Perturbation Analysis of Closed Queueing Networks with General Service Time Distributions

Xi-Ren Cao

Abstract—The note studies perturbation analysis of closed queueing networks with nonexponential service time distributions. Perturbation analysis formulas utilizing utilization probabilities are extended to these networks. A perturbation generation function, which generalizes the perturbation generation rule, is defined; equations for realization probability and formulas for sensitivity of the system throughput with respect to service time distribution parameters are presented. The formulas provide a new analytical method of calculating throughput sensitivity and an explanation of the application of perturbation analysis algorithms for general service time distributions. The note emphasizes the main concepts and makes no attempt at the rigor of the derivations of the formulas.

I. INTRODUCTION

Perturbation analysis (PA) has been proposed as an efficient method of estimating the performance sensitivity with respect to a parameter in a discrete-event simulation [1], [2]. The PA algorithms for estimating the derivative of the system throughput of a closed Jackson queueing network were proposed, and analytical formulas for calculating the PA estimate were developed [3]. The formulas were rigorously proved and the convergence properties of the PA estimates were studied [4]. The analytical formulas provide a theoretical foundation for PA of Jackson networks. The PA algorithms can be viewed as practical implementations of these formulas.

The purpose of this note is to extend the results in [3] and [4] to networks with nonexponential service distributions. Closed-form solutions do not exist for the steady-state probabilities of these networks. One distinction between such a network and the Jackson network is that in networks on general service time distributions, the perturbations generated in different periods of a customer’s service time usually have different sizes even if the lengths of the periods are the same. To capture this feature, a perturbation generation function is defined in this note. The generation function is an extension of the perturbation generation rule. The main result of this note is that the normalized derivative of the system throughput equals the negative expected value of the product of the generation function and the realization probability. Furthermore, a set of equations specifying the realization probability is presented.

In this note, we shall focus on the extension of concepts and intuitive explanations of the formulas rather than on mathematical derivations. In fact, the concepts and intuitions behind the lengthy mathematical expressions are more important for practical purposes.

In many cases, including the results in [3] and [4], analytical formulas were first proposed based on intuitions before rigorously proved. This note is an analog of [3] for networks with general service distributions. No attempt at rigor is made. Readers will be referred to [3] and [4] for some common concepts in perturbation analysis.

II. THE NETWORK

Consider a closed network consisting of $N$ customers and $M$ servers. Let $q_{ij}$, $i, j = 1, 2, \cdots, M$, be the routing probabilities. We assume that the routing matrix $Q = (q_{ij})_{i,j=1}^{M}$ is irreducible.

The buffer size of each server is infinite (i.e., no less than $N$) and the service discipline is first come first served. The service times of a customer at server $i$ are independent and have a cumulative distribution function $F_{j}(s)$, $i = 1, 2, \cdots, M$. We assume that the probability density functions $f_{j}(s) = dF_{j}(s)/ds$, $i = 1, 2, \cdots, M$, exist and $0 \leq f_{j}(s) < \infty$ for all $0 \leq s < \infty$.

Let $x = (n, r)$ be the state of the system, where $n = (n_{1}, n_{2}, \cdots, n_{M})$ is the discrete part of the state with $n_{i}$ being the number of customers in server $i$, and $r = (r_{1}, r_{2}, \cdots, r_{M})$ is the continuous part of the state with $r_{i}$ being the expected service time of the customer being served at server $i$. The state process is denoted as a right-continuous vector process $\mathcal{X}(t) = (\mathcal{A}(t), \mathcal{R}(t))$. We employ the following notations: $r + \Delta t = (r_{1} + \Delta t_{1}, r_{2} + \Delta t_{2}, \cdots, r_{M} + \Delta t_{M})$, $r_{-i} = (r_{1}, r_{2}, \cdots, r_{i-1}, 0, \cdots, r_{M})$, and $r_{-i,j} = (r_{1}, \cdots, r_{j}, 0, \cdots, r_{M})$.

We also define $r + \Delta t_{j} (r \geq r'_{j})$ as $r_{j} > r'_{j}$ for all $i$ and denote by $R_{n}^{M}$ the quadrant with $r_{j} \geq 0$, $i = 1, 2, \cdots, M$, in the $M$-dimensional real space $R^{M}$.

Let $P(x, t) = P(n, r, t) = Pr[\mathcal{A}(t) = n, \mathcal{R}(t) \leq r]$ be the probability of the event $\{\mathcal{A}(t) = n, \mathcal{R}(t) \leq r\}$ and $p(x, t) = p(n, r, t) = (d/dt)P(n, r, t)$ be the probability density function of the state process. We assume that the state process is ergodic and the steady-state probability density $\pi(x) = \pi(n, r)$ exists, i.e., $p(x, t) = \pi(x)$, as $t \rightarrow \infty$. This requires the routing probability matrix $Q = (q_{ij})$ to be irreducible.

A perturbation of $Q$ has a positive probability of reaching every other server in the network, either directly or by going through some other servers. The irreducibility of $Q$ is also sufficient for the stability (the existence of a unique steady-state distribution $\pi(x)$) if all the service distributions are exponential [4]. Some conditions for stability of networks with general service distributions can be found in [5]–[7].

III. PERTURBATION GENERATION

The three basic elements of perturbation analysis are: perturbation generation, perturbation propagation, and perturbation realization. The essential idea of PA is, roughly speaking, as follows. A change in a service time distribution parameter of a server induces changes in the service times of the customers in the server. The change in a service completion time of a server is called a perturbation of that server. The perturbation induced by a parameter change can be determined by the perturbation generation rule. A perturbation of a server will affect the service completion times of other servers. This is described by perturbation propagation rules. The final effect of a perturbation to the system can be described by perturbation realization. The sensitivity of a performance with respect to the parameter change can be determined by taking account of the final effects of all the perturbations induced by the parameter change.

We discuss perturbation generation of service times with general distributions in this section. Assume that a service distribution parameter, $\theta$, changes to $\theta + \Delta \theta$. The service time of a customer corresponding to $\theta$ is $s = F^{-1}(\xi, \theta)$, where $\xi \in (0, 1)$ is a uniformly distributed random variable. In the system with parameter $\theta + \Delta \theta$ (called the perturbed system, to be distinguished from the system with parameter $\theta$, called the nominal system), the service time of the customer is

$$s' = F^{-1}(\xi, \theta + \Delta \theta).$$
Let $s' = s + \Delta s$. Then we have

$$\Delta s = F^{-1}(\xi, \theta + \Delta \theta) - F^{-1}(\xi, \theta) = \frac{\partial F^{-1}(\xi, \theta)}{\partial \theta} \bigg|_{\xi = F(\xi, \theta)} \Delta \theta.$$  \hspace{1cm} (1)

$\Delta s$ is the perturbation generated during the service period because of $\Delta \theta$. In (1), the same random variable $\xi$ is used for both $s$ and $s'$, and we assume that the partial derivative exists and that the higher order terms in the Taylor expansions can be ignored. Equation (1) is called the perturbation generation rule [8]. $\Delta \theta$ can be chosen as small as desired so that the deterministic similarity between the nominal path (a sample path of the nominal system) and the perturbed path (a sample path of the perturbed system) holds; i.e., the state sequences of the two sample paths are the same. The deterministic similarity guarantees that the infinitesimal propagation rules (see Section IV) can be applied (for details, see [3] and [4]). The size of $\Delta \theta$ can be arbitrarily chosen because the perturbations generated are proportional to $\Delta \theta$, and the performance sensitivity is determined by the ratio of the final perturbation and $\Delta \theta$ and, therefore, is independent of $\Delta \theta$ (see (10) in Section V).

An important concept extension in this note is, the perturbation generated in the entire service period can be "spread" throughout the service period. Let $s$ be the service time, then $\Delta s$ shown in (1) is the perturbation generated in the entire service period. Let $r_0 \leq r \leq s$ be the elapsed service time and $\rho = F(\xi, \theta)$. As $r$ increases from $0$ to $s$, $\rho$ increases from $0$ to $\xi$. Now, what is the perturbation generated in a small interval $[r, r + dr]$? Again, if $\theta$ changes to $\theta + \Delta \theta$, the elapsed time $r$ changes to $r + \Delta r$ (corresponding to the same $\rho$) with (see Fig. 1).

$$\Delta r = \frac{\partial F^{-1}(\rho, \theta)}{\partial \theta} \bigg|_{\rho = F(r, \theta)} \Delta \theta.$$  

$\Delta r$ can be viewed as the perturbation generated in the elapsed service time period $[0, r]$ due to $\Delta \theta$. Therefore, the perturbation generated in the service interval $[r, r + dr]$ is

$$d(\Delta r) = \frac{\partial F^{-1}(\rho, \theta)}{\partial \theta} \bigg|_{\rho = F(r, \theta)} \Delta \theta dr = G(r, \theta) \Delta \theta dr,$$  \hspace{1cm} (2)

where

$$G(r, \theta) = \frac{\partial}{\partial r} \left\{ \frac{\partial F^{-1}(\rho, \theta)}{\partial \theta} \bigg|_{\rho = F(r, \theta)} \right\}$$

is called the perturbation generation function. The distinction between $\Delta r$ and $dr$ is clearly shown in Fig. 1: $\Delta r$ is the perturbation generated in $[0, r]$ due to $\Delta \theta$, and $dr$ is the length of a small interval. Also, as shown in Fig. 1, $d(\Delta r)$ is the perturbation generated in $[r, r + dr]$.

If the distribution takes the form $f(r/g(\theta))$, where $f$ and $g$ are two differentiable functions, then

$$r = F^{-1}(\rho, \theta) = g(\theta) F^{-1}(\rho)$$

and

$$G(r, \theta) = \frac{\partial}{\partial r} \left\{ \frac{dg(\theta)}{d\theta} f^{-1}(\rho) \bigg|_{\rho = f(r, g(\theta))} \right\} = \frac{dln(g(\theta))}{d\theta}.$$  

Therefore, the perturbation generated in any service interval $[r, r + dr]$ is $dln(g(\theta)) d\theta dr$, which is proportional to $dr$. In particular, this linear property holds for exponential service distributions. In this case, $F(s, \theta) = 1 - \exp(-s\theta)$ with $\theta = \mu$ being the mean service rate, we have $r = F^{-1}(\rho, \theta) = -(1/\theta)ln(1 - \rho)$ and $G(r, \theta) = -1/\theta$.

IV. PERTURBATION PROPAGATION AND REALIZATION

As discussed in Section III, a change $\Delta \theta$ in server $i$'s service time distribution induces a perturbation $G(r, \theta) \Delta \theta dr$ at server $i$ during the service period $[r, r + dr]$. (A subscript $i$ is added in the expression.) This perturbation will be propagated to the end of the service completion time of the server and then to other servers. Note that perturbations generated in other intervals $[r', r' + dr']$ are also propagated to (or accumulated at) the end of the service completion time and propagated to other servers. From (2), the perturbation generated in a service time $[0, s]$ is $\Delta s = \int G(r, \theta) \Delta \theta dr$. The effect of all these perturbations equals the sum of the effect of each perturbation being propagated separately. Thus, in perturbation analysis we first consider the effect of a single perturbation and then determine the effect of a parameter change by adding up the effects of all the perturbations generated by this parameter change [4]. The perturbation propagation rules are the same as those for Jackson networks [3], which are stated below.

Perturbation Propagation Rules

1) A server will keep its perturbation (real or null) until the end of an idle period. (If $n_i = 0$, then server $i$ is said to be in an idle period.)

2) If a customer from server $i$ terminates an idle period of server $j$, i.e., server $j$ receives a customer from server $i$ to start a new busy period, then after this idle period server $j$ will have the same perturbation (real or null) as server $i$. In this case, we say that the perturbation of server $i$ is propagated to server $j$.

The propagation rules hold if the perturbed and the nominal paths are deterministic similar. This can be achieved by choosing $\Delta \theta$ sufficient small (see [3] and [4]).

Now let us consider the propagation of a single perturbation with a sufficient small size. Because of propagation, the perturbation will be propagated from servers to servers. The size of the perturbation is assumed to be small enough so that the perturbation propagation rules described above hold. (For steady-state performance, this raises the well-known question about the interchangeability of the derivative and the expectation. Regarding to this issue, there is no essential difference between the network discussed here and that studies in [4]. We shall not provide a detailed explanation for this.) Note that some servers may lose the perturbation after obtaining it. Based on the propagation rules, the size of the perturbation does not...
change during propagation. Let \( \Gamma = \{1, 2, \ldots, M\} \) and \( V \) be a subset of \( \Gamma \). We say that \( V \) is a perturbation set of the system, meaning that all the servers in \( V \) have the perturbation (with the same size) and all the servers in \( U = \Gamma - V \) do not have the perturbation. Because of propagation, \( V \) and \( U \) depend on time \( t \).

By the propagation rules, it is easy to check that sets \( \Gamma \) and \( \emptyset \) are two absorbing sets of \( V \) (i.e., if \( V(t_0) = \Gamma \) or \( \emptyset \), then \( V(t) = \Gamma \) or \( \emptyset \) for all \( t > t_0 \)). Denote the \((x, V, t) = (n, r, V, t)\) the perturbation in set \( V \) at time \( t \) when the system state is \( J(t) = (n, r) \). We sometimes drop the time \( t \) and write the perturbation as \((x, V) = (n, r, V)\). Also, we write \((x, i) \) for \((x, \{i\})\).

Now let us extend the notation of realization to the networks under consideration. The definition of realization is just as the same that for Jackson networks which is stated as follows:

Definition 1: A perturbation \((x, V, t_0)\) is said to be realized (or lost) on a sample path if, as a result of propagation of the perturbation on the sample path, every server in the network eventually gets (or loses) the perturbation, i.e., \( V(t) = \emptyset \) (or \( V(t) = \Gamma \)) for sufficiently large \( t \).

The following Theorem is an extension of the similar theorem for Jackson networks:

Theorem 1: In a closed network with an ergodic state process, any perturbation \((x, V)\) will, with probability one, be either realized or lost.

Proof: Since the state process is ergodic, any sample path (with probability one) will reach a state whose discrete part is \((N, 0, 0, \ldots, 0)\). Furthermore, for any perturbation \((x, V)\) at time \( t_0 \), if at some time \( t > t_0 \), \( V(t) = \emptyset \), then \( V(t) = \emptyset \), and \( i \in V(t) \) (or \( i \in U(t) \)), then according to the perturbation propagation rules, the perturbation carried by server \( i \) (real or null) will be propagated to all the other servers, and \( V(t) = \emptyset \) (or \( V(t) = \Gamma \)) for any \( t' > t \). This implies that the perturbation in \((x, V, t_0)\) is realized (or lost) after the system reaches state \((N, 0, 0, \ldots, 0)\). Therefore, if the state process is ergodic, any perturbation will, with probability one, be realized or lost.

Some explanation is in order. Any perturbation in \((x, V)\) at time \( t \) will be either realized or lost, depending on the evolution of the sample path after \( t \). Theorem 1 says that, given a perturbation at \((x, V)\), the sum of the probability that the perturbation will be realized and the probability that the perturbation will be lost equals one. The realization probability of a perturbation can be defined as follows:

Definition 2: The probability that a perturbation \((x, V)\) is realized is called the realization probability of the perturbation and is denoted as \( c(n, r, i) = c(n, r, V, i) \).

Note that the realization probabilities are independent of \( i \). Just as for Jackson networks, the realization probabilities satisfy some interesting properties stated below.

Theorem 2:

(i) If \( n_i = 0 \), then \( c(n, r, i) = 0 \).
(ii) \( c(n, r, \emptyset) = 1 \).
(iii) If \( V_1 \cap V_2 = \emptyset \) and \( V_1 \cup V_2 = V \),
then \( c(n, r, V_1) + c(n, r, V_2) = c(n, r, V) \).
(iv) \( c(n, r, 1) + c(n, r, 2) + \cdots + c(n, r, M) = 1 \).

Equation (3) involves a convention, i.e., we may consider an idle server as having a perturbation. This perturbation will be lost immediately after receiving a customer from any other server, which does not have the perturbation. Equation (4) follows directly from the “absorbing” property of the set \( \Gamma \). Equation (5) in fact holds in a sample path sense, i.e., the effect of a perturbation \((x, V)\) on a sample path equals the sum of the effects of the two perturbations \((x, V_1)\) and \((x, V_2)\) on the same sample path. Equation (6) is a consequence of (4) and (5). The proof of the theorem is essentially the same as that in [4] and is omitted here.

Next, we shall derive a differential equation for the realization probability. This can be done by considering the relation between the realization probability of a perturbation at server \( k \) at time \( t \), \((x, k, t)\), and the realization probabilities of perturbations at all servers at time \( t + \Delta t \). Specifically, we have

\[
\begin{align*}
\frac{d c(n, r, k, t)}{d t} & = \left(1 - \sum_{j=1}^{M} c(n_j) g(r_j) \Delta t \right) \\
& \quad + \sum_{i=1}^{M} \sum_{j=1}^{M} c(n_i) c(r_i) g(r_j) \Delta t_i \\
& \quad \cdot q_{i,j} \left[ c(n_i, r_{i-1} + \Delta t, k, t + \Delta t) + \sum_{j=1}^{M} c(n_j) \left[1 - c(n_j)\right] g(r_j) \Delta r_j \\
& \quad \cdot q_{i,j} \left[ c(n_{i,j}, r_{i-1} - j + \Delta t, k, t + \Delta t) + \sum_{j=1}^{M} \left[1 - c(n_j)\right] g(r_j) \Delta r_j \right] \\
& \quad + q_{i,j} \left[ c(n_{i,j}, r_{i-1} - j + \Delta t, k, t + \Delta t) + \sum_{j=1}^{M} \left[1 - c(n_j)\right] g(r_j) \Delta r_j \right]
\end{align*}
\]

where \( c(n_i) = 1 \) if \( n_i > 0 \), 0 if \( n_i = 0 \). In the above equation, \( g(r_i) = f(r_i) \) is the hazard rate representing the service completion rate of server \( i \) given that the elapsed time is \( r_i \); \( \Delta r_i = \Delta t, \Delta t + 1, \ldots, M; \Delta t \). \( c(n_i) \) represents the high order terms of \( \Delta t \). The first term on the right-hand side represents the probability of no transition in \([t, t+\Delta t]\). The second term represents the case in which a customer transfers from server \( i \) to server \( j \). The probability of a customer transfer to server \( j \) is \( c(n_i) \). In this case, there is no propagation and \( r_j = 0 \) after the transition. The third term represents the situation in which an idle period of server \( j \) is terminated by server \( i \) in \([t, t+\Delta t]\); in this case, \( r_j = 0 \) after the transition. Finally, the fourth term represents the perturbation propagation effect; if server \( k \) terminates an idle period of server \( j \), the perturbation will be propagated to server \( j \).

Substituting the following equation

\[
\begin{align*}
c(n, r, k, t + \Delta t) & = c(n, r, k, t) + \Delta t \sum_{i=1}^{M} \frac{d c(n, r, k, t)}{d t} \\
& = c(n, r, k, t) + \Delta t \sum_{i=1}^{M} \frac{d c(n, r, k, t)}{d t} \\
& = \left(1 - \sum_{i=1}^{M} c(n_i) g(r_i) \Delta t \right) \\
& \quad + \sum_{i=1}^{M} \sum_{j=1}^{M} c(n_i) c(r_i) g(r_j) \Delta t_i \\
& \quad \cdot q_{i,j} \left[ c(n_i, r_{i-1} + \Delta t, k, t + \Delta t) + \sum_{j=1}^{M} c(n_j) \left[1 - c(n_j)\right] g(r_j) \Delta r_j \\
& \quad \cdot q_{i,j} \left[ c(n_{i,j}, r_{i-1} - j + \Delta t, k, t + \Delta t) + \sum_{j=1}^{M} \left[1 - c(n_j)\right] g(r_j) \Delta r_j \right] \\
& \quad + q_{i,j} \left[ c(n_{i,j}, r_{i-1} - j + \Delta t, k, t + \Delta t) + \sum_{j=1}^{M} \left[1 - c(n_j)\right] g(r_j) \Delta r_j \right]
\end{align*}
\]

into (7) and letting \( \Delta t \to 0 \), we obtain

\[
\begin{align*}
\frac{d c(n, r, k, t)}{d t} & = c(n, r, k, t) + \sum_{i=1}^{M} \frac{d c(n, r, k, t)}{d t} \\
& = c(n, r, k, t) + \sum_{i=1}^{M} \frac{d c(n, r, k, t)}{d t} \\
& = \left(1 - \sum_{i=1}^{M} c(n_i) g(r_i) \Delta t \right) \\
& \quad + \sum_{i=1}^{M} \sum_{j=1}^{M} c(n_i) c(r_i) g(r_j) \Delta t_i \\
& \quad \cdot q_{i,j} \left[ c(n_i, r_{i-1} + \Delta t, k, t + \Delta t) + \sum_{j=1}^{M} c(n_j) \left[1 - c(n_j)\right] g(r_j) \Delta r_j \\
& \quad \cdot q_{i,j} \left[ c(n_{i,j}, r_{i-1} - j + \Delta t, k, t + \Delta t) + \sum_{j=1}^{M} \left[1 - c(n_j)\right] g(r_j) \Delta r_j \right] \\
& \quad + q_{i,j} \left[ c(n_{i,j}, r_{i-1} - j + \Delta t, k, t + \Delta t) + \sum_{j=1}^{M} \left[1 - c(n_j)\right] g(r_j) \Delta r_j \right]
\end{align*}
\]
Since the realization probability is independent of $t$, \( \delta c(n, r, k, t) / \delta t = 0 \) and c(n, r, k, t) = c(n, r, k). Finally, we have the following.

**Theorem 3:** For a closed queueing network, the realization probability satisfies:

\[
\sum_{i=1}^{M} \frac{\partial c(n, r, k)}{\partial r_i} = c(n, r, k) \sum_{j=1}^{M} \epsilon(n_j) g_j(r_j) - \sum_{i=1}^{M} \sum_{j=1}^{M} \epsilon(n_j) \epsilon(n_i) g_i(r_i) - \sum_{i=1}^{M} \sum_{j=1}^{M} \sum_{k=1}^{k} \epsilon(n_j) \epsilon(n_i) g_i(r_i) - \sum_{i=1}^{M} \sum_{j=1}^{M} \sum_{k=1}^{k} \epsilon(n_j) \epsilon(n_i) g_i(r_i)\]

Equations (3), (6), and (8) specify the realization probabilities. If all the service time distributions are exponential (i.e., \( E_i(s) = 1 - \exp(-\mu_i s) \)), then the system state does not depend on \( r \), nor does the realization probability. Therefore, \( \delta c(n, r, k) / \delta r = 0 \). Note that \( g_i(r_i) = \mu_i \) is the mean service rate of server \( i \). In this case, (8) reduces to

\[
\sum_{i=1}^{M} \sum_{j=1}^{M} \epsilon(n_j) \mu_i c(n, k) - \sum_{i=1}^{M} \sum_{j=1}^{M} \epsilon(n_j) \mu_i c(n, k) + \sum_{i=1}^{M} \sum_{j=1}^{M} \sum_{k=1}^{k} \epsilon(n_j) \epsilon(n_i) g_i(r_i) - \sum_{i=1}^{M} \sum_{j=1}^{M} \sum_{k=1}^{k} \epsilon(n_j) \epsilon(n_i) g_i(r_i) = 0 \]

This is the same as the equation derived in [3] and [4] for networks with exponential servers.

V. DERIVATIVES OF SYSTEM THROUGHPUT

Now, suppose that \( \theta \) is a parameter of the service distribution function of server \( i \), and we want to estimate the derivative of the system throughput with respect to \( \theta \). Given a nominal service path (with parameter \( \theta \)) of finite length, we can construct a perturbed service path (with parameter \( \theta + \Delta \theta \)) by applying perturbation generation and propagation rules. The procedure can be formally described by the following algorithm. (In the algorithm, \( s_{ij} \) is the service time of the \( k \)th customer of server \( i \).)

**Perturbation Analysis Algorithm**

1) At time \( t = 0 \), set \( \Delta_t = 0 \), for \( j = 1, 2, \ldots, M \).

2) At the \( k \)th service completion time of server \( i \), set \( \Delta_t = \Delta_{t+1} + \Delta \).

\[
\Delta \theta = \frac{\partial F^{-1}(\xi_{i, j}, \theta)}{\partial \theta} |_{\xi_{i, j} = F^{-1}(\xi_{i, j}, \theta)}
\]

is the perturbation generated in \( s_{ij} \).

3) After every interval period of server \( i, k = 1, 2, \ldots, M \), set \( \Delta_t = \Delta_{t+1} \), where server \( j \) is the server which terminates the idle period.

The PA algorithm provides all the perturbations for all the servers. Let \( t_{i, j} \), \( j = 1, 2, \ldots, M \), \( k = 1, 2, \ldots, k \), be the service completion time of the \( k \)th customer of server \( j \) in the nominal system; then the service completion times of the perturbed system can be determined by \( t_{i, j} = t_{i, j} + \Delta \), with \( \Delta_0 \) being the perturbation of service \( i \) for the corresponding service completion time \( t_{i, j} \).

As discussed in Section III, \( \Delta S_{ij} \) can be spread throughout the service period. The perturbation generated in \( [r_i, r_i + dr] \) is \( d(\Delta S_{ij}) \).

Using the perturbation path, we can calculate the system throughput of the perturbed system and, further, obtain the sensitivity of the system throughput. Suppose that we observe the system in steady state for a time period \([0, T]\), with \( T \) being a customer transition time. Let \( L \) be the number of customer transitions in \([0, T]\).

Then the system throughput, \( \eta \), can be estimated as \( \eta = L / T \). Let \( \Delta \theta \) be the perturbation of \( T \) due to \( \Delta \). This means that the perturbed system completes the \( L \) customer transitions in time period \([0, T + \Delta T]\).

Then the estimated throughput for the perturbed system is \( \eta' = L / T + \Delta T \). Thus,

\[
\frac{\Delta \eta}{\eta} = \frac{\Delta T}{T},
\]

Now, recall that the steady-state probability of the system being in state \((n, r^*)\), \( r^* < r + dr \), is \( x(n, r) \). The total time in \([0, T]\) during which the system is in such states is \( T x(n, r) \). Let \( r_i \) be the \( i \)th component of \( r \). According to the perturbation generation rule, the sum of all the perturbation generated in \([0, T]\) due to the parameter change \( \Delta \theta \) when the system is in states \((n, r^*)\), \( r^* < r + dr \), is \( T x(n, r) G_i(r, \theta) \Delta \theta dr \), where \( G_i(r, \theta) \) is the performance generation function of server \( i \). Among these perturbations, only

\[
\Delta T(n, r, i) = c(n, r, i) \int T x(n, r) G_i(r, \theta) \Delta \theta dr
\]

will be finally realized by the system through propagation. That is, the perturbation generated at server \( i \) when the system is in states \((n, r^*)\), \( r^* < r + dr \), contributes \( \Delta T(n, r, i) \) to the perturbation of \( T \). Therefore, \( \Delta T \) should be the sum and integration of \( \Delta T(n, r, i) \). That is,

\[
\Delta T = \sum_{n^* \in X} \int G_i(r, \theta) c(n, r, i) x(n, r) dr.
\]

From this, we get

\[
\frac{1}{T} \frac{\Delta T}{\Delta \theta} = \sum_{n^* \in X} \int G_i(r, \theta) c(n, r, i) x(n, r) dr.
\]

Using (9) and (10) and letting \( \Delta \theta \rightarrow 0 \), we get

\[
\frac{1}{T} \frac{\Delta \eta}{\Delta \theta} = \sum_{n^* \in X} \int G_i(r, \theta) c(n, r, i) x(n, r) dr.
\]

This gives an intuitive explanation for the following theorem.

**Theorem 4:** For a closed queueing network with nonexponential service distributions, the normalized derivative of the steady-state throughput, \( 1 / \eta \frac{\partial \eta}{\partial \theta} \), equals the negative expected value of the product of the perturbation generation function and the realization

\[
\frac{1}{T} \frac{\Delta \eta}{\Delta \theta} = \sum_{n^* \in X} \int G_i(r, \theta) c(n, r, i) x(n, r) dr.
\]
probability. That is,
\[
\frac{1}{\eta} \frac{\partial \eta}{\partial \theta} = -E[G_1(r, \theta) c(n, r, \nu)]
\]
\[
= - \sum_{\mu \neq n} \int_{n}^{\infty} G_1(r, \theta) c(n, r, \nu) \pi(n, n, r) \, dr.
\]

A rigorous proof of this theorem is extremely lengthy and purely technical; it is the topic of a forthcoming paper [9].

VI. CONCLUSION

We have extended the realization theory to networks with nonexponential service time distributions. A perturbation generation function is defined; a set of equations specifying realization probabilities is derived; a formula for the normalized derivative of the system throughput is obtained. The results provide a new analytical method of calculating the sensitivity of the system throughput and justifies the application of the PA algorithms for networks with general service time distributions. The previous results about Jackson networks become special cases of the results in this note.

Another interesting work on PA of networks with nonexponential service time distributions is [10].

ACKNOWLEDGMENT

The author appreciates the reviewers' valuable comments which helped to improve the presentation of the note.

REFERENCES


On Choosing the Characterization for Smoothed Perturbation Analysis

Michael C. Fu and Jian-Qiang Hu

Abstract—Using second derivative estimators of the GI/G/1 queue as an illustrative example, we demonstrate that smoothed perturbation analysis estimators are not necessarily as "distribution-free" as in

finestesimal perturbation analysis estimators, in the sense that the appropriate choice of conditioning quantities—the so-called characterization—may depend on the underlying distribution. Through a different choice of characterization, we derive an estimator that works for distributions for which a previously derived estimator fails.

I. INTRODUCTION

The technique of using conditional expectation to derive perturbation analysis (PA) estimators was first introduced by Zazanis and Suri [10] to estimate second derivatives of mean steady-state system time in a GI/G/1 queue, where the technique of infinitesimal perturbation analysis (IPA) [7] gives consistent estimators for the first derivative [9] but not for the second derivative. The technique was subsequently generalized, formalized, and given the name Smoothed Perturbation Analysis (SPA) by Gong and Ho [6], who first pointed out that the crucial problem in applying this technique is choosing suitable conditioning quantities along the sample path, termed the characterization. In [10], Zazanis and Suri derive second derivatives estimators of mean steady-state system time in a GI/G/1 queue under one choice of characterization. However, we show that these estimators fail for the case of deterministic arrivals, but by choosing a different characterization, we derive an estimator which works for the case of deterministic arrivals and show via numerical results that the estimator is correct. Thus, for the same system and the same quantity of interest, we have two different SPA estimators, which have different domains of consistency and different variance properties. This example serves to highlight the importance of finding the "right" characterization in applying the technique of SPA to a particular problem. This notion of alternative characterizations yielding SPA estimators with different consistency properties is somewhat analogous in spirit to the notion of alternative representations of a stochastic process yielding different IPA estimators, pointed out by Glasserman [4], in the sense that just as the user of IPA must choose an appropriate representation for the stochastic process of interest in order for IPA to work correctly, the SPA user must also choose an appropriate characterization for the particular system and performance measure of interest; this being in addition to having to choose a representation, which the SPA user must also do. The other interesting and related point that can be gleaned from this example is that one must derive separate SPA second derivative estimators for the uniform and deterministic distributions, whereas the IPA first derivative estimators for uniform and deterministic distributions (w.r.t. mean) will always be the same (since the mean is a location parameter in both cases). Thus, in some sense, SPA is not as "distribution-free" as IPA. Furthermore, since a deterministic distribution can be viewed loosely as the limit of a certain sequence of uniform distributions, we show that the variance properties of the original SPA estimators degenerate as we take this "limit."

In closing this introduction, we should perhaps emphasize that although we have chosen second derivative estimation of GI/G/1 queues as our illustrative example, any of the examples found in [6] could also have been used to illustrate the points discussed in the previous paragraph. The rest of the note is organized as follows. In Section II, we review the original algorithm for estimating second derivatives of a GI/G/1 queue. In Section III, we discuss the conditions needed to prove consistency, based on previous work and some simulation experiments. Further simulation experiments demonstrate the degradation of the estimator for the limiting uniform distribution case, leading to the D/M/1 counterexample for which the original estimator completely fails. In Section IV, we