy period breaking up instead of several busy periods coalescing. To implement the SPA algorithm we will condition w.r.t $\mathcal{F}$, the whole history of the *nominal* sample path. An analysis similar to that of §2 gives

$$\frac{d}{d\theta} P^0(T_0(\theta) > B) = \frac{1}{(1-\theta)E^*Q} E^*[\sum_{i=0}^{Q-1} \sum_{j=i}^{Q-1} \sum_{k=1}^{\sigma_i(\theta)-1} 1(B < T_i(\theta) \le B + L_{ij} \wedge k)], \tag{6}$$

where

$$L_{ij} = \min\{W_{i+1}, \dots, W_j\}, \qquad j \ge i,$$

with the convention that the minimum element of the empty set is $+\infty$.

5 Stochastic Intensity Based Estimators

Consider again a G/D/RS queue but assume that the server has nonzero setup times. This means that it takes the server a nonzero amount of time to switch from one queue to another. This situation may be encountered in practice. The setup times are small compared to the service times (still taken to be of unit duration) but random. Let $f_s(x)$ denote their common density, supported on the interval $[0, \epsilon]$, where $\epsilon < 1$. The virtual service time $\sigma_n(\theta)$ of the n^{th} cell is now seen to have density

$$g(\theta, x) = \sum_{k=1}^{\infty} \theta(1-\theta)^{k-1} f^{(k)}(x), \tag{7}$$

where $f(x) = f_s(1 + x)$, and $f^{(k)}(x)$ is the k-fold convolution of f with itself. Let $G(\theta, x) = \int_0^x g(\theta, y)dy$ be corresponding distribution function. The random variable $\sigma_n(\theta)$ is now generated by the formula $\sigma_n(\theta) = G^{-1}(\theta, \xi_n)$, where G^{-1} is the inverse function of G with respect to the second variable, and ξ_n is a sequence of i.i.d. random variables, uniformly distributed in the interval $[0, 1]$. Finally, we let

$$\sigma'_n(\theta) = \frac{d}{d\theta} G^{-1}(\theta, \xi_n). \tag{8}$$

It can be seen that this derivative is defined (i.e., it is finite) for all ξ_n outside an interval of size of the order of $1-(1-\theta)^{1/\epsilon}$. This is due to the nature of the the density g defined in (7) as the sum of convolutions of a density f supported on an interval of size ϵ. It is natural to require that this size be small so that the error in the algorithm to follow is negligible. We thus need $(1-\theta)^{1/\epsilon} \approx 1$. This is for instance the case if ϵ is large or if θ is small. It is thus conjectured that the algorithm works well for low priority classes.

These assumptions, namely in cases where one can afford an additional randomization for the service times, lead to a considerable simplification of the perturbation analysis estimator. Indeed, it is shown in [8], that an infinitesimal perturbation analysis estimator can be constructed. The construction is based on knowledge of the stochastic intensity, say α_t, of the arrival process of the queue under consideration. Recall that the stochastic intensity of a point process, c.f. [1], with respect to a σ-field $\mathcal{F}_t$ of observations (here taken to be the information of the simulated sample path of the queue under consideration up to time t), is defined by

$$\alpha_t = \lim_{\delta \to 0} \frac{1}{\delta} E[N(t, t+\delta)|\mathcal{F}_t],$$

where $N(t, t+\delta)$ is the number of arrivals between t and $t+\delta$. Note that not every point process has a stochastic intensity, but almost all models encountered in practice do. For instance, a renewal process has stochastic intensity (with respect to the σ-field of the sample path up to time t) zero when the queue is empty, and $h(a_t)$ otherwise. Here h is the hazard rate of the interarrival time ($h(x)$ defined as $f(x)/(\int_x^\infty f(y)dy)$, with f being the density of the interarrival time), and a_t is the distance between time t and the previously observed arrival. Likewise, simple formulas for the stochastic intensity can be found for most models used in practice (e.g., Markov modulated Poisson processes).

Let $W_t(\theta)$ be the total work in the queue at time t, as accounted for by the (remaining) virtual service times of the cells in the queue. Let now $W(\theta)$ be the total queueing delay of a typical cell in steady state. This is related to the sojourn time $T(\theta)$ by $T(\theta) = W(\theta) + \sigma_0(\theta)$, where $\sigma_0(\theta)$ is a typical virtual service time with density g as above. We now simulate the queue for a total of N cell arrivals (say T_N is the time of the N^{th} arrival) and define the following quantities. Let $D_t(\theta)$ be the sum of the derivatives of the service times, given by (8), of all cells from the start of the busy period that contains t, up to the last cell arriving before t. If t is in an idle period then $D_t(\theta)$ is taken to be zero. We are interested in estimating the derivative of $P(W(\theta) > x)$. The queue is simulated for N cell arrivals, and observations are made at times at which the total work $W_t(\theta)$ *downcrosses* level x. Call S_j these times. It is shown in [8] that an estimator for $\frac{d}{d\theta}P(W(\theta) \geq x)$ is given by the quantity

$$\zeta(N) := \frac{1}{N} \sum_{j \geq 1,\ S_j < T_N} D_{S_j}(\theta)\alpha_{S_j}.$$

In other words, at each downcrossing time S_j, $j = 1, 2, \ldots$, the stochastic intensity α_{S_j} is computed and multiplied by the accumulated perturbation $D_{S_j}(\theta)$. The sum of these products up to the N^{th} arrival divided by N gives a simple expression for the sensitivity estimator. The reader is referred to [8] for the relevant details and proofs.

6 Simulations

Results of the following simulations are pending: A discrete-time Markov source described in [6] is used. We compare the result of the smoothed perturbation analysis estimators to that obtained by running two simulations (one using parameter θ and the other using $\theta - \Delta\theta$) and estimating the sensitivity by a difference quotient. Finally, we apply the algorithm of the previous section to estimated sensitivities associated with low priority classes.

7 Conclusions

We have described perturbation analysis estimators for the sensitivity of tail of the queuing delay distribution for a queue sharing a server via simple randomized service. We assumed a stationary and ergodic source of customers (cells) all requiring the same deterministic amount of service. Currently, we are working on this problem for state-dependent (in particular, work conserving) randomized service disciplines.

References

[1] P. Brémaud. *Point Processes and Queues.* Springer, 1981.

[2] G. de Veciana and J. Walrand. Effective bandwidths: Call admission, traffic policing & filtering for ATM networks. *preprint*, 1993.

[3] J. Filipiak. *Real Time Network Management.* North-Holland, New York, NY, 1991.

[4] W.B. Gong and Y.C. Ho, Smoothed (conditional) perturbation analysis of discrete event dynamic systems. *IEEE Trans. Auto. Control*, Vol. 32, No. 10: pp. 858–866, 1987.

[5] J.Y. Hui. *Switching and Traffic Theory for Integrated Broadband Networks.* Kluwer Acad. Publ., Boston, 1990.

[6] G. Kesidis, J. Walrand, and C.-S. Chang. Effective bandwidths for multiclass Markov fluids and other ATM sources. *to appear in IEEE/ACM Trans. Networking*, Aug. 1993.

[7] T. Konstantopoulos and M. Zazanis. Sensitivity Analysis for Stationary and Ergodic Queues. *Adv. Appl. Prob.*, Vol. 24, 738-750, 1992.

[8] T. Konstantopoulos and M. Zazanis. Stochastic Intensity Based Sensitivity Estimators for Stationary and Ergodic Queues. *Preprint*, Aug., 1993.

[9] A. A. Lazar, G. Pacifici, and J. S. White. Real-time traffic measurement on MAGNET II. *IEEE JSAC*, Vol. 8, No. 3:467–493, April 1990.

[10] J.M. Pitts and J.A. Schormans. Analysis of ATM switch model with time priorities. *Electronic Letters*, Vol. 26, No. 15:1192,1193, July 1990.

[11] R. Suri. Perturbation analysis: the state of the art and research issues explained via the GI/G/1 queue. *Proceedings of the IEEE*, vol. 77, no. 1:114–137, 1989.