THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 62, No. 9, November 1983
Printed in U.S.A.

The Queueing Network Analyzer

By W. WHITT*

(Manuscript received March 11, 1983)

This paper describes the Queueing Network Analyzer (QNA), a software package developed at Bell Laboratories to calculate approximate congestion measures for a network of queues. The first version of QNA analyzes open networks of multiserver nodes with the first-come, first-served discipline and no capacity constraints. An important feature is that the external arrival processes need not be Poisson and the service-time distributions need not be exponential. Treating other kinds of variability is important. For example, with packet-switched communication networks we need to describe the congestion resulting from bursty traffic and the nearly constant service times of packets. The general approach in QNA is to approximately characterize the arrival processes by two or three parameters and then analyze the individual nodes separately. The first version of QNA uses two parameters to characterize the arrival processes and service times, one to describe the rate and the other to describe the variability. The nodes are then analyzed as standard GI/G/m queues partially characterized by the first two moments of the interarrival-time and service-time distributions. Congestion measures for the network as a whole are obtained by assuming as an approximation that the nodes are stochastically independent given the approximate flow parameters.

I. INTRODUCTION AND SUMMARY

Networks of queues have proven to be useful models to analyze the performance of complex systems such as computers, switching machines, communications networks, and production job shops.[1-7] To facilitate the analysis of these models, several software packages have

* Bell Laboratories.

been developed in recent years, e.g., BEST/1,[8] CADS,[9] PANACEA,[10–12] and one based on Heffes.[13] These software packages contain algorithms for Markov models that can be solved exactly. For some applications, the model assumptions are at least approximately satisfied, so that the analysis can be very helpful. For many other applications, however, the model assumptions are not even approximately satisfied, so that the analysis can be misleading.

A natural alternative to an exact analysis of an approximate model is an approximate analysis of a more exact model. This paper describes a software package called the Queueing Network Analyzer (QNA), which was recently developed at Bell Laboratories to calculate approximate congestion measures for networks of queues. QNA goes beyond existing exact methods by treating non-Markov networks: The arrival processes need not be Poisson and the service-time distributions need not be exponential. QNA treats other kinds of variability by approximately characterizing each arrival process and each service-time distribution with a variability parameter. It is also possible to analyze large networks quickly with QNA because the required calculations are minimal, the most complicated part being the solution of a system of linear equations. The current version of QNA is written in FORTRAN.

Here is a rough description of the model: There is a network of nodes and directed arcs. The nodes represent service facilities and the arcs represent flows of customers, jobs, or packets. There is also one external node, which is not a service facility, representing the outside world. Customers enter the network on directed arcs from the external node to the internal nodes, move from node to node along the internal directed arcs, and eventually leave the system on one of the directed arcs from an internal node to the external node. The flows of customers on the arcs are assumed to be random so that they can be represented as stochastic processes.

If all servers are busy at a node when a customer arrives, then the customer joins a queue and waits until a server is free. When there is a free server, that customer begins service, which is carried out without interruption. Successive service times at each node are assumed to be random variables, which may depend on the type of customer but which otherwise are independent of the history of the network and are mutually independent and identically distributed. After the customer completes service, he goes along some directed arc from that node to another node. The customer receives service in this way from several internal nodes and then eventually leaves the network. A picture of a typical network (without the external node) is given in Fig. 1.

An important feature of the model is that there may be flows from node j to node i, as well as flows from node i to node j. This is of

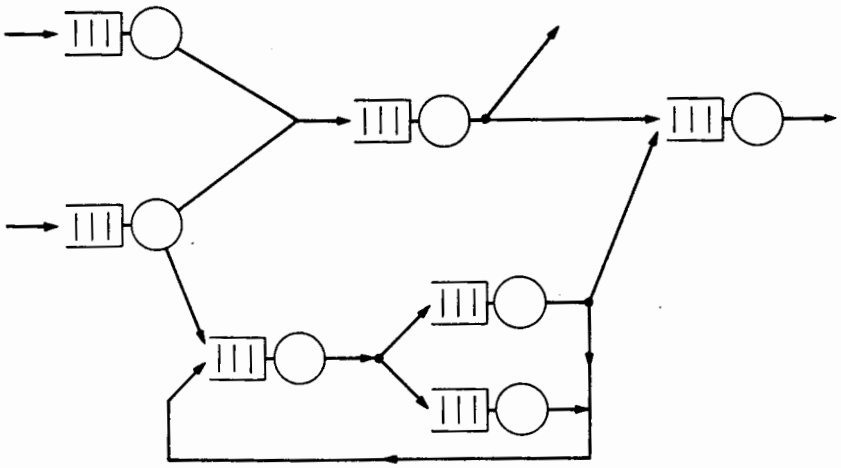

Fig. 1—An open network of queues.

course useful when customers can return to a node where they previously received service, but it is also useful when customers cannot return to a node where they previously received service. Then flows from node j to node i represent different customers than the customers that flow from node i to node j.

To be precise about the model, we give a list of basic assumptions. It is worth noting, however, that work is under way to extend QNA so that it can analyze systems in which each of the following assumptions is replaced by obvious alternatives. The general approximation technique is flexible, so that it is not difficult to modify and extend the algorithm.

Assumption 1. The network is *open* rather than closed. Customers come from outside, receive service at one or more nodes, and eventually leave the system.

Assumption 2. There are *no capacity constraints.* There is no limit on the number of customers that can be in the entire network and each service facility has unlimited waiting space.

Assumption 3. There can be *any number of servers* at each node. They are identical independent servers, each serving one customer at a time.

Assumption 4. Customers are selected for service at each facility according to the *first-come, first-served* discipline.

Assumption 5. There can be *any number of customer classes*, but customers cannot change classes. Moreover, much of the analysis in QNA is done for the aggregate or typical customer (see Sections 2.3 and VI).

Assumption 6. Customers can be created or combined at the nodes,

e.g., an arrival can cause more than one departure (see Section 2.2). (Think of messages.)

The general approach is to represent all the arrival processes and service-time distributions by a few parameters. The congestion at each facility is then described by approximate formulas that depend only on these parameters. The parameters for the internal flows are determined by applying an elementary calculus that transforms the parameters for each of the three basic network operations: superposition (merging), thinning (splitting), and flow through a queue (departure). These basic operations are depicted in Fig. 2. When the network is acyclic (e.g., queues in series), the basic transformations can be applied successively one at a time, but in general it is necessary to solve a system of equations or use an iterative method. To summarize, there are four key elements in this general approach:

1. *Parameters* characterizing the flows and nodes that will be readily available in applications and that have considerable descriptive power in approximations of the congestion at each node.
2. *Approximations for multiserver queues* based on the partial information provided by the parameters characterizing the arrival process and the service-time distribution at each node.
3. A *calculus for transforming the parameters* to represent the basic network operations: merging, splitting, and departure.
4. A *synthesis algorithm* to solve the system of equations resulting from the basic calculus applied to the network.

The current version of QNA uses two parameters to characterize the arrival processes and the service times, one to describe the rate and the other to describe the variability. (Three-parameter algorithms are being developed, however.) For the service times, the two param-

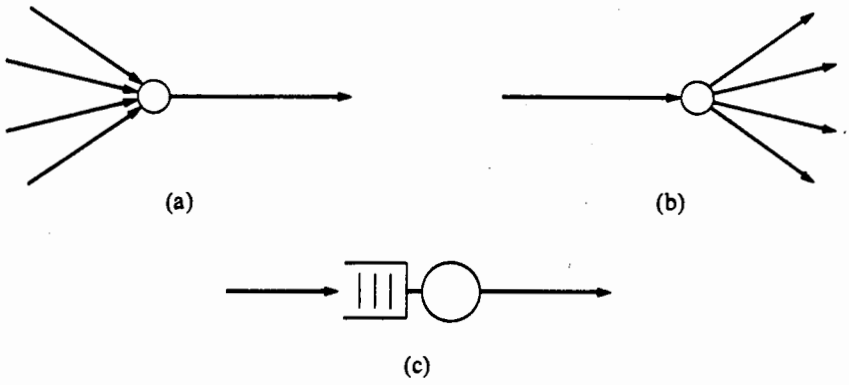

Fig. 2—Basic network operations: (a) Superposition or merging. (b) Decomposition or splitting. (c) Departure or flow through a queue.

eters are the first two moments. However, we actually work with the mean service time τ and the squared coefficient of variation c_s^2, which is the variance of the service time divided by the square of its mean. The user has the option of working with the service rate $\mu = \tau^{-1}$ instead of τ. For the arrival processes, the parameters are associated with renewal-process approximations. The first two parameters are equivalent to the first two moments of the renewal interval (interval between successive points) in the approximating renewal process. The equivalent parameters we use are the arrival rate λ, which is the reciprocal of the renewal-interval mean, and the squared coefficient of variation c_a^2, which is the variance of the renewal interval divided by the square of its mean.

We obtain the approximation of the flows by applying the general framework and the basic procedures for approximating point processes in Whitt,[14] incorporating refinements such as the hybrid procedures developed for merging by Albin.[15,16] Of course, the general idea of simple two-parameter approximations for stochastic point processes goes back at least to the equivalent random method for approximating overflow streams (see Wilkinson,[17] Cooper,[18] and references there). Renewal-process approximations for such point processes were introduced by Kuczura[19] (also see Rath and Sheng[20]). Two-parameter approximations for networks of queues similar to QNA have also been developed by others, apparently first by Reiser and Kobayashi[21] (also see Kuehn,[22] Sevcik et al.,[23] Chandy and Sauer,[24] Chapter 4 of Gelenbe and Mitrani,[4] and Shanthikumar and Buzacott[6]). These two-parameter approximations for networks of queues are also similar in spirit to two-parameter approximations for networks of blocking systems with alternate routing (see Katz[25]).

Some authors have referred to these two-parameter heuristic approximations for networks of queues as diffusion approximations,[4,21] but diffusion processes are not actually used. Diffusion approximations and associated heavy-traffic limit theorems have motivated some of the heuristic approximations in the literature and in QNA, and they are closely related to the asymptotic method for approximating point processes,[14] but the heuristic approximations in QNA are not the same as the more complicated diffusion approximations for networks of queues in Iglehart and Whitt,[26] Harrison and Reiman,[27] and Reiman.[28,29]

The approximation method in QNA is perhaps best described as a parametric-decomposition method,[22] because the nodes are analyzed separately after the parameters for the internal flows are determined. Moreover, when the congestion measures are calculated for the network as a whole, the nodes are treated (approximately) as being stochastically independent. This independence can be interpreted as

a generalization of the product-form solution that is valid for Markovian networks, i.e., in the Markov models the components of the vector representing the equilibrium number of customers at each node are stochastically independent, so that the probability mass functions for the vector is the product of the probability mass functions for the components. While QNA can be thought of as a decomposition method or an extended-product-form solution, an effort is made to capture the dependence among the nodes. The idea is to represent this dependence approximately through the internal flow parameters.

To see the motivation for QNA, consider the elementary open network containing a single node with a single server, an infinite waiting room, and the first-come, first-served discipline. Suppose there is a single customer class with each customer being served only once before departing. The standard Markov model of this elementary network, which is embodied in BEST/1, CADS, and PANACEA, is the classical M/M/1 queue,[1,3,18] which has a Poisson arrival process and an exponential service-time distribution. For the M/M/1 model, the expected waiting time EW (before the customer begins service) is

$$EW = \tau\rho/(1 - \rho), \tag{1}$$

where τ is the mean service time and ρ is the traffic intensity, which is assumed to satisfy $0 \leq \rho < 1$.

On the other hand, QNA uses an approximation for the GI/G/1 model to represent this one-node network. The GI/G/1 model has a renewal arrival process and both the interarrival-time distribution and the service-time distribution are general. In QNA, the arrival process is represented by a renewal process partially characterized by two parameters: the arrival rate λ and the variability parameter c_a^2. The service-time distribution is also partially characterized by two parameters: the mean service time τ and the variability parameter c_s^2. In contrast to (1), the formula for the expected waiting time in QNA is

$$EW = \tau\rho(c_a^2 + c_s^2)g/2(1 - \rho), \tag{2}$$

where $g \equiv g(\rho, c_a^2, c_s^2)$ is either one (when $c_a^2 \geq 1$) or less than one (when $c_a^2 < 1$); see (45). When $g(\rho, c_a^2, c_s^2) = 1$, (2) differs from (1) by the factor $(c_a^2 + c_s^2)/2$. When the arrival process is Poisson, $c_a^2 = 1$; when the service-time distribution is exponential, $c_s^2 = 1$. Hence, if the GI/G/1 model is actually an M/M/1 model, (2) reduces to (1). Of course, the user of QNA can set $c_a^2 = c_s^2 = 1$ and obtain (1). In fact, the values $c_a^2 = 1$ and $c_s^2 = 1$ are default values that the program uses if the user does not have variability parameters to provide. Each c^2 can assume any nonnegative value: $c^2 = 0$ for the degenerate deterministic distribution; $c^2 = k^{-1}$ for an erlang E_k, the sum of k i.i.d. exponential random variables; and $c^2 > 1$ for mixtures of exponential

distributions. Obviously, the difference between (2) and (1) can be large, so that (2) often significantly reduces the error.

To obtain (2), we studied the GI/G/1 queue partially characterized by the moments of the interarrival-time and service-time distributions. Building on previous work by Holtzman,[30] Rolski,[31] and Eckberg,[32] we investigated the set of possible values of EW given the partial information.[33-37] When $c_s^2 \geq 1$, formula (2) is always a possible value, i.e., there always is a GI/G/1 system with interarrival-time and service-time distributions having the specified moments in which (2) is correct. In general, (2) appears to be a reasonably typical value.

For the single-node example just considered, the arrival process was a renewal process. More generally, it is natural to think of all the non-Poisson arrival processes in the model as renewal processes, either because they are initially renewal processes or because the algorithm can be interpreted as approximating general arrival processes by renewal processes. Hence, with one customer class, it is natural to think of the model as a generalization of the open Jackson network M/M/m queues to an open Jackson network of GI/G/m queues. Each node is approximated by a GI/G/m queue having a renewal arrival process independent of service times that are independent and identically distributed with a general distribution. It is significant that QNA is consistent with the Jackson network theory: If there is a single class of customers, if all the arrival processes are Poisson, and if all the service-time distributions are exponential, then QNA is exact. However, for the general model few analytical results are available, so approximations are needed.

The software package QNA has a flexible input procedure: the model will accept more than one kind of input (see Section II). For the standard input, only limited information is required. Only two parameters are needed for each service-time distribution and each external arrival process. Also, a routing matrix is needed, which gives the proportion of those customers completing service at facility i that go next to facility j. (The algorithm is based on Markovian routing.) Hence, for n nodes, the input consists of $n^2 + 4n$ numbers.

There is also an alternate input by classes and routes. In this scheme there are different classes of customers and each class enters the network at a fixed node and passes through a specified sequence of nodes. For each class, there are two parameters characterizing its external arrival process and two parameters characterizing the service-time distribution at each node on its route. With this input by routes, different classes can have different service-time distributions at a given node and the same class can have different service-time distributions during different visits to the same node. For a class with n nodes on its route, the input consists of $3n + 2$ numbers. (This includes

the n nodes on the route.) QNA analyzes this route input by aggregation: All the classes are aggregated by QNA to convert the route input into the standard input. Afterwards, the special parameters of each class are used to describe its sojourn times.

QNA also provides a fairly rich output. Several different congestion measures are calculated for each node: the traffic intensity (utilization), the expected number of busy servers (offered load), and the mean and variance of the equilibrium delay and number of customers present. In fact, for single-server nodes the delay distribution itself is described. Congestion measures for the entire network are also calculated, under the approximation assumption that the nodes are stochastically independent given the approximate flow parameters. Means and variances of total service times, total delays, and total sojourn times (response times) are given. When the input is by routes, these characteristics are given for each customer class. Otherwise, these characteristics are given for any route requested by the user.

A desirable feature of QNA is the structure of the calculus to transform the parameters to characterize the internal flows. The calculus is linear for each network operation, so that the parameters for the internal flows are determined simply by solving systems of linear equations. For the rates, the system of linear equations is just the familiar traffic rate equations occurring in the Jackson network of M/M/m queues. After having obtained the rates, we obtain the variability parameters of the internal flows (the squared coefficients of variations) by solving another system of linear equations. As a by-product, the existence of a unique nonnegative solution for the flow parameters is trivially guaranteed. There is no guarantee that an iterative scheme will converge, and if it does, there is typically no guarantee that a solution is unique. The linearity also guarantees that the computation required is not great. Since there is only one linear equation per node in the network, QNA can be used to analyze large networks repeatedly at minimal cost.

The linear calculus for transforming the variability parameters incorporates results of recent studies to improve the accuracy of the approximations. The general framework for approximating point processes in Whitt[14] is used. Significant improvement over previous approximation methods of this kind has been obtained by paying particular attention to the difficult superposition operation. For superposition, we use a modification of the hybrid procedure developed at Bell Laboratories by Albin.[15,16,38,39]

We emphasize that QNA is approximate. In applications it is important to validate the QNA output by comparing it with simulations and/or measurements. QNA is designed so that it is easy to incorporate improvements and it is easy to tune QNA for particular applications.

QNA also provides a useful framework for developing new approximation procedures. Moreover, it is easy to use QNA in conjunction with other special algorithms available to analyze the nodes or the flows.

The rest of this paper describes QNA in more detail. The paper is organized—just as the output is—according to the main steps in the analysis. The input is described in Section II. Section III describes the preliminary analysis to eliminate immediate feedback. The procedures to determine the internal flow parameters are contained in Section IV, and the procedures to calculate approximate congestion measures for the nodes are contained in Section V. Section VI contains the procedures to calculate approximate congestion measures for the network as a whole.

In a sequel in this issue of the *Journal*,[40] we describe the performance of QNA by comparing it with simulation and other approximations of several networks of queues. The sequel illustrates how to apply QNA and demonstrates the importance of the variability parameters.

II. THE INPUT

In this section we describe the input options currently available for QNA. We anticipate more input options in the future. In Section 2.1 we describe the standard input, which is relatively compact. In Section 2.2 we describe a minor modification of the standard input, which allows for the creation or combination of customers at the nodes. For example, when a packet completes service at some node, it may cause several packets to be sent to other nodes. In Section 2.3 we describe an alternate input for different classes of customers having specified routes. We also describe the way QNA converts this input by classes and routes into the standard input of Section 2.1.

2.1 The standard input

With the standard input, there is a single customer class and no creation of customers at the nodes. Any number of networks can be processed during a single run, so the user first specifies the number of networks. Then, for each network, the user specifies the number of nodes, and for each node the number of servers. For each node in the network, there are two parameters for the service-time distribution and two parameters for the external arrival process. Finally, there is a routing matrix, indicating the proportion of customers that go to node j from node i. Here is a list of the input data for each network with the notation we use:

n = number of (internal) nodes in the network
m_j = number of servers at node j
λ_{0j} = external arrival rate to node j

c_{0j}^2 = variability parameter of the external arrival process to node j (squared coefficient of variation of the renewal interval in the approximating renewal process)

τ_j = mean service time at node j

c_{sj}^2 = squared coefficient of variation of the service-time distribution at node j

q_{ij} = proportion of those customers completing service at node i that go next to node j.

In matrix notation, $Q \equiv (q_{ij})$ is an $n \times n$ matrix and $\Lambda_0 \equiv (\lambda_{0j})$ is an $1 \times n$ vector. The user has the option of inputting τ_j or its reciprocal μ_j, the service rate at node j. (The same form must be used for all nodes.)

The user need not specify the variability parameters c_{0j}^2 and c_{sj}^2, in which case they are set equal to the default value one, corresponding to the M/M/1 model having a Poisson arrival proccess and an exponential service-time distribution. (Again, this option applies to all nodes.) Alternatively, the user can specify only the service-time variability parameters, c_{sj}^2, in which case either all the arrival-process variability parameters are automatically set equal to 1, yielding an M/G/1 approximation for each node, or the QNA algorithm is applied.

2.2 Creating and combining customers

QNA has an option to allow creating or combining customers at the nodes following the completion of service. For example, a message processed at some node might cause messages to be sent to several other nodes. Alternatively, messages might be divided into packets after service at one node and then later recombined into messages after service at another node. In a job shop, the focus might shift back and forth between units and lots, e.g., at different nodes we might consider bottles, six-packs, cases, and even truckloads.

With this option, the user must specify the multiplicative factor γ_j of customer creation or combination at node j for each j. There is customer creation (combination) at node j if $\gamma_j > 1$ ($\gamma_j < 1$). If customers are neither created nor combined, then $\gamma_j = 1$. If λ_j is the overall arrival rate to node j, then the departure rate, after this modification, is $\lambda_j\gamma_j$ and the rate of departure from the network j is

$$\lambda_j\gamma_j\left(1 - \sum_{k=1}^{n} q_{jk}\right).$$

When artificial nodes are used, the creation or combination can also be placed before service.

To obtain our approximation formulas, we work with the following models of customer creation and combination. These models require integer values, but the approximation formulas and the QNA input do

not. For customer creation, we replace each departure from node j with a batch of size γ_j. For customer combination, we replace γ_j^{-1} successive interdeparture intervals by a single one. From these models it is not difficult to calculate the impact of γ_j, e.g., the departure rate at node j is simply multiplied by γ_j.

2.3 Input by classes and routes

QNA provides the option of defining different customer classes. Each class has its own route or itinerary that specifies the sequence of nodes visited. Thus, for each class the routing is deterministic. Each class has an external arrival process that goes to the first node on the route. As usual, the external arrival process is characterized by rate and variability parameters. Also, each class may have its own service-time distribution at each node on its route. The service-time distributions can be different, not only for different classes, but also for different visits to the same node by the same class. These service-time distributions are also characterized by rate and variability parameters. (Alternatively, the user can elect to input the service-time parameters for each node. Then all classes have the same service-time distribution at each visit to a particular node.)

As with the standard input, the user must specify the number of nodes and the number of servers at each node. Now we need the number of routes too. The required data are:

n = number of nodes
m_j = number of servers at node j
r = number of routes.

Here is a list of the input data for the kth customer class of a network:

n_k = number of nodes on route k
$\hat{\lambda}_k$ = external arrival rate of class k
c_k^2 = variability parameter of the external arrival process for class k
n_{kj} = the jth node visited by customer class k
τ_{kj} = the mean service time of class k at the jth node of its route
c_{skj}^2 = the variability parameter of the service-time distribution of class k at the jth node of its route.

QNA converts this input by classes and routes into the standard input in Section 2.1. It then calculates the parameters of a typical or aggregate customer. Later, when computing sojourn times or response times of each customer class, QNA uses the service-time parameters of that customer class. The first version of QNA assumes as an approximation that each customer sees independent versions of the equilibrium distribution at each node. Hence, the waiting time before beginning service at each node is assumed to be the same for all classes and all visits.

We now indicate how QNA converts the input by classes and routes into the standard input of Section 2.1. For this purpose, let $1H$ be the indicator function of the set H, i.e., $1H(x) = 1$ if $x \in H$ and $1H(x) = 0$ otherwise.

First, we obtain the external arrival rates by

$$\lambda_{0j} = \sum_{k=1}^{r} \hat{\lambda}_k 1\{k{:}n_{k1} = j\}, \tag{3}$$

i.e., the external arrival rate at node j, λ_{0j}, is the sum of all route arrival rates $\hat{\lambda}_k$ for which the first node on the route is j. (Here the $1H$ notation is used for $H = \{k{:}n_{k1} = j\}$.) Similarly, the flow rate from i to j is

$$\lambda_{ij} = \sum_{k=1}^{r} \sum_{\ell=1}^{n_k-1} \hat{\lambda}_k 1\{(k, \ell){:}n_{k\ell} = i,\ n_{k,\ell+1} = j\} \tag{4}$$

and the flow from i out of the network is

$$\lambda_{i0} = \sum_{k=1}^{r} \hat{\lambda}_k 1\{k{:}n_{kn_k} = i\}. \tag{5}$$

From (4) and (5), we obtain the routing matrix Q. The proportion of customers that go to j from i is

$$q_{ij} = \lambda_{ij} \Big/ \left(\lambda_{i0} + \sum_{k=1}^{n} \lambda_{ik}\right). \tag{6}$$

If node i is an active part of the network, then the denominator will be strictly positive. Otherwise, QNA gives an error message.

Next, if the service-time parameters are given by routes, we obtain the service-time parameters for the nodes by averaging:

$$\tau_j = \frac{\sum_{k=1}^{r} \sum_{\ell=1}^{n_k} \hat{\lambda}_k \tau_{k\ell} 1\{(k, \ell){:}n_{k\ell} = j\}}{\sum_{k=1}^{r} \sum_{\ell=1}^{n_k} \hat{\lambda}_k 1\{(k, \ell){:}n_{k\ell} = j\}}. \tag{7}$$

The denominator in (7) will be strictly positive if node j is ever visited. Otherwise, as with (6), QNA supplies an error message.

We obtain the node variability parameters c_{aj}^2 using the property that the second moment of a mixture of distributions is the mixture of the second moments. Therefore, we have

$$\tau_j^2(c_{aj}^2 + 1) = \frac{\sum_{k=1}^{r} \sum_{\ell=1}^{n_k} \hat{\lambda}_k \tau_{k\ell}^2 (c_{sk\ell}^2 + 1) 1\{(k, \ell){:}n_{k\ell} = j\}}{\sum_{k=1}^{r} \sum_{\ell=1}^{n_k} \hat{\lambda}_k 1\{(k, \ell){:}n_{k\ell} = j\}}. \tag{8}$$

At this point, QNA has calculated enough information about the

standard input to compute the internal flow rates λ_j and the traffic intensities ρ_j as described in Section 4.1, i.e.,

$$\rho_j = \lambda_j \tau_j / m_j. \tag{9}$$

QNA uses this information to calculate the variability parameters c_{0j}^2 of the external arrival process. The hybrid approximation for superposition arrival processes in Section 4.2 is also used here because the external arrival process to node j is the superposition of the external arrival processes to node j from the different classes. If $\lambda_{0j} = 0$, then c_{0j}^2 does not matter and QNA sets $c_{0j}^2 = 1$. Otherwise,

$$c_{0j}^2 = (1 - \bar{w}_j) + \bar{w}_j \left[\sum_{k=1}^{r} c_k^2 \left(\hat{\lambda}_k 1\{k{:}n_{k1} = j\} \Big/ \sum_{\ell=1}^{r} \hat{\lambda}_\ell 1\{k{:}n_{k1} = j\} \right) \right], \tag{10}$$

where

$$\bar{w}_j \equiv \bar{w}_j(\rho_j, \bar{\nu}_j) = [1 + 4(1 - \rho_j)^2(\bar{\nu}_j - 1)]^{-1}, \tag{11}$$

ρ_j is the traffic intensity in (9), and

$$\bar{\nu}_j = \left[\sum_{k=1}^{r} \left(\hat{\lambda}_k 1\{k{:}n_{k1} = j\} \Big/ \sum_{\ell=1}^{r} \hat{\lambda}_\ell 1\{\ell{:}n_{\ell 1} = j\} \right)^2 \right]^{-1}. \tag{12}$$

Example 1: To help fix the ideas, we consider an elementary example with $n = 2$ nodes and $r = 3$ routes. Let the number of servers at the nodes be $m_1 = 40$ and $m_2 = 10$. Let the route input be described by vectors

$$(n_k, \hat{\lambda}_k, c_k^2; n_{k1}, \tau_{k1}, c_{sk1}^2; \cdots; n_{kn_k}, \tau_{kn_k}, c_{skn_k}^2). \tag{13}$$

Here suppose that the r vectors are:

$$\begin{aligned} &(2, 2, 1; 1, 1, 1; 1, 3, 3) \\ &(3, 3, 2; 1, 2, 0; 2, 1, 1; 1, 2, 1) \\ &(2, 2, 4; 2, 1, 1; 1, 2, 1). \end{aligned} \tag{14}$$

The first route corresponds to a Poisson arrival process at rate 2 to node 1, with all customers being fed back immediately for a second service before departing from the network. (Of course, the arrival process need not actually be Poisson; a Poisson process always has $c^2 = 1$ but other processes could have $c^2 = 1$ too.) The second class also enters at node 1, then goes to node 2 and back to node 1 before departing from the network, etc.

By (3), the external arrival rates are $\lambda_{01} = 5$ and $\lambda_{02} = 2$. By (4), the internal flow rates are $\lambda_{11} = 2$, $\lambda_{12} = 3$, $\lambda_{21} = 5$, and $\lambda_{22} = 0$. By (5), the flow rates out of the network are $\lambda_{10} = 7$ and $\lambda_{20} = 0$. By (6), the routing probabilities are: $q_{11} = 1/6$, $q_{12} = 1/4$, $q_{21} = 1$, and $q_{22} = 0$. By

(7), the mean service times are $\tau_1 = 2$ and $\tau_2 = 1$. By (8), $c_{s1}^2 = 1.67$ and $c_{s2}^2 = 1.00$. Note that both service times at node 2 in (14) have mean 1 and squared coefficient of variation 1, as with a common exponential distribution, so we should want $\tau_2 = c_{s2}^2 = 1$.

To obtain the internal arrival rates, we solve the traffic rate equations as in Section 4.1, i.e.,

$$\lambda_j = \lambda_{0j} + \sum_{i=1}^{n} \lambda_i q_{ij}, \tag{15}$$

to obtain $\lambda_1 = 12$, $\lambda_2 = 5$, $\rho_1 = 0.6$, and $\rho_2 = 0.5$.

Finally we obtain the variability parameters c_{0j}^2. First, from (12), $\bar{\nu}_1 = 25/13$ and $\bar{\nu}_2 = 1.0$. Then, from (11), $\bar{w}_1 = 0.629$ and $\bar{w}_2 = 1$, so that $c_{01}^2 = 1.38$ and $c_{02}^2 = 4$. Since there is only one external arrival process to node 2, we should have $\bar{\nu}_2 = \bar{w}_2 = 1$ and $c_{02}^2 = c_3^2 = 4$.

III. ELIMINATING IMMEDIATE FEEDBACK

In this section we describe a function that QNA can perform before calculating the internal flow parameters and analyzing the congestion. The user can elect to reconfigure the network to eliminate immediate feedback. This procedure, which was originally suggested by Kuehn,[22] usually improves the quality of approximations. Hence, it is recommended and is performed in the standard version of QNA.

Immediate feedback occurs whenever $q_{ii} > 0$. Since QNA assumes Markovian routing, each customer completing service at node i is immediately fed back to node i to be served again with probability q_{ii}. Each time the customer goes to the end of the line. With the decomposition method, QNA assumes the customer finds the equilibrium number of customers at the node each time, with each visit being an independent experiment.

QNA eliminates immediate feedback by giving each customer, upon arrival from another node, his or her total service time before going to a different node. This is equivalent to putting a customer immediately fed back at the head of the line instead of at the end of the line. Transitions from node i back to node i are eliminated and the new probability of a transition to node j becomes the old conditional probability given that the customer departs from node i. In other words, each visit to node i from elsewhere plus all subsequent times immediately fed back are interpreted as a single visit. The service time is increased to compensate.

The motivation for this procedure is easy to explain. For a multiserver node with Bernoulli (Markovian) feedback and iid service times that are independent of a general arrival process (not necessarily Poisson or renewal), the distribution of the queue length process (but not the waiting times) is the same after this transformation. Hence,

we calculate the approximate values of the mean and variance of the equilibrium queue length for the transformed node without feedback and use them to derive approximate waiting time characteristics. By Little's formula,[41,42] the expected waiting time is also exact, i.e., the only error is in the approximation of the arrival process by a renewal process and the approximations for the characteristics of the GI/G/m queue; there is no additional error due to the immediate feedback.

The first step of the reconfiguring procedure is quite simple: the new service time is regarded as a geometric mixture of the n-fold convolution of the old service-time distribution. The parameters τ_i, c_{si}^2, and q_{ij} are changed to $\hat{\tau}_i$, $\hat{c}_{si}^2$, and $\hat{q}_{ij}$ when $q_{ii} > 0$:

$$\begin{aligned}
\hat{\tau}_i &= \tau_i/(1 - q_{ii}) \\
\hat{c}_{si}^2 &= q_{ii} + (1 - q_{ii})c_{si}^2 \\
\hat{q}_{ii} &= 0 \\
\hat{q}_{ij} &= q_{ij}/(1 - q_{ii}), \qquad j \neq i.
\end{aligned} \tag{16}$$

Afterwards, when calculating congestion measures for node i, QNA makes further adjustments. When we eliminate immediate feedback according to (16), we no longer count the times a customer is fed back immediately as separate visits. Hence, we need to adjust the congestion measures that are expressed per visit. For example, since the expected number of visits to node i per visit from outside is $(1 - q_{ii})^{-1}$, to obtain the expected waiting time per original visit to node i, we multiply the values of the expected waiting time EW_i obtained from (16) by $(1 - q_{ii})$. Of course, the number of customers at each node is not affected by the feedback treatment.

Let $\bar{\lambda}_i$, $\bar{\tau}_i$, etc., represent the new adjusted values. In terms of the parameters λ_i, τ_i, etc., obtained using (16), the new adjusted values are:

$$\begin{aligned}
\bar{\lambda}_i &= \lambda_i/(1 - q_{ii}) \\
\bar{\tau}_i &= (1 - q_{ii})\tau_i \\
c_{si}^2 &= (c_{si}^2 - q_{ii})/(1 - q_{ii}) \\
E\bar{W}_i &= (1 - q_{ii})EW_i \\
\mathrm{Var}(\bar{W}_i) &= (1 - q_{ii})\mathrm{Var}(T_i') - \bar{c}_{si}^2\bar{\tau}_i^2 \\
\mathrm{Var}(T_i') &= c^2(T_i')(EW_i + \tau_i)^2 \\
c^2(T_i') &= c^2(\bar{T}_i')(1 + q_{ii}) + q_{ii} \\
c^2(\bar{T}_i') &= (\mathrm{Var}\ \bar{W}_i' + \bar{c}_{si}^2\bar{\tau}_i^2)(E\bar{W}_i + \bar{\tau}_i)^{-2} \\
\mathrm{Var}(\bar{W}_i') &= EN_i\bar{c}_{si}^2\bar{\tau}_i^2 + c^2(N_i)(EN_i)^2\bar{\tau}_i^2,
\end{aligned} \tag{17}$$

where N_i represents the equilibrium number of customers at node i and the T_i variables represent the sojourn time per visit at node i.

We obtain the variables $\bar{\tau}_i$ and $\bar{c}_{si}^2$ in (17) by inverting the operation in (16), so we receive the original data again. The last five formulas in (17) involving the second-moment characteristics of $\bar{W}_i$ are based on the results of N_i in the transformed system and heavy-traffic limit theorems for networks of queues by Reiman.[28,29] The main quantity desired is $\mathrm{Var}(\bar{W}_i)$; the variable $\bar{W}_i'$ is a preliminary approximation for $\bar{W}_i$.

In heavy traffic, the changes in the queue length at the nodes are negligible during a customer's sojourn in the network. Hence, if node i is visited X_i times by some customer, then the total sojourn time at node i, say T_i', is distributed approximately (in heavy traffic) as $X_i\bar{T}_i'$, where X_i is independent of $\bar{T}_i'$ and $\bar{T}_i'$ is the sojourn time per individual visit in (17). (We use T_i' and $\bar{T}_i'$ instead of T_i and $\bar{T}_i$ because we do not use the description of T_i obtained directly from (16) and $\bar{T}_i$ will differ from $\bar{T}_i'$.) By the independence, $ET_i'^2 = EX_i^2E\bar{T}_i'^2$. Since X_i is geometrically distributed with mean $(1 - q_{ii})^{-1}$, $c^2(X_i) = q_{ii}$, and we obtain the seventh formula in (17).

The sixth and eighth formulas in (17) just express the formula for c^2 in terms of the mean and variance and the fact that the sojourn time is the sum of a waiting time and a service time. The final formula for $\mathrm{Var}(W_i')$ is obtained by approximating W_i' by the sum of N_i iid service times, using standard formulas for the variance of a random sum (e.g., compute $EW_i'^2$ by first conditioning on N_i). Finally, we obtain the fifth formula for $\mathrm{Var}(\bar{W}_i)$ by splitting the variance of T_i' into waiting-time and service-time components and dividing by the expected number of visits to node i. As a consequence, $\mathrm{Var}(T_i')$ seems more reliable than $\mathrm{Var}(\bar{W}_i)$. This procedure makes $\mathrm{Var}(T_i)$, computed from $\mathrm{Var}(\bar{W}_i)$ by adding variance components as in Section VI, agree with the direct formula for $\mathrm{Var}(T_i')$ in (17).

The congestion measures based on (16) can be used to describe the total delays and total sojourn times of arbitrary customers in the network as in Section 6.2, but the congestion measures based on (17) are needed to describe the behavior of particular customers with specified routes as in Section 6.3. However, as stated above, $\mathrm{Var}(T_i')$ in (17) is an attractive alternative to $\mathrm{Var}(T_i)$ obtained via (16).

Experience indicates that eliminating immediate feedback often yields a better approximation (see Kuehn[22] and Sections V and VII of Whitt[40]). It is also often desirable to reconfigure the network to eliminate almost-immediate feedback, e.g., flows that return relatively quickly after passing through one or more other nodes (see Section V of Whitt[40]). Further study is needed to understand feedback phenomena and to develop improved approximations.

IV. THE INTERNAL FLOW PARAMETERS

In this section we indicate how QNA calculates the internal flow parameters. In Section 4.1 we focus on the flow rates, which are obtained via the traffic rate equations, just as with the Markov models. In Section 4.2 we display the corresponding system of linear equations yielding the variability parameters. The remaining subsections explain how the variability parameter equations were obtained. The basic operations of superposition, splitting, and departure are discussed in Section 4.3, Section 4.4, and Section 4.5, and their synthesis in Section 4.6.

4.1 Traffic-rate equations

In this step QNA calculates the total arrival rate to each node. Let λ_j be the total arrival rate to node j, let γ_j be the multiplicative factor of customer creation at node j as specified in Section 2.2, and let δ_j be the departure rate (to other nodes as well as out of the network) at node j. In general, $\delta_j = \lambda_j\gamma_j$. If there is no customer creation, then $\gamma_j = 1$ and the rate in equals the rate out.

The fundamental *traffic-rate equations* are just

$$\lambda_j = \lambda_{0j} + \sum_{i=1}^{n} \lambda_i\gamma_i q_{ij} \tag{18}$$

for $j = 1, 2, \cdots, n$, or in matrix notation

$$\Lambda = \Lambda_0(I - \Gamma Q)^{-1}, \tag{19}$$

where $\Lambda_0 \equiv (\lambda_{0j})$ is the external arrival-rate vector, $Q \equiv (q_{ij})$ is the routing matrix, and $\Gamma = (\gamma_{ij})$ is the diagonal matrix with $\gamma_{ii} = \gamma_i$ and $\gamma_{ij} = 0$ for $i \neq j$. When there is no customer creation, $\gamma_i = 1$ and $\Gamma = I$. Of course, (18) is just a system of linear equations. To solve them is equivalent to inverting the matrix $(I - \Gamma Q)$ in (19). When customers can be created at the nodes as in Section 2.2, special care should be taken to be sure that (18) has a solution. We need to have $sp(\Gamma Q) < 1$ where $sp(\Gamma Q)$ is the spectral radius of ΓQ.

Given the arrival rates, it is possible to solve for the *traffic intensities* or *utilizations* at each node, defined by

$$\rho_i = \lambda_i\tau_i/m_i, \qquad 1 \leqslant i \leqslant n. \tag{20}$$

If $\rho_i \geqslant 1$, then the ith node is *unstable*. If any node is unstable, the algorithm gives an error message, prints out the traffic intensities, and stops. The associated *offered load* at node i, which coincides with the expected number of busy servers [see p. 400 of Heyman and Sobel[41] or (4.2.3) of Franken et al.[42]] is

$$\alpha_i = \lambda_i\tau_i, \qquad 1 \leqslant i \leqslant n. \tag{21}$$

The parameters α_i and ρ_i coincide for a single server, with α_i tending to be more useful as the number of servers, m_i, increases (obviously when $m_i = \infty$).

After the arrival rates have been calculated for the nodes, QNA calculates related quantities for the arcs:

$$\begin{aligned} \lambda_{ij} &= \lambda_i \gamma_i q_{ij} && \text{—the arrival rate to node j from node i} \\ p_{ij} &= \lambda_{ij}/\lambda_j && \text{—the proportion of arrivals to j that came from } i,\ i \geqslant 0. \end{aligned} \tag{22}$$

Similarly, QNA calculates the following output rates:

$$\begin{aligned} d_i &= \lambda_i \gamma_i \left(1 - \sum_{j=1}^{n} q_{ij}\right) && \text{—the departure rate out of the network from node i} \\ d &= \sum_{i=1}^{n} d_i — && \text{the total departure rate out of the network.} \end{aligned} \tag{23}$$

4.2 Traffic variability equations

The heart of the approximation is the system of equations yielding the variability parameters for the internal flows, i.e., the squared coefficients of variation for the arrival processes, c_{aj}^2. (These are derived in Sections 4.3 through 4.7.) The equations are linear, of the form

$$c_{aj}^2 = a_j + \sum_{i=1}^{n} c_{ai}^2 b_{ij}, \qquad 1 \leqslant j \leqslant n, \tag{24}$$

where a_j and b_{ij} are constants, depending on the input data:

$$a_j = 1 + w_j \left\{ (p_{0j} c_{0j}^2 - 1) + \sum_{i=1}^{n} p_{ij}[(1 - q_{ij}) + (1 - \nu_{ij})\gamma_i q_{ij} \rho_i^2 x_i] \right\} \tag{25}$$

and

$$b_{ij} = w_j p_{ij} q_{ij} \gamma_i [\nu_{ij} + (1 - \nu_{ij})(1 - \rho_i^2)], \tag{26}$$

where x_i, ν_{ij}, and w_j depend on the basic data determined previously, e.g., ρ_i, m_i and c_{ai}^2, but not on the variability parameters c_{aj}^2 being calculated. The parameter γ_i is the multiplicative factor of customer creation or combination, introduced in Section 2.2. The variables x_i and ν_{ij} are used to specify the departure operation; the variable w_j is used to specify the superposition operation. The variables ν_{ij} and w_j are weights or probabilities that are used in convex combinations

arising in hybrid approximations for departure and superpositions, respectively. The variables x_j, ν_{ij}, and w_j are included to make modification of the algorithm based on (24) easy. The specific values in this version of QNA are:

$$x_i = 1 + m_i^{-0.5}(\max\{c_{si}^2, 0.2\} - 1), \tag{27}$$

$$\nu_{ij} = 0, \tag{28}$$

and

$$w_j = [1 + 4(1 - \rho_j)^2(\nu_j - 1)]^{-1} \tag{29}$$

with

$$\nu_j = \left[\sum_{i=0}^{n} p_{ij}^2\right]^{-1} \tag{30}$$

and p_{ij} in (22).

It is significant that it is easy to modify this system of equations. For example, other hybrid procedures for departures or superpositions can be introduced just by changing ν_{ij} and w_j, respectively. In this way, it is easy to calculate and compare the variability parameters for several different approximation procedures.

4.3 Superposition

The purpose of the following sections is to explain the key approximation equations (24) through (30), which yield the variability parameters for the internal flows. The approximations are all based on the basic methods in Whitt:[14] the asymptotic method and the stationary-interval method. We consider the basic operations—superposition, splitting, and departure—in turn, and then their synthesis.

For superposition, the stationary-interval method is nonlinear so it presents difficulties.[14–16,22] Moreover, there appears to be no natural modification that makes it linear. On the other hand, the asymptotic method is linear. By the asymptotic method, the superposition squared coefficient of variation c_A^2 as a function of component squared coefficients of variation c_i^2 and the rates λ_i is just the convex combination

$$c_A^2 = \sum_i \left(\lambda_i \Big/ \sum_k \lambda_k\right) c_i^2. \tag{31}$$

However, neither the asymptotic method nor the stationary-interval method alone works very well over a wide range of cases, e.g., see Section III of Whitt.[40] Albin[15,16] found that considerable improvement could be obtained by using a refined composite procedure, which is based on a convex combination of c_A^2 for the asymptotic method and c_{SI}^2 for the stationary-interval method. Her hybrid c_H^2 is of the form

$$c_H^2 = wc_A^2 + (1 - w)c_{SI}^2. \tag{32}$$

Unfortunately, since c_{SI}^2 is nonlinear, so is c_H^2. However, Albin found that a convex combination of c_A^2 and the exponential c^2 of 1 worked almost as well, having 4-percent average absolute error as opposed to 3 percent. Hence, we use such a hybrid procedure, namely,

$$\begin{aligned} c_H^2 &= wc_A^2 + (1 - w) \\ &= w \sum_i \left(\lambda_i \Big/ \sum_k \lambda_k \right) c_i^2 + 1 - w, \end{aligned} \tag{33}$$

where w is a function of ρ and the rates. Extensive simulation prompted Albin to suggest the weighting function

$$w = [1 + 2.1(1 - \rho)^{1.8}\nu]^{-1}, \tag{34}$$

where

$$\nu = \left[\sum_i \left(\lambda_i \Big/ \sum_k \lambda_k \right)^2 \right]^{-1}. \tag{35}$$

Note that if there are k component processes with equal rates then $\nu = k$. The parameter ν can be thought of as the number of component streams, with it being an equivalent number if the rates are unequal.

However, the weighting function (58) fails to satisfy an important consistency condition: We should have $w = 1$ when $\nu = 1$; if there is a single arrival process, the superposition operation should leave the c^2 parameter unchanged. Moreover, new theoretical results[39] indicate that the exponent of $(1 - \rho)$ in (34) should be 2. Hence, we use formula (33) based on the weight function w in (29).

4.4 Splitting

No approximation is needed for splitting because a renewal process that is split by independent probabilities (Markovian routing) is again a renewal process. However, approximation is of course indirectly associated with this step because the real process being split is typically not a renewal process and the splitting is often not according to Markovian routing.

Since a renewal process split according to Markovian routing is a renewal process, the asymptotic method and the stationary-interval method coincide. If a stream with a parameter c^2 is split into k streams, with each being selected independently according to probabilities p_i, $i = 1, 2, \cdots, k$, then the ith process obtained from the splitting has squared coefficient of variation c_i^2 given by

$$c_i^2 = p_i c^2 + 1 - p_i, \tag{36}$$

which is clearly linear. Formula (36) is easy to obtain because the

renewal-interval distribution in the split stream is a geometrically distributed random sum of the original renewal intervals.

4.5 Departures

For the stationary-interval method with single-server nodes, we apply Marshall's formula for the squared coefficient of variation of an interdeparture time, say c_d^2, in a GI/G/1 queue:[43,44]

$$c_d^2 = c_a^2 + 2\rho^2 c_s^2 - 2\rho(1-\rho)\mu EW, \tag{37}$$

where EW is the mean waiting time. Since EW appears in (37), the congestion at the node affects the variability of the departure process. A stationary-interval method approximation for c_d^2 is obtained by inserting an approximation for EW in a GI/G/1 queue. Our analysis[33-37] suggests that it suffices to use the linear approximation (2) with g set equal to one. When this is combined with (37), we obtain the simple formula

$$c_d^2 = \rho^2 c_s^2 + (1-\rho^2)c_a^2. \tag{38}$$

A simple extension of (38) for GI/G/m queues that is being used in the current version of QNA is

$$c_d^2 = 1 + (1-\rho^2)(c_a^2 - 1) + \frac{\rho^2}{\sqrt{m}}(c_s^2 - 1). \tag{39}$$

Note that (39) agrees with (38) when $m = 1$ and (39) yields $c_d^2 = 1$ as it should for M/M/m and M/G/∞ systems for which the stationary departure process is known to be Poisson. The third term in (39) approaches 0 as m increases, reflecting the way multiple servers tend to act as a superposition operation. A basis for further refinements of (39) is the asymptotic analysis of departure processes in Whitt.[45] This asymptotic analysis shows that in some cases the variability of the departure process depends on the arrival and service processes in a more complicated way.

As with superposition, the asymptotic method yields a more elementary approximation than the stationary-interval method. In fact, the asymptotic-method approximation for the departure process is just the arrival process itself, i.e., the asymptotic-method approximation for c_d^2 is just c_a^2.[44] The number of departures in a long interval of time is just the number of arrivals minus the number in queue, and the number in queue fluctuates around its steady-state distribution, whereas the number of arrivals goes to infinity.

It remains to combine the basic methods to form a refined hybrid procedure. However, limited experience indicates that this refinement is not as critical as for superposition. The stationary-interval method

alone seems to perform better for departure processes than for superposition processes.[44]

The most appropriate view for the departure process—the stationary-interval method or the asymptotic method—depends on the traffic intensities at the next nodes where the departures are arrivals. As the traffic intensity of the next node increases, the asymptotic-method approximation for the departure process becomes more relevant. For example, consider the case of two queues in series with parameters λ_{01}, c_{01}^2, μ_1, c_{s1}^2, μ_2, and c_{s2}^2. If $\mu_2 \to \lambda$ while μ_1 remains unchanged, then $\rho_2 \to 1$ and the second queue is in heavy traffic. Under such heavy traffic conditions, it has been shown[26] that the congestion measures at the second node are asymptotically the same as if the first facility were removed, i.e., as if the arrival process to the second node were just the arrival process to the first node. More generally, for any arrival process it has been proved that the asymptotic method is an asymptotically correct approximation for a queue in heavy traffic.[26]

Hence, it is natural to tune the departure approximation by using the traffic intensities in the following nodes. Since the departure process typically will be split and sent to different nodes with different traffic intensities, it is appropriate to do the tuning after splitting. Let c_{di}^2 be the departure c^2 at node i. Then

$$c_{ij}^2 = q_{ij}c_{di}^2 + 1 - q_{ij} \tag{40}$$

is the c^2 for the portion of the departures going to node j. We let c_{ij}^2 be a weighted combination of the approximations obtained by the asymptotic method and the stationary-interval method [using (39)]:

$$\begin{aligned} c_{ij}^2 = \nu_{ij}(q_{ij}c_{ai}^2 + 1 - q_{ij}) \\ + (1 - \nu_{ij})[q_{ij}[1 + (1 - \rho_i^2)(c_{ai}^2 - 1) \\ + \rho_i^2 m_i^{-0.5}(c_{si}^2 - 1)] + 1 - q_{ij}], \end{aligned} \tag{41}$$

where ν_{ij} is chosen to satisfy $0 \leq \nu_{ij} \leq 1$ and be increasing in ρ_j with $\nu_{ij} \to 1$ as $\rho_j \to 1$. However, we have not yet found that positive ν_{ij} helps,[44] so the current version of the QNA uses (28).

From (38) it is clear that the departure process variability, as depicted by QNA, is an appropriate weighted average of the arrival-process variability and the service-time variability. Hence, when the service time is deterministic, so that $c_{sj}^2 = 0$, the departure process is less variable than the arrival process. However, the actual reduction of variability in a network caused by deterministic service times often is not as great as predicted by (38) or (39). Hence, we have replaced (39) by

$$c_d^2 = 1 + (1 - \rho^2)(c_a^2 - 1) + \frac{\rho^2}{\sqrt{m}}(\max\{c_s^2, 0.2\} - 1). \qquad (42)$$

After making this change, we get (25) through (28).

4.6 *Customer creation or combination*

We treat customer creation or combination as a modification of the departure process. When there is customer creation at node i, we replace each departure by a batch of size γ_i. When there is combination at node i, we replace each interdeparture interval by the sum of γ_i^{-1} such intervals. These make more sense for integer values, but we do not require it. Hence, as described in Section 2.2, the departure rate from node i is $\gamma_i\lambda_i$ when the arrival rate is λ_i. We use the asymptotic method to obtain the variability parameter. Since the number of departures from node i in a large time interval is γ_i times the number of arrivals, the asymptotic-method approximation of the variability parameter for customer creation or combination is just to multiply c_{di}^2 by γ_i. (By the asymptotic method, $c^2 = \lim_{t\to\infty}\text{Var } N(t)/EN(t)$; see Section 2 of Whitt.[14]) This is done before splitting.

4.7 *Synthesis*

We obtain the basic system of equations (24) through (30) by combining Sections 4.3 through 4.6 as follows:

$$\begin{aligned} c_{aj}^2 &= 1 - w_j + w_j \sum_{i=0}^{n} p_{ij} c_{ij}^2 \\ &= 1 - w_j + w_j \Bigg[p_{0j} c_{0j}^2 + \sum_{i=1}^{n} p_{ij} (v_{ij}[q_{ij}\gamma_i c_{ai}^2 + (1 - q_{ij})] \\ &\quad + (1 - v_{ij})\{\gamma_i q_{ij}[1 + (1 - \rho_i^2)(c_{ai}^2 - 1) \\ &\quad + \rho_i^2 m_i^{-0.5}(\max\{c_{si}^2, 0.2\} - 1)] + 1 - q_{ij}\}) \Bigg]. \end{aligned} \qquad (43)$$

The first line is based on superposition, Section 4.3, and the second line is based on departure, splitting and customer creation, Sections 4.4 through 4.6.

V. CONGESTION AT THE NODES

Having calculated the rate and variability parameters associated with each internal arrival process, we are ready to calculate the approximate congestion measures for each node. At this point we have decomposed the network into separate service facilities that are analyzed in isolation. Each facility is a standard GI/G/m queue partially characterized by five parameters: the number of servers plus the first

two moments of the interarrival time and the first two moments of the service time. Instead of the moments we use the arrival rate λ, the mean service time τ, and the squared coefficients of variation c_a^2 and c_s^2. Since we are focusing on a single node, we omit the subscript indexing the node throughout this section.

There are many procedures that could be applied at this point. We could fit complete distributions to the parameters,[14] and then apply any existing algorithm for solving a GI/G/m queue or a special case. Among the attractive options are procedures for analyzing the GI/G/1 queue,[46] the M/PH/m queue with phase-type service-time distributions,[47-52] the GI/H_k/m queue with hyperexponential service-time distributions[53,54] and the GI/E_k/m queue with Erlang service-time distributions.[55] Also available are approximations based on heavy-traffic and light-traffic limiting behavior.[56,57] The actual procedures used in this version of QNA, however, are quite elementary. Our study of the GI/G/1 queue[33-37] indicates that these elementary procedures are consistent with the limited information available. Since the arrival process is usually not a renewal process, and since only two moments are known for each distribution, there is little to be gained from more elaborate procedures. In fact, a user of QNA should be cautioned not to rely too heavily on detailed descriptions such as the tail of the waiting-time distribution. Such detailed descriptions may prove to be reasonably accurate, but they should certainly be checked by simulation.

We now describe the congestion measures provided by QNA. In Section 5.1 we treat the single-server node and in Section 5.2 we treat the multiserver node.

5.1 The GI/G/1 queue

We begin with the steady-state waiting time (before beginning service), here denoted by W. The main congestion measure is the mean EW, but we also generate an entire probability distribution for W. First, the approximation formula for the mean is as in (2):

$$EW = \tau\rho(c_a^2 + c_s^2)g/2(1 - \rho), \tag{44}$$

where $g \equiv g(\rho, c_a^2, c_s^2)$ is defined as

$$g(\rho, c_a^2, c_s^2) = \begin{cases} \exp\left[-\dfrac{2(1-\rho)}{3\rho}\dfrac{(1-c_a^2)^2}{c_a^2 + c_s^2}\right], & c_a^2 < 1 \\ 1, & c_a^2 \geq 1. \end{cases} \tag{45}$$

When $c_a^2 < 1$, (44) is the Kraemer and Langenbach-Belz approximation,[58] which is known to perform well.[33-37,59] When $c_a^2 > 1$, the original Kraemer and Langenbach-Belz refinement does not seem to help, so

it is not used. Note that (44) is exact for the M/G/1 queue having $c_a^2 = 1$.

Let the number of customers in the facility, including the one in service, be denoted by N. The probability that the server is busy at an arbitrary time, $P(N > 0)$, and the mean EN can be obtained from Little's formula (see Section 11.3 of Heyman and Sobel[41]):

$$P(N > 0) = \rho \tag{46}$$

and

$$EN = \rho + \lambda EW. \tag{47}$$

Formula (46) is exact even for stationary nonrenewal arrival processes and (47) is exact given EW.

For the probability of delay, $P(W > 0)$, denoted here by σ, we use the Kraemer and Langenbach-Belz approximation:[58]

$$\sigma \equiv P(W > 0) = \rho + (c_a^2 - 1)\rho(1 - \rho)h(\rho, c_a^2, c_s^2), \tag{48}$$

where

$$h(\rho, c_a^2, c_s^2) = \begin{cases} \dfrac{1 + c_a^2 + \rho c_s^2}{1 + \rho(c_s^2 - 1) + \rho^2(4c_a^2 + c_s^2)}, & c_a^2 \leqslant 1 \\ \dfrac{4\rho}{c_a^2 + \rho^2(4c_a^2 + c_s^2)}, & c_a^2 \geqslant 1. \end{cases} \tag{49}$$

Formula (48) also yields the correct value for M/G/1 systems, namely, ρ. Additional supporting evidence for (48) is contained in Whitt.[60]

We next focus on the conditional delay given that the server is busy, denoted by D. Obviously, $ED = EW/\sigma$. We first give an approximation formula for the squared coefficient of variation of D, c_D^2. This formula is the exact formula for the M/G/1 queue, with the service-time distribution being H_2^b when $c_s^2 \geqslant 1$ and E_k when $c_s^2 = k^{-1}$, where H_2^b is the hyperexponential distribution with balanced means and E_k is the Erlang distribution (see p. 256 of Cohen[61] and Section 3 of Whitt[14]). The idea underlying this approximation is that the conditional delay D in a GI/G/1 queue (rather than the total delay W) depends more on the service-time distribution than on the interarrival-time distribution. Hence, the M/G/1 formula for c_D^2 is used as an approximation for all GI/G/1 systems. The M/G/1 formula for c_D^2 is:

$$c_D^2 = 2\rho - 1 + 4(1 - \rho)d_s^3/3(c_s^2 + 1)^2, \tag{50}$$

where $d_s^3 = E(\nu^3)/(E\nu)^3$ with ν being a service-time random variable. Even $E(\nu^3)$ is available, it can be used in (50), but since $E(\nu^3)$ is not available with two parameters, we use approximations for d_s^3. The approximations are based on the H_2^b and E^k distributions.

Case 1: When $c_s^2 \geq 1$,

$$d_s^3 = 3c_s^2(1 + c_s^2), \tag{51}$$

which comes from the H_2^b formulas:

$$d_s^3 = \frac{3}{4}\left[\frac{1}{q^2} + \frac{1}{(1-q)^2}\right]$$

and

$$q = [1 + \sqrt{4(c_s^2 - 1)/(c_s^2 + 1)}]/2.$$

Case 2: When $c_s^2 < 1$,

$$d_s^3 = (2c_s^2 + 1)(c_s^2 + 1). \tag{52}$$

We obtain formula (52) by considering an Erlang E_k variable, which can be represented as the sum of k iid exponential random variables X_i with mean τ/k, where τ is the mean of the E_k variable. In this case

$$\begin{aligned} E(X_1 + \cdots + X_k)^3 &= kE(X_1^3) + 3k(k-1)E(X_1^2)E(X_1) \\ &\quad + k(k-1)(k-2)(EX_1)^3 \\ &= \left(\frac{\tau}{k}\right)^3 [6k + 6k(k-1) + k(k-1)(k-2)] \end{aligned}$$

so that

$$d_s^3 = \frac{(k+2)(k+1)k}{k^3} = \left(1 + \frac{2}{k}\right)\left(1 + \frac{1}{k}\right),$$

which reduces to (52) because $c_s^2 = k^{-1}$ for an E_k variable. Note that (51) and (52) agree at the boundary when $c_s^2 = 1$.

From (44), (48), and (50) through (52), we immediately obtain formulas for $\mathrm{Var}(D)$ and ED^2:

$$\begin{aligned} \mathrm{Var}(D) &= (ED)^2 c_D^2 = (EW)^2 c_D^2/\sigma^2 \\ E(D^2) &= \mathrm{Var}(D) + (ED)^2. \end{aligned} \tag{53}$$

From D we then obtain second-moment characteristics for W:

$$c_W^2 = \frac{E(W^2)}{(EW)^2} - 1 = \frac{\sigma E(D^2)}{(\sigma ED)^2} - 1 = \frac{c_D^2 + 1 - \sigma}{\sigma},$$

$$\mathrm{Var}(W) = (EW)^2 c_W^2 \quad \text{and} \quad E(W^2) = \mathrm{Var}(W) + (EW)^2. \tag{54}$$

We now indicate how QNA calculates an approximate probability distribution for W. The distribution has an atom at zero as given in (48) and a density above zero. The density is chosen so that W and D

have the first two moments already determined for them. (This is the general rule, but it is not quite followed in Cases 2 and 4 below.)

Case 1: $c_D^2 > 1.01$. Let D have the H_2^b density (hyperexponential with balanced means)

$$f_D(x) = p\gamma_1 e^{-\gamma_1 x} + (1-p)\gamma_2 e^{-\gamma_2 x}, \qquad x \geq 0, \tag{55}$$

where

$$p = [1 + \sqrt{(c_D^2 - 1)/(c_D^2 + 1)}]/2,$$

$$\gamma_1 = 2p/ED \quad \text{and} \quad \gamma_2 = 2(1-p)/ED. \tag{56}$$

Case 2: $0.99 \leq c_D^2 \leq 1.01$. Let D have the exponential density with mean ED.

Case 3: $0.501 \leq c_D^2 < 0.99$. Let the distribution of D be the convolution of two exponential distributions with parameters γ_1 and γ_2 $(\gamma_1 > \gamma_2)$, i.e., let D have density

$$f_D(x) = \left(\frac{\gamma_1\gamma_2}{\gamma_1 - \gamma_2}\right)(e^{-\gamma_2 x} - e^{-\gamma_1 x}), \qquad x \geq 0, \tag{57}$$

where

$$\gamma_2^{-1} = \frac{ED + \sqrt{2\ \mathrm{Var}(D) - (ED)^2}}{2}$$

and

$$\gamma_1^{-1} = ED - \gamma_2^{-1}. \tag{58}$$

The associated tail probabilities are

$$P(D > x) = (\gamma_1 e^{-\gamma_2 x} - \gamma_2 e^{-\gamma_1 x})/(\gamma_1 - \gamma_2). \tag{59}$$

Case 4: $c_D^2 < 0.501$. Let D have an E_2 (Erlang) distribution with mean ED, which has $c^2 = 0.5$. Its density is

$$f_D(x) = \gamma^2 x e^{-\gamma x}, \qquad x \geq 0, \tag{60}$$

where $\gamma = 2/ED$. The associated tail probabilities are

$$P(D > x) = e^{-\gamma x}(1 + \gamma x), \qquad x \geq 0. \tag{61}$$

For deterministic service times, $d_s^3 = 1$, so that the smallest possible c_D^2 via (50) is $(1 + 2\rho)/3$. Hence, Case 4 above will not occur often.

Finally, we come to the second moment and variance of N, the number in system. For the M/G/1 queue, it is not difficult to compute $E(N^2)$. Since the steady-state number in the facility is equal to the number of arrivals during a customer's time in the facility, it is easy to compute the moments of N from the moments of W; for example,

$$E(N^2) = \lambda(EW + E\nu) + \lambda^2[E(W^2) + 2EWE\nu + E(\nu^2)]$$
$$= \lambda EW + \rho + \lambda^2 E(W^2) + 2\lambda\rho EW + \rho^2(c_s^2 + 1), \quad (62)$$

and

$$\mathrm{Var}(N) = \lambda EW + \rho + \rho^2 c_s^2 + \lambda^2 \mathrm{Var}(W). \quad (63)$$

We now modify the M/G/1 formulas (62) and (63) for the GI/G/1 queue. Let c_N^2 be defined by

$$c_N^2 = Y_1 Y_2 / Y_3, \quad (64)$$

where Y_1 is the M/G/1 value of $\mathrm{Var}(N)$ in (63) using (44) for EW and (54) and $\mathrm{Var}(W)$,

$$Y_2 = (1 - \rho + \sigma)/\max\{(1 - \sigma + \rho), 0.000001\}$$
$$Y_3 = \max\{(\rho + \lambda EW)^2, 0.000001\}, \quad (65)$$

and σ is the probability of delay in (48). The maximum is used in (65) to avoid dividing by zero. For the M/G/1 queue, $Y_2 = 1$; for GI/M/1 queues, Y_2 in (65) provides just the right correction, so that (64) is exact given the true value of σ, EW and $\mathrm{Var}(W)$. The correction Y_2 in (65) makes (64) too small for D/D/1 queues by a factor of $(1 + \rho)^{-1}$, but (64) is asymptotically correct in heavy traffic: $c_N^2 \to 1$ as $\rho \to 1$ if either $c_a^2 > 0$ or $c_s^2 > 0$.

From (47) and (64) we immediately obtain

$$\mathrm{Var}(N) = (EN)^2 c_N^2$$

and

$$E(N^2) = \mathrm{Var}(N) + (EN)^2. \quad (66)$$

When there is immediate feedback at the node and it is eliminated, adjustments are necessary in the formulas of this section, as indicated in Section III.

5.2 The GI/G/m queue

The first congestion measures for multiserver nodes provided by QNA are exact. Even for nonrenewal stationary-arrival processes, the expected number of busy servers is just the offered load [see p. 400 of Heyman and Sobel[41] and (4.2.3) of Franken et al.[42]]:

$$E \min\{N, m\} = \alpha = \lambda\tau \quad (67)$$

and the traffic intensity or utilization is

$$\rho = \alpha/m. \quad (68)$$

By Little's formula, as in (47),

$$EN = \alpha + \lambda EW. \quad (69)$$

QNA currently provides only a few simple approximate congestion measures for multiserver nodes. These are obtained by modifying the exact formulas for the M/M/m model.[18] Let characteristics such as $EW(c_a^2, c_s^2, m)$ represent the characteristic as a function of the parameters c_a^2, c_s^2, and m, and let characteristics such as $EW(\mathrm{M/M/m})$ be the exact value for M/M/m system. A simple approximation for EW based on heavy-traffic limit theorems[26,56,62,63] is:

$$EW(c_a^2, c_s^2, m) = \left(\frac{c_a^2 + c_s^2}{2}\right) EW(M/M/m). \tag{70}$$

Formula (70) has frequently been used for M/G/m queues and is known to perform quite well in that case.[64–67] By virtue of heavy-traffic limit theorems, we know that (70) is also asymptotically correct for GI/G/m systems as $\rho \to 1$ for fixed m. Limited additional study indicates that (70) is also reasonable for moderate values of ρ when $c_a^2 \geq 0.9$ and $c_s^2 \geq 0.9$, or when $c_a^2 \leq 1.1$ and $c_s^2 \leq 1.1$. The actual value may be significantly smaller (larger) when $c_a^2 < 0.9$ and $c_s^2 > 1.1$ ($c_a^2 > 1.1$ and $c_s^2 < 0.9$).

The simple approximation (70) is also supplemented by simple approximations for the second moments of W and N. They are obtained from:

$$c_W^2(c_a^2, c_s^2, m) = c_W^2(M/M/m)$$

and

$$c_N^2(c_a^2, c_s^2, m) = c_N^2(M/M/m). \tag{71}$$

Related second-moment characteristics are computed as in (54) and (66).

More detailed and sophisticated approximations for multiserver nodes are being studied. As we indicated before, a variety of methods and algorithms can be applied given the parameters of the arrival process.[47–57]

VI. TOTAL NETWORK PERFORMANCE MEASURES

In this section we describe the approximate congestion measures calculated by QNA for the network as a whole. In Section 6.1 we discuss congestion measures representing the system view, e.g., throughput and number of customers in the network; in Sections 6.2 and 6.3 we discuss congestion measures representing the customer view, e.g., number of nodes visited and response times. In fact, there are actually two different customer views. In Section 6.2 we discuss the view of an arbitrary, typical, or aggregate customer; in Section 6.3 we discuss the view of a particular customer with a specified route through the network.

6.1 System congestion measures

A basic total network performance measure is the *throughput*, which we define as the *total external arrival rate* λ_0,

$$\lambda_0 = \lambda_{01} + \cdots + \lambda_{0n}. \tag{72}$$

When no customers are created at the nodes, the total external arrival rate equals the total departure rate from the network, so that there is little ambiguity about what we mean by throughput. However, when customers are created or combined at the nodes, as in Section 2.2, there is more than one possible interpretation. We might be interested in the rate at which arrivals are processed, i.e., (72). For example, the customers created at the nodes might be regarded only as extra work that must be done to serve the arrivals. On the other hand, we might be interested in the rate at which customers leave the network or in the rate of service completions. The departure *rate from the network* is

$$d = \sum_{i=1}^{n} d_i = \sum_{i=1}^{n} \lambda_i \gamma_i \left(1 - \sum_{j=1}^{n} q_{ij}\right) \tag{73}$$

and the total *rate of service completions* is

$$s = \sum_{i=1}^{n} s_i = \sum_{i=1}^{n} \lambda_i \gamma_i. \tag{74}$$

A description of the overall congestion is provided by the mean and variance of the number N of customers in the entire network. In general,

$$EN = EN_1 + \cdots + EN_n \tag{75}$$

and, as an approximation based on assuming that the nodes are independent, we have

$$\mathrm{Var}(N) = \mathrm{Var}(N_1) + \cdots + \mathrm{Var}(N_n). \tag{76}$$

Formula (76) is valid for the Markovian models as a consequence of the product-form solution, but is an approximation in general.

6.2 The experience of an aggregate customer

When we turn to the congestion experienced by individual customers, there are two very different approaches. The first approach keeps strict adherence to the model assumptions with the standard input in Section 2.1, and is based on interpreting the routing matrix as independent probabilities (Markovian routing). This means that each time any customer completes service at node i, that customer proceeds to node j with probability q_{ij}, independent of the current state

and history of the network. If the network is cyclic, this means that every customer has positive probability of visting some nodes more than once. This is the perspective of an aggregate customer. It might be that no individual customer actually ever visits the same node more than once.

If the aggregate view is desired, then the customer experience can be described by employing the basic theory of absorbing Markov chains as in Chapter III of Kemeny and Snell.[68] We can regard the external node as a single absorbing state to which all customers go when they leave the network or we can have more absorbing states, to distinguish between network departures from different nodes or different subsets of nodes. For this interpretation, the routing matrix Q is the transient subchain associated with the absorbing Markov chain and the inverse $(I - Q)^{-1}$ in (19) with $\Gamma = I$ is the *fundamental matrix* of the absorbing chain (see p. 45 of Kemeny and Snell[68]). Solving the traffic-rate equations is tantamount to solving for this fundamental matrix.

From the fundamental matrix it is easy to calculate the moments of the number n_{ij} of visits to any state j starting from any state i (on an external arrival process). For example, En_{ij} is just the (i, j)-th entry of $(I - Q)^{-1}$. It is also easy to calculate the probability of absorption into each of the absorbing states starting from any initial distribution. These various congestion measures are easily obtained working with $n \times n$ matrices.[68]

Suppose that we focus on an arbitrary, typical, or aggregate customer arriving on an external arrival process. Then that customer enters node i with probability λ_{0i}/λ_0, where λ_0 is defined in (72) and the expected number of visits to node i for each customer is

$$EV_i = \lambda_i/\lambda_0. \tag{77}$$

(We have used the fundamental matrix to get λ_i.) The mean of the time, T_i, that an arbitrary customer spends in node i during his or her time in the network is thus

$$ET_i = (EV_i)(\tau_i + EW_i) \tag{78}$$

and the expected total sojourn time (time spent in the network from first arrival to final departure) for an arbitrary customer is thus

$$ET = \sum_{i=1}^{n} ET_i = \sum_{i=1}^{n} EV_i(\tau_i + EW_i). \tag{79}$$

The variance of T_i is thus

$$\mathrm{Var}(T_i) = EV_i(\mathrm{Var}(W_i) + \tau_i^2 c_{si}^2) + \mathrm{Var}(V_i)(EW_i + \tau_i)^2. \tag{80}$$

The term $\mathrm{Var}(V_i)$ in (80) as well as EV_i is easily obtained from the fundamental matrix. In particular, $\mathrm{Var}(V_i) = EV_i^2 - (EV_i)^2$ and

$$EV_i^2 = \sum_{j=1}^{n} (\lambda_{0j}/\lambda_0)[F(2F_{dg} - 1)]_{ji}, \tag{81}$$

where F is the fundamental matrix $(I - Q)^{-1}$, F_{dg} is the $n \times n$ matrix with all off-diagonal entries 0 and diagonal entries the same as F.

To obtain an approximation for the variance of the total sojourn time in the network, we assume that the sojourn times at the different nodes are conditionally independent, given any particular routing. (This is not valid even for all acyclic networks of M/M/1 nodes,[69] but is often approximately true.[7,70]) In particular, for a customer entering some specified node and making V_j visits to node j, $1 \leq j \leq n$, before eventually leaving the network,

$$T = \left(\sum_{j=1}^{n} \sum_{k=1}^{V_j} T_{kj} \right), \tag{82}$$

where T_{kj} is the sojourn time for the kth visit to node j. Our approximation assumption is that the variables T_{kj} are mutually independent given the vector $(V_1, V_2, \cdots, V_n)$.

Hence,

$$\begin{aligned} E(T^2) &= \sum_{i=1}^{n} E\left(\sum_{k=1}^{V_i} T_{ki} \right)^2 \\ &\quad + 2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} E\left(\sum_{k=1}^{V_i} T_{ki} \sum_{\ell=1}^{V_j} T_{\ell j} \right) \\ &= \sum_{i=1}^{n} \{EV_i E(T_{1i}^2) + E[V_i(V_i - 1)]E(T_{1i})^2\}, \\ &\quad + 2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} E(T_{1i})E(T_{1j})E(V_i V_j) \end{aligned} \tag{83}$$

so that

$$\mathrm{Var}(T) = \sum_{i=1}^{n} \mathrm{Var}(T_i) + 2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} E(T_{1i})E(T_{1j})\mathrm{Cov}(V_i, V_j). \tag{84}$$

However, the current version of QNA ignores the covariance terms in (84) in the calculation of $\mathrm{Var}(T)$.

6.3 The experience of a particular customer

Another approach is to decouple the macroscopic and microscopic interpretations. This view is common in statistical mechanics. The total network may exhibit statistical regularity not evidenced in any single particle (customer). In this view, we think of the total system evolving as if customers were routed according to independent probabilities, even though individual customers may have very different

routing probabilities, perhaps nonrandom routing or acyclic routing. For example, we may consider the cyclic network entirely appropriate for the macroscopic view even though no individual customer ever visits any node more than once. In order for this view to be realistic, each individual customer should have a relatively negligible effect on the total network.

The procedure here is to solve for the equilibrium or macroscopic behavior of the network first and then afterwards consider particular customers. The particular customers will have their own routes through the network and perhaps their own service times at the nodes along the way. There are two cases, depending on whether the input is by classes and routes, as in Section 2.3 or the standard input as in Section 2.1.

6.3.1 Input by classes and routes

First, suppose that we are using the input by classes and routes in Section 2.3. Then the particular customers correspond to the customer classes specified in the input. Hence, each customer has a deterministic route through the network and possibly special service times at the nodes on the route. In this case, as described in Section 2.3, QNA first converts the input by classes and routes into the standard input in Section 2.1. Then QNA solves for the equilibrium behavior. Finally, congestion measures are calculated for the different classes under the assumption that they follow their originally specified special routes and that upon arrival at the nodes on the route they see independent versions of the equilibrium state of the network. Hence, in the notation of Section 2.3, for a customer in class k, the expected total service time is

$$\sum_{j=1}^{n_k} \tau_{kj}, \tag{85}$$

the expected total waiting time is

$$\sum_{j=1}^{n_k} E(W_{n_{kj}}), \tag{86}$$

and the expected total sojourn time or response time is the sum of (85) and (86). Similarly, for a customer in class k the variance of the total service time is

$$\sum_{j=1}^{n_k} \tau_{kj}^2 c_{skj}^2, \tag{87}$$

the variance of the total waiting time is

$$\sum_{j=1}^{n_k} \mathrm{Var}(W_{n_{kj}}), \tag{88}$$

and the variance of the total sojourn time is the sum of (87) and (88).

6.3.2 The standard input

With the standard input in Section 2.1, the user must specify the particular customers to be analyzed. In this case, the user specifies classes with routes and possibly service times (rate and variability parameters), but these data are not used in calculating the equilibrium behavior. The decoupling principle is used with greater force here; there need not be any consistency between the microscopic and macroscopic views: This additional input does not affect the equilibrium behavior of the total network.

In the current version of QNA the individual customer routes are deterministic, so that the additional input required is just as in Section 2.3 and the congestion measures are just as in (85) through (88) in Section 6.3.1. However, it is possible to modify QNA to allow random routes. Then the additional input would be just as in Section 2.1; for each class it would consist of a routing matrix plus parameters for the arrivals process and service times.

VII. ACKNOWLEDGMENTS

The software package QNA was written by Anne Seery. She used a subroutine for analyzing the M/M/m queue written by Shlomo Halfin. It has been a pleasure collaborating with Anne Seery on this venture. I also appreciate the help from many other colleagues and the continued support of my management: W. A. Cornell, C. S. Dawson, J. C. Lawson, C. J. McCallum, Jr., M. Segal, R. E. Thomas, and E. Wolman.

REFERENCES

1. L. Kleinrock, *Queueing Systems, Volume 2: Computer Applications*, New York: John Wiley and Sons, 1976.
2. M. Schwartz, *Computer-Communications Network Design and Analysis*, Englewood Cliffs: Pentice-Hall, 1977.
3. F. P. Kelly, *Reversibility and Stochastic Networks*, New York: John Wiley, 1979.
4. E. Gelenbe and I. Mitrani, *Analysis and Synthesis of Computer Systems*, New York: Academic Press, 1980.
5. C. H. Sauer and K. M. Chandy, *Computer Systems Performance Modeling*, Englewood Cliffs: Prentice-Hall, 1981.
6. J. G. Shanthikumar and J. A. Buzacott, "Open Queueing Network Models of Dynamic Job Shops," Int. J. Prod. Res., *19*, No. 3 (1981), pp. 255–66.
7. R. L. Disney, *Queueing Networks and Applications*, Baltimore: The John Hopkins University Lectures, 1982; to be published by the Johns Hopkins University Press.
8. "User's Guide for BEST/1," BGS Systems, Inc., Waltham, Massachusetts, 1980.
9. "User's Manual for CADS," Austin, TX: Information Research Associates, 1978.
10. J. McKenna, D. Mitra, and K. G. Ramakrishnan, "A Class of Closed Markovian Queueing Networks: Integral Representations, Asymptotic Expansions, and Generalizations," B.S.T.J., *60*, No. 5 (May–June 1981), pp. 599–641.

11. J. McKenna and D. Mitra, "Intergral Representations and Asymptotic Expansions for Closed Markovian Queueing Networks: Normal Usage," B.S.T.J., *61*, No. 5 (May–June 1982), pp. 661–83.
12. K. G. Ramakrishnan and D. Mitra, "An Overview of PANACEA, A Software Package for Analyzing Markovian Queueing Networks," B.S.T.J., *61*, No. 10, Part 1 (December 1982), pp. 2849–72.
13. H. Heffes, "Moment Formulae for a Class of Mixed Multi-Job-Type Queueing Networks," B.S.T.J., *61*, No. 5 (May–June 1982), pp. 709–45.
14. W. Whitt, "Approximating a Point Process by a Renewal Process, I: Two Basic Methods," Oper. Res., *30*, No. 1 (January–February 1982), pp. 125–47.
15. S. L. Albin, *Approximating Queues with Superposition Arrival Processes*, Ph.D dissertation, Department of Industrial Engineering and Operations Research, Columbia University, 1981.
16. S. L. Albin, "Approximating a Point Process by a Renewal Process, II: Superposition Arrival Processes to Queues," Department of Industrial Engineering, Rutgers University, 1982.
17. R. I. Wilkinson, "Theories for Toll Traffic Engineering in the U.S.A.," B.S.T.J., *35*, No. 2 (March 1956), pp. 421–514.
18. R. B. Cooper, *Introduction to Queueing Theory*, Second Edition, New York: North Holland, 1981.
19. A. Kuczura, "The Interrupted Poisson Process as an Overflow Process," B.S.T.J., *52*, No. 3 (March 1973), pp. 437–48.
20. J. H. Rath and D. D. Sheng, "Approximations for Overflows from Queues with a Finite Waiting Room," Oper. Res., *27*, No. 6 (November–December 1979), pp. 1208–16.
21. M. Reiser and H. Kobayashi, "Accuracy of the Diffusion Approximation for Some Queueing Systems," IBM J. Res. Dev., *18* (March 1974), pp. 110–24.
22. P. J. Kuehn, "Approximate Analysis of General Queueing Networks by Decomposition," IEEE Trans. Commun., *COM-27*, No. 1 (January 1979), pp. 113–26.
23. K. C. Sevcik, A. I. Levy, S. K. Tripathi, and J. L. Zahorjan, "Improving Approximations of Aggregated Queueing Network Subsystems," in *Computer Performance*, K. M. Chandy and M. Reiser (eds.), Amsterdam: North Holland, 1977, pp. 1–22.
24. K. M. Chandy and C. H. Sauer, "Approximate Methods for Analyzing Queueing Network Models of Computing Systems," ACM Computing Surveys, *10*, No. 3 (September 1978), pp. 281–317.
25. S. Katz, "Statistical Performance Analysis of a Switched Communications Network," Fifth Int. Teletraffic Cong., Rockefeller University, New York, 1967, pp. 566–75.
26. D. L. Iglehart and W. Whitt, "Multiple Channel Queues in Heavy Traffic, II: Sequences, Networks and Batches," Adv. Appl. Prob., *2*, No. 2 (Autumn 1970), pp. 355–69.
27. J. M. Harrison and M. I. Reiman, "On the Distribution of Multidimensional Reflected Brownian Motion," SIAM J. Appl. Math., *41*, No. 2 (October 1981), pp. 345–61.
28. M. I. Reiman, "Open Queueing Networks in Heavy Traffic," unpublished work, 1981.
29. M. I. Reiman, "The Heavy Traffic Diffusion Approximation for Sojourn Times in Jackson Networks," *Applied Probability and Computer Science—The Interface*, Volume 2, R. L. Disney and T. J. Ott (eds.), Boston: Birkhauser, 1982, pp. 409–21.
30. J. M. Holtzman, "The Accuracy of the Equivalent Random Method with Renewal Inputs," B.S.T.J., *52*, No. 9 (November 1973), pp. 1673–9.
31. T. Rolski, "Some Inequalities for GI/M/n Queues," Zast. Mat., *13*, No. 1 (1972), pp. 43–7.
32. A. E. Eckberg, Jr., "Sharp Bounds on Laplace-Stieltjes Transforms, with Applications to Various Queueing Problems," Math. Oper. Res., *2*, No. 2 (May 1977), pp. 135–42.
33. W. Whitt, "On Approximations for Queues, I: Extremal Distributions," B.S.T.J., *63*, No. 1, Part 1 (January 1984), to be published.
34. J. G. Klincewicz and W. Whitt, "On Approximations for Queues, II: Shape Constraints," B.S.T.J., *63*, No. 1, Part 1 (January 1984).
35. W. Whitt, "On Approximations for Queues, III: Mixtures of Exponential Distributions," B.S.T.J., *63*, No. 1, Part 1 (January 1984).
36. W. Whitt, "The Marshall and Stoyan Bounds for IMRL/G/1 Queues are Tight," Oper. Res. Letters, *1*, No. 6 (December 1982), pp. 209–13.

37. W. Whitt, "Refining Diffusion Approximations for Queues," Oper. Res. Letters, *1*, No. 5 (November 1982), pp. 165–9.
38. S. L. Albin, "On Poisson Approximations for Superposition Arrival Processes in Queues," Management Sci., *28*, No. 2 (February 1982), 126–37.
39. W. Whitt, "Queues with Superposition Arrival Processes in Heavy Traffic," unpublished work, 1982.
40. W. Whitt, "Performance of the Queueing Network Analyzer," B.S.T.J., this issue.
41. D. P. Heyman and M. J. Sobel, *Stochastic Models in Operations Research*, Vol. I, New York: McGraw-Hill, 1982.
42. P. Franken, D. König, U. Arndt, and V. Schmidt, *Queues and Point Processes*, Berlin: Akademie-Verlag, 1981.
43. K. T. Marshall, "Some Inequalities in Queueing," Oper. Res., *16*, No. 3 (May–June 1968), pp. 651–65.
44. W. Whitt, "Approximations for Departure Processes and Queues in Series," Nav. Res. Log Qtr., to be published.
45. W. Whitt, "Departures from a Queue with Many Busy Servers," Math. Oper. Res., *9* (1984).
46. A. A. Fredericks, "A Class of Approximations for the Waiting Time Distribution in a GI/G/1 Queueing System," B.S.T.J., *61*, No. 3 (March 1982), pp. 295–325.
47. Y. Takahashi and Y. Takami, "A Numerical Method for the Steady-State Probabilities of a GI/G/c Queueing System in a General Class," J. Oper. Res. Soc. Japan, *19* (1976), pp. 147–57.
48. H. C. Tijms, M. H. van Hoorn, and A. Federgruen, "Approximations for the Steady-State Probabilities in the M/G/c Queue," Adv. Appl. Prob., *13*, No. 1 (March 1981), pp. 186–206.
49. H. Groenevelt, M. H. van Hoorn, and H. C. Tijms, "Tables for M/G/c Queueing Systems with Phase-Type Service," Report No. 85, Department of Actuarial Sciences and Econometrics, The Free University, Amsterdam, The Netherlands, 1982.
50. M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models—An Algorithmic Approach*, Baltimore: The Johns Hopkins University Press, 1981.
51. M. F. Neuts, "A Program for Analyzing the M/PH/c Queue," Department of Mathematics, University of Delaware, 1981.
52. P. Hokstad, "Some Numerical Results and Approximations for the Many Server Queue with Nonexponential Service Time," Department of Mathematics, University of Trondheim, Norway, 1981.
53. J. H. A. de Smit, "The Queue GI/M/s with Customers of Different Types or the Queues $GI/H_m/s$," Adv. Appl. Prob., *15*, No. 2 (June 1983), pp. 392–419.
54. J. H. A. de Smit, "A Program for Analyzing the $GI/H_k/s$ Queue," Department of Applied Mathematics, Twente University of Technology, Enschede, The Netherlands, 1982.
55. A. Ishikawa, "On the Equilibrium Solution for the Queueing System: $GI/E_k/m$," T.R.U. Mathematics, *15*, No. 1 (1979), pp. 47–66.
56. S. Halfin and W. Whitt, "Heavy-Traffic Limits for Queues with Many Exponential Servers," Oper. Res., *29*, No. 3 (May–June 1981), pp. 567–88.
57. D. Y. Burman and D. R. Smith," A Light-Traffic Theorem for Multiserver Queues," Math. Oper. Res., *8*, No. 1 (February 1983), pp. 15–25.
58. W. Kraemer and M. Langenbach-Belz, "Approximate Formulae for the Delay in the Queueing System GI/G/1," *Congressbook*, Eighth Int. Teletraffic Cong., Melbourne, 1976, pp. 235–1/8.
59. J. G. Shanthikumar and J. A. Buzacott, "On the Approximations to the Single Server Queue," Int. J. Prod. Res. *18*, No. 6 (1980), pp. 761–73.
60. W. Whitt, "Minimizing Delays in the GI/G/1 Queue," Oper. Res., *32* (1984), to be published.
61. J. W. Cohen, *The Single Server Queue*, Amsterdam: North-Holland, 1969.
62. J. Köllerström, "Heavy Traffic Theory for Queues with Several Servers. I," J. Appl. Prob., *11*, No. 3 (September 1974), pp. 544–52.
63. J. Köllerström, "Heavy Traffic Theory for Queues with Several Servers. II," J. Appl. Prob., *16*, No. 2 (June 1979), pp. 393–401.
64. A. M. Lee and P. A. Longton, "Queueing Processes Associated with Airline Passenger Check-In," Oper. Res. Quart., *10*, No. 1 (March 1959), pp. 56–71.
65. P. Hokstad, "Approximations for the M/G/n Queue," Oper. Res., *26*, No. 3 (May–June 1978), pp. 510–23.
66. S. A. Nozaki and S. M. Ross, "Approximations in Finite Capacity Multi-Server Queues with Poisson Arrivals," J. Appl. Prob., *15* (1978), pp. 826–34.

67. F. S. Hillier and O. S. Yu, *Queueing Tables and Graphs*, New York: North-Holland, 1981.
68. J. G. Kemeny and J. L. Snell, *Finite Markov Chains*, Princeton: Van Nostrand, 1960.
69. B. Simon and R. D. Foley, "Some Results on Sojourn Times in Acyclic Jackson Networks," Management Sci., *25*, No. 10 (October 1979), pp. 1027–34.
70. P. C. Kiessler, "A Simulation Analysis of Sojourn Times in a Jackson Network," Report VTR 8016, Department of Industrial Engineering and Operations Research, Virginia Polytechnic Institute and State University, 1980.

AUTHOR

Ward Whitt, A.B. (Mathematics), 1964, Dartmouth College; Ph.D. (Operations Research), 1968, Cornell University; Stanford University, 1968–1969; Yale University, 1969–1977; Bell Laboratories, 1977—. At Yale University, from 1973–1977, Mr. Whitt was Associate Professor in the departments of Administrative Sciences and Statistics. At Bell Laboratories he is in the Operations Research Department in the Network Analysis Center.