Journal of the Operations Research
Society of Japan
Vol. 40, No. 2, June 1997

TWO-LAYER QUEUEING NETWORKS

Issei Kino
C&C Research Laboratories NEC

(Received December 5, 1994; Final December 5, 1996)

Abstract A family of queueing networks with a two-layer configuration is proposed and analyzed in order to provide well structured hierarchy of network models for performance analysis of computer and/or communication systems. The upper layer describes the often-disregarded software behavior while the lower layer describes the usual hardware behavior. For the case in which the upper layer includes no outstanding queues, a product form equilibrium joint distribution is established assuming state-dependent arrival and state-dependent service rate functions with general service time distributions. The marginal distributions are derived for convenience in applying the results. For the case in which the upper layer dose include outstanding queues, an approximation method is proposed, which generalizes the flow-equivalent methods.

1. Introduction

Queueing networks have been used effectively as performance evaluation models in practical applications, and most of these are based on queueing networks with product form distributions studied by Jackson [16], Baskett et al. [2], Chandy [10], Chandy and Martin [11], Kelly [19], and Whittle [36]. Many authors, including Serfozo [28], [29], [30], Van Dijk [32], [5], Henderson et al. [15], and Miyazawa [25] have extended the class of product form type of networks. For related studies, see Refs. [4], [14], and [34].

In this paper, queueing networks with two layer configuration are proposed to provide well structured hierarchical network models for performance analysis of practical applications. Two-layer queueing network consists of an upper layer and a lower layer. The upper layer consists of multiple stations and the lower layer consists of multiple queues. Each station in the upper layer is associated with a routing chain according to which customers travel through queues in the lower layer. The routing structure for customers to travel through in the upper layer may consist of open, closed, or mixed routing chains, and customers in the upper layer are distinguished by types according to their routing chains. Each queue in the lower layer is assumed to be either a symmetric queue or a local balance queue. A general framework is considered for service time distribution and for service-rate and arrival-rate functions that depend on the global state of the network.

This paper considers two types of two-layer queueing networks:those without outstanding queues, and those with outstanding queues. For the two-layer queueing network without outstanding queues, each station in the upper layer is assumed to be an infinite server station so that there is no capacity constraint for a customer to enter each station. For the two-

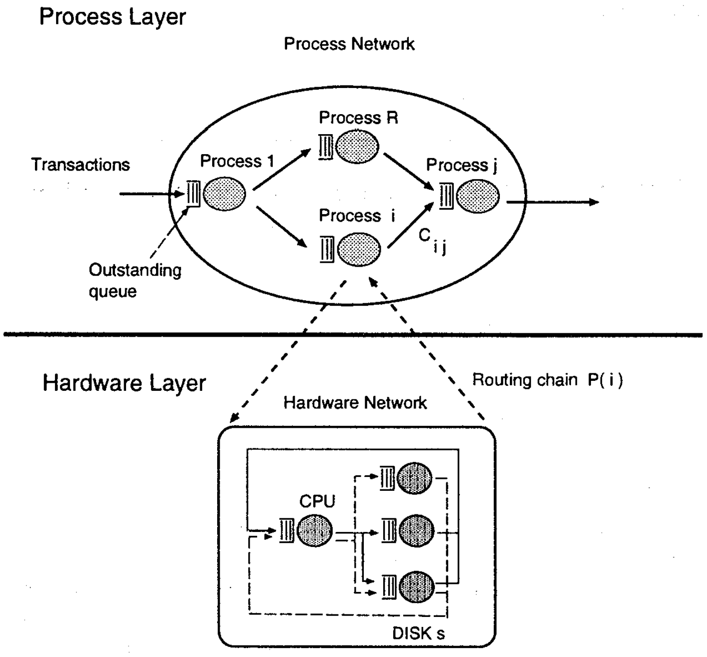

Figure 1: Two-layer network for a computer system.

layer queueing networks with outstanding queues, on the other hand, capacity constraints for each station in the upper layer are assumed. Thus, customers less than or equal to the constraint number can enter the station and can travel through the queues in the lower layer according to the routing chain associated with the station. The excess customers have to wait in the outstanding queue until the number of customers in the station become less than the constraint number of the station. The two-layer queueing networks with outstanding queues are generalizations of simultaneous resource possession models and passive server models proposed by Sauer [26]. Here the product form equilibrium distribution for the state of the two-layer networks without outstanding queues is derived by using the supplementary variable method [12]. This distribution shows that arrival rate of customers depends on the population vector of customers in the upper layer, but dose not depend on the state of the lower layer. On the other hand, the service rate for customers at each queue in the lower layer depends on the states of both the upper layer and the lower layer.

An approximate method to analyze two-layer networks with outstanding queues is then proposed, and applications of this method to the evaluation of the performance of computer systems are suggested. The approximation technique is an extension of the standard decomposition or flow-equivalent method to which many authors have contributed: see, for example, Refs. [8], [9], [1], [3], [13], [22], [35]. Thus one of effective application of the two-layer networks with outstanding queues may be found in the field of performance evaluation of computer systems. Various software packages for the application field have been developed, such as BEST/1 [7], RESQ [27], QNAP [33], and QM-X [20]. Conventionally, hardware resources CPU, DISK, etc. are considered to be the principal origin of congestion delay in a computer system. Congestion due to software resources, however,also causes significant performance degradation. For example, one may find a hierarchical structure between the hardware resource congestion and the *process* congestion ("*process*" is used here in the same sense in which it is used in the field of computer science). A transaction requires a number of *processes* to finish its work, which is accomplished by using each *process* one by one. To allocate hardware resources to the transaction, an operating system always allocates a *process* to a transaction in advance. A transaction thus cannot use any hardware resource unless a *process* is assigned to the transaction. Each *process* has a capacity constraint specifying the greatest number of transactions that can use the *process* at the same time. A transaction that finds the *process* busy has to wait in an outstanding queue until the operating system can assign the *process* to it. Thus a *process* itself may cause congestion, and the service time at the *process* may depend on delay times due to hardware resource congestion. Because *process* congestion and hardware resource congestion are closely related, the calculation of performance measures for computer systems requires unified analysis of both types of congestion.

The two-layer queueing network with outstanding queues can consistently represent the hierarchical structures of *processes* as well as the hardware resources. Each *process* can be described by a station in the upper layer, and *process* switching sequences for a type of transaction can be described by a Markov transition matrix. Each hardware resource can be described by a queue in the lower layer, and a process's sequence of hardware resource utilization can be described by the associated routing chain. Each waiting queue due to *process* congestion can be described by an outstanding queue in the upper layer. Figure 1 shows an application of the two-layer model with outstanding queues to a computer system with both *process* congestion and hardware congestion.

The outline of this paper is as follows. Model and notations are described in the next Section, the product form distribution for the two layer queueing networks without outstanding queues is derived in Section 3. To make the use of this kind of distribution more convenient in practical applications, marginal distributions of the original product form distribution and throughput between the upper layer and the lower layer are derived with respect to the aggregate network states in Section 4. The two-layer queueing networks with outstanding queues are described and an approximation technique is proposed in Section 5, and conclusions are given in Section 6. Details of the proof for the product form distribution is given in Appendix 1, and notations are listed in Appendix 2.

2. Two-layer networks without outstanding queues

Model description and notations for two-layer networks without outstanding queues are given in this section. The notations are summarized in Appendix 2. An example of a two-layer queueing network without outstanding queues is shown in Fig. 2.

Vector notations: Throughout the paper vectors are denoted by boldface italic letters, and are row vectors. The vector $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ is written in the form $\boldsymbol{x} = (x_i)_{i=1}^{n}$, and the transpose of vector $\boldsymbol{x}$ is denoted by $\boldsymbol{x}^t$. Let $\boldsymbol{x} = (x_i)_{i=1}^{n}$, $\boldsymbol{y} = (y_i)_{i=1}^{n}$, and $\boldsymbol{\rho} = (\rho_i)_{i=1}^{n}$.

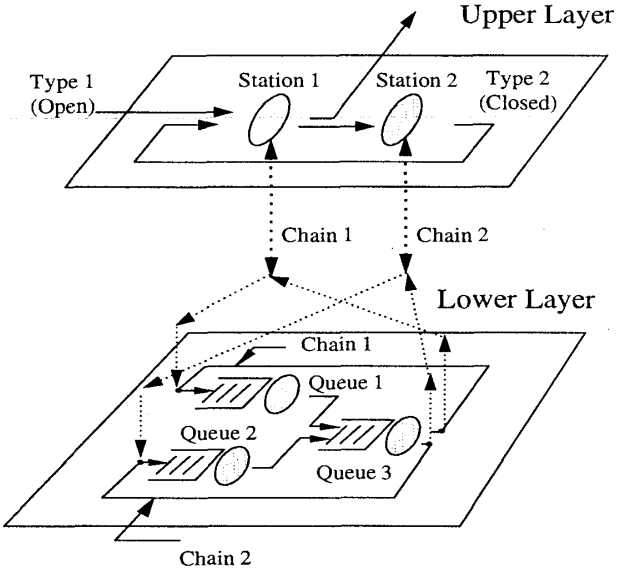

Figure 2: Two-layer network without outstanding queues.

For simplicity,

$$| \boldsymbol{x} |= n \ , \quad \|\boldsymbol{x}\| = x_1 + x_2 + \cdots + x_n \ , \quad \boldsymbol{x}! = x_1 ! x_2 ! \cdots x_n ! \ ,$$

$$\boldsymbol{\rho}^{\boldsymbol{x}} = \rho_1^{x_1} \rho_2^{x_2} \cdots \rho_n^{x_n} \quad \text{and} \quad \boldsymbol{e}_i = (0, \cdots, 0, \overset{(i-th)}{1}, 0, \cdots, 0) \ .$$

The length of vector $\boldsymbol{e}_i$ may be interpreted in the context in which it appears.

Random variables: For simplicity, the notations used in this paper will usually not distinguish between random variables and particular values of them. For example, if the context is such that confusion is unlikely, the notation $P(\boldsymbol{x})$ is used for the probability that the random variables $(\boldsymbol{x})$ take the generic values of $(\boldsymbol{x})$.

Stations and queues: Consider a queueing network consisting of two layers, an upper layer and a lower layer. Suppose that there are a number R of stations labeled $1, 2, \cdots, R$ in the upper layer and a number N of queues labeled $1, 2, \cdots, N$ in the lower layer. The outside of the network is labeled by 0. Each station in the upper layer is assumed to be an infinite server queue, so that there is no waiting time for a customer to enter in each station.

Type of customer: Suppose there are a number M of customer types labeled $1, 2, \cdots, M$. A customer of type m at station i, after the completion of his service, either joins station j with probability $s_{ij}(m)$ or leaves the network with probability $s_{i0}(m)$. A type-m customer who arrives from outside of the network joins station i with probability $s_{0i}(m)$. Let I_M be an integer set of $\{1, 2, \cdots, M\}$. Define G_m for $m \in I_M$ be a set of stations which a type-m customer may visit. Each G_m is a subset of I_M. Assume that

$$s_{i0}(m) = 1 - \sum_{j \in G_m} s_{ij}(m) \quad \text{and} \quad \sum_{i \in G_m} s_{0i}(m) = 1 \quad \text{for} \quad m \in I_M \ .$$

The routing chain of a type-m customer in the upper layer is written in the form

$$S(m)=\left(\begin{array}{c|c} 0 & \boldsymbol{s}_{(+)}(m) \\ \hline \boldsymbol{s}_{(-)}(m) & s_{ij}(m) \end{array}\right) \tag{2.1}$$

for $m \in I_M$ and $i,j \in G_m$, where $\boldsymbol{s}_{(+)}(m)=(s_{0i}(m))_{i\in G_m}$ and $\boldsymbol{s}_{(-)}(m)=(s_{i0}(m))^t_{i\in G_m}$.

If a customer of type m enters the network again immediately after his departure from the network, then the routing chain becomes a closed routing chain in the form

$$S_c(m)=\boldsymbol{s}_{(-)}(m)\boldsymbol{s}_{(+)}(m)+S^*(m) , \tag{2.2}$$

where the matrix $S^*(m)=\{s_{ij}(m)\}$,$i,j \in G_m$. Thus the routing chain of each type of customers may be an open or closed chain, and the routing structure in the upper layer may consist of mixed routing chains.

Class of customer: Suppose that there are a number K of customer classes labeled $1,2,...,K$. Let I_K be an integer set of $\{1,2,\cdots,K\}$. The class of a customer identifies the service-time distribution for the customer. Without loss of generality, those classes are treated globally across the queues since individual queues need not be given to customers of all classes. Service time S_k for a class-k customer is assumed to have a distribution function F_k for each $k \in I_K$. Assume that each distribution function is differentiable and has a finite mean. The probability density function and the reciprocal of the mean are denoted by $f_k(y)$ and $\mu_k=1/E(S_k)$.

Routing chains in the lower layer: Let I_N and I_R be integer sets of $\{1,2,\cdots,N\}$ and $\{1,2,\cdots,R\}$. Suppose that there are a number R of Markovian routing chains labeled $1,2,...,R$ associated with stations $1,2,\cdots,R$. Those routing chains specify the routing rules for customers traveling through queues in the lower layer. When a customer arrives at station i in the upper layer, the routing chain $P(i)$ is assigned to the customer without regard to his customer type. A customer whose routing chain is $P(i)$ will be henceforth referred to as a station-i customer or a customer of station i. A customer of station i may change both his class and queue according to the routing chain $P(i)$ after completion of his service time at a queue.

A customer may join queue j in the lower layer as a class k customer with probability $r_{*,(j,k)}(i)$ ($j \in I_N$,$k \in I_K$, $i \in I_R$) immediately after his arrival at station i in the upper layer. After service completion of a station-i customer of class k at queue j, that customer may join queue h as a class-l customer with probability $r_{(j,k),(h,l)}(i)$ ($j,h \in I_N$, $k,l \in I_K$, $i \in I_R$) or depart from the lower layer and come back to the upper layer with probability $r_{(j,k),*}(i)$. When a type-m customer comes back to the upper layer, he can either leave the network or choose his next station and immediately enter the lower layer again as a customer of a different station according to a routing chain $S(m)$. Assume that for $i \in I_R$,

$$r_{(j,k),*}(i)=1-\sum_{h\in I_N,\, l\in I_K} r_{(j,k),(h,l)}(i) \quad (j\in I_N,\ k\in I_K) , \qquad \sum_{j\in I_N,\, k\in I_K} r_{*,(j,k)}(i)=1.$$

Define the matrix $Q(i)=\{r_{(j,k),(h,l)}(i)\}$ ($j,h \in I_N$; $k,l \in I_K$; $i \in I_R$), the row vector $\boldsymbol{r}(i)=((r_{*,(j,k)}(i))^K_{k=1})^N_{j=1}$, and the column vector $\boldsymbol{q}(i)=(((r_{(j,k),*}(i))^K_{k=1})^N_{j=1})^t$. The

Markovian routing chain for a station- i customer is written in the form

$$P(i)=\left(\begin{array}{c|c} 0 & \boldsymbol{r}(i) \\ \hline \boldsymbol{q}(i) & Q(i) \end{array}\right) \quad \text{for} \quad i \in I_R. \tag{2.3}$$

Service time at a station: The service time of a customer at a station in the upper layer is defined by a period of time from an epoch in which a customer enters the lower layer to an epoch in which the customer comes back the upper layer. Characteristics of the service time of a customer at each station in the upper layer cannot be specified a priori because they may depend on the overall state of the network.

States description and network occupancy: Each queue in the lower layer consists of a set of positions, each of which may be occupied by one customer. If there are n customers in a queue, the occupied positions are indexed by $1,2,\cdots,n$.

A customer in position l at queue j is indexed by his type index u_{jl}, his station index v_{jl} , and his class index w_{jl} . That is, if a type-m and station- i customer of class k is in position l at queue j,

$$u_{jl}=m\ , \quad v_{jl}=i\ , \quad \text{and} \quad w_{jl}=k\ .$$

Define the state of position l at queue j by a triplet of indexes (type, station, class) of a customer in position l at queue j. Let $\boldsymbol{c}_{jl}=(u_{jl},\ v_{jl},\ w_{jl})$, $\boldsymbol{c}_j=(\boldsymbol{c}_{jl})_{l=1}^{|\boldsymbol{c}_j|}$, and $\boldsymbol{c}=(\boldsymbol{c}_j)_{j=1}^{N}$. Those vectors, – $\boldsymbol{c}_{jl}$, $\boldsymbol{c}_j$; and $\boldsymbol{c}$ – are respectively referred to as the occupancy of position l at queue j, the occupancy of queue j, and the occupancy of network. They describe discrete parts of the network state. Let y_{jl} be a positive real supplementary random variable representing the remaining service time of a customer in position l at queue j. Let $\boldsymbol{y}_j=(y_{jl})_{l=1}^{|\boldsymbol{c}_j|}$ and $\boldsymbol{y}=(\boldsymbol{y}_j)_{j=1}^{N}$. Those random variables provide the continuous parts of the network state.

When a customer of class k arrives at position l of queue j, the new y_{jl} is chosen according to the distribution function F_k. When the y_{jl} reaches zero, the customer departs from the position. Using those notations, we can denote the state of queue j ($j\in I_N$) by $(\boldsymbol{c}_j,\ \boldsymbol{y}_j)$ and can write the complete state description of queues as $(\boldsymbol{c},\ \boldsymbol{y})$, where

$$\boldsymbol{c}=(\boldsymbol{c}_1,\ \boldsymbol{c}_2,\ \cdots,\ \boldsymbol{c}_N\ ) \quad \text{and} \quad \boldsymbol{y}=(\boldsymbol{y}_1,\ \boldsymbol{y}_2,\ \cdots,\ \boldsymbol{y}_N\ )\ .$$

State-dependent queueing discipline: Assume that queue j ($j\in I_N$) operates in the following manner. When a customer finds the occupancy $\boldsymbol{c}$ at his arrival instant, he moves into position l ($l=1,2,\cdots,|\boldsymbol{c}_j|+1$) of the queue j with a queue-occupancy- dependent probability $\delta_j(l,\ \boldsymbol{c})$. Customers previously in positions $l,l+1,\cdots,|\boldsymbol{c}_j|$ move to positions $l+1,l+2,\cdots,|\boldsymbol{c}_j|+1$ at the queue j. Assume that for all feasible $\boldsymbol{c}$,

$$\sum_{l=1}^{|\boldsymbol{c}_j|+1} \delta_j(l,\boldsymbol{c})=1 \quad \text{for} \quad j\in I_N\ . \tag{2.4}$$

Given a network occupancy is $\boldsymbol{c}$, let $\gamma_j(l,\ \boldsymbol{c})$ be a network-occupancy-dependent service rate for a customer in position l at queue j. That is,

$$\gamma_j(l,\ \boldsymbol{c})=-\frac{d}{dt}\,y_{jl} \quad \text{for} \quad j\in I_N\ , \quad l=1,2,\cdots,|\boldsymbol{c}_j|\ . \tag{2.5}$$

When a customer in position l at queue j leaves the queue after his service completion, customers in positions $l+1, l+2, \cdots, |\boldsymbol{c}_j|$ move to positions l, $l+1, \cdots, |\boldsymbol{c}_j|-1$.

Aggregation of states: Given queue occupancy $\boldsymbol{c}_j$, let $x_{j(m,i)}(\boldsymbol{c}_j)$ be the total number of type-m and station-i customers at queue j. That is,

$$x_{j(m,i)}(\boldsymbol{c}_j) = \sum_{l=1}^{|\boldsymbol{c}_j|} \mathrm{I}(u_{jl} = m, v_{jl} = i) \ , \quad j \in I_N \ , \quad m \in I_M \ , \quad i \in I_R$$

where the $\mathrm{I}(A)$ is an indicator function for a statement A, and let

$$\boldsymbol{x}_{jm}(\boldsymbol{c}_j) = (\ x_{j(m,1)}(\boldsymbol{c}_j),\ x_{j(m,2)}(\boldsymbol{c}_j),\ \cdots\ ,\ x_{j(m,R)}(\boldsymbol{c}_j))$$

and

$$\boldsymbol{x}_j(\boldsymbol{c}_j) = (\ \boldsymbol{x}_{j1}(\boldsymbol{c}_j),\ \boldsymbol{x}_{j2}(\boldsymbol{c}_j),\ \cdots\ ,\ \boldsymbol{x}_{jM}(\boldsymbol{c}_j))$$

for $j \in I_N$ and $m \in I_M$. Note that $\|\boldsymbol{x}_j(\boldsymbol{c}_j)\| = |\boldsymbol{c}_j|$. The aggregate network occupancy vector $\boldsymbol{x}(\boldsymbol{c})$ is defined by

$$\boldsymbol{x}(\boldsymbol{c}) = (\boldsymbol{x}_1(\boldsymbol{c}_1),\ \boldsymbol{x}_2(\boldsymbol{c}_2),\ \cdots,\ \boldsymbol{x}_N(\boldsymbol{c}_N))\,.$$

Given network occupancy $\boldsymbol{c}$, let $m_s(\boldsymbol{c})$ be a total number of type-s customers in the network. That is, for $s \in I_M$,

$$m_s(\boldsymbol{c}) = \sum_{j=1}^{N} \sum_{l=1}^{|\boldsymbol{c}_j|} \mathrm{I}(u_{jl} = s) \ = \sum_{j=1}^{N} \sum_{i=1}^{R} x_{j(s,i)}(\boldsymbol{c}_j).$$

The type occupancy $\boldsymbol{m}(\boldsymbol{c})$, given the network occupancy $\boldsymbol{c}$, is defined by

$$\boldsymbol{m}(\boldsymbol{c}) = (m_1(\boldsymbol{c}), m_2(\boldsymbol{c}), \cdots, m_M(\boldsymbol{c}))\,.$$

Let $z_{ji}(\boldsymbol{c}_j)$ be a total number of station-i customers at queue j. Formally,

$$z_{ji}(\boldsymbol{c}_j) = \sum_{l=1}^{|\boldsymbol{c}_j|} \mathrm{I}(v_{jl} = i) \ = \sum_{m=1}^{M} x_{j(m,i)}(\boldsymbol{c}_j)\ .$$

Denote

$$\boldsymbol{z}_j(\boldsymbol{c}_j) = (z_{j1}(\boldsymbol{c}_j), z_{j2}(\boldsymbol{c}_j), \cdots, z_{jR}(\boldsymbol{c}_j)) \quad \text{and} \quad \boldsymbol{z}(\boldsymbol{c}) = (\boldsymbol{z}_1(\boldsymbol{c}_1), \boldsymbol{z}_2(\boldsymbol{c}_2), \cdots, \boldsymbol{z}_N(\boldsymbol{c}_N))\,.$$

Let $n_i(\boldsymbol{c})$ be the number of station-i customers in a network whose occupancy is $\boldsymbol{c}$. Formally, for $i \in I_R$,

$$n_i(\boldsymbol{c}) = \sum_{j=1}^{N} \sum_{l=1}^{|\boldsymbol{c}_j|} \mathrm{I}(v_{jl} = i) \ = \sum_{j=1}^{N} \sum_{m=1}^{M} x_{j(m,i)}(\boldsymbol{c}_j).$$

The station occupancy $\boldsymbol{n}(\boldsymbol{c})$ is defined by

$$\boldsymbol{n}(\boldsymbol{c}) = (n_1(\boldsymbol{c}), n_2(\boldsymbol{c}), \cdots, n_R(\boldsymbol{c}))\,.$$

Note that

$$\boldsymbol{z}_j(\boldsymbol{c}_j) = \boldsymbol{x}_{j1}(\boldsymbol{c}_j) + \boldsymbol{x}_{j2}(\boldsymbol{c}_j) + \cdots + \boldsymbol{x}_{jM}(\boldsymbol{c}_j)$$

and

$$\boldsymbol{n}(\boldsymbol{c}) = \boldsymbol{z}_1(\boldsymbol{c}_1) + \boldsymbol{z}_2(\boldsymbol{c}_2) + \cdots + \boldsymbol{z}_R(\boldsymbol{c}_R) .$$

Network-occupancy-dependent service rate.: Let $\Phi(\boldsymbol{x}(\boldsymbol{c}))$ be a positive real function of the aggregate network occupancy $\boldsymbol{x}(\boldsymbol{c})$. Assume that service rate $\gamma_j(l, \boldsymbol{c})$ depends on the aggregate network occupancy in the following way:

$$\gamma_j(l,\, \boldsymbol{c}) = \frac{\Phi(\boldsymbol{x}(\boldsymbol{c}) - \boldsymbol{e}_j(u_{jl}, v_{jl}))}{\Phi(\boldsymbol{x}(\boldsymbol{c}))}\, \beta_j(l, \boldsymbol{c}) , \quad \text{for} \quad l = 1, 2, \cdots, |\boldsymbol{c}_j| . \tag{2.6}$$

Here $\boldsymbol{e}_j(m, i)$ is a unit vector whose element is one if the position corresponds to the position of type m and station i at queue j in the vector $\boldsymbol{x}(\boldsymbol{c})$. Formally, $\boldsymbol{e}_j(m, i)$ is a unit vector whose $((j-1)RM + (m-1)R + i)$-th element is one and whose others elements are zero. Assume that $\Phi(0, 0, \cdots, 0) = 1$ and

$$\sum_{l=1}^{|\boldsymbol{c}_j|} \beta_j(l, \boldsymbol{c}) = 1 , \quad j \in I_N , \quad l = 1, 2, \cdots, |\boldsymbol{c}_j| . \tag{2.7}$$

The function $\Phi(\boldsymbol{x})$ is referred to a service-rate function. By choosing the service-rate function $\Phi(\boldsymbol{x})$ appropriately, one can represent various types of queueing behavior.

Type-occupancy-dependent arrival rate: Define a positive real function $\Lambda(\boldsymbol{m})$ of vector $\boldsymbol{m}$ $(= (m_i)\ {}_{i=1}^{M})$ each element of which is a non-negative integer. The function Λ is referred to as an arrival function. Assumed that $\Lambda(0, 0, \cdots, 0) = 1$.

Assume that customers of type m arrive at the network according to a Poisson stream with rate $\lambda_m(\boldsymbol{c})$ that depends on type occupancy $\boldsymbol{m}(\boldsymbol{c})$ in the following manner:

$$\lambda_m(\boldsymbol{c}) = \frac{\Lambda(\boldsymbol{m}(\boldsymbol{c}) + \boldsymbol{e}_m)}{\Lambda(\boldsymbol{m}(\boldsymbol{c}))} , \quad m \in I_M . \tag{2.8}$$

Each Poisson stream is assumed to be independent of the others. The total arrival rate of customers from the outside of the network becomes

$$\lambda(\boldsymbol{c}) = \sum_{m \in I_M} \lambda_m(\boldsymbol{c}) . \tag{2.9}$$

One can formulate various types of networks – such as open, closed, mixed networks [2] as well as loss systems and triggered arrival systems [23] – by specifying the arrival function Λ appropriately.

3. Product form distribution

The product form equilibrium distribution for the two layer queueing network is derived in this section.

Traffic equation: Define the $(N \times K)$-dimensional square matrix

$$D_{ik} = \boldsymbol{q}(i)\, \boldsymbol{r}(k) \quad \text{for} \quad i, k \in I_R .$$

For $m \in I_M$, define a block diagonal matrix

$$Q^*(m) = diag\{\,0\,,\ Q(i_1),\ Q(i_2),\ \cdots,\ Q(i_r)\}\ ,$$

a square matrix

$$V(m) = \left(\begin{array}{c|cccc} 0 & s_{0i_1}(m) \times \boldsymbol{r}(i_1) & s_{0i_2}(m) \times \boldsymbol{r}(i_2) & \cdots & s_{0i_r}(m) \times \boldsymbol{r}(i_r) \\ \hline s_{i_10}(m) \times \boldsymbol{q}(i_1) & s_{i_1i_1}(m)D_{i_1\,i_1} & s_{i_1\,i_2}(m)D_{i_1\,i_2} & \cdots & s_{i_1\,i_r}(m)D_{i_1\,i_r} \\ s_{i_20}(m) \times \boldsymbol{q}(i_2) & s_{i_2\,i_1}(m)D_{i_2\,i_1} & s_{i_2\,i_2}(m)D_{i_2\,i_2} & \cdots & s_{i_2\,i_r}(m)D_{i_2\,i_r} \\ \vdots & \vdots & \vdots & & \vdots \\ s_{i_r0}(m) \times \boldsymbol{q}(i_r) & s_{i_r\,i_1}(m)D_{i_r\,i_1} & s_{i_r\,i_2}(m)D_{i_r\,i_2} & \cdots & s_{i_ri_r}(m)D_{i_r\,i_r} \end{array}\right)$$

and a traffic matrix

$$T(m) = V(m) + Q^*(m) \tag{3.1}$$

where $i_j \in G_m$ for $j = 1, 2, \cdots, r$.

The matrix $Q^*(m)$ expresses transition probabilities for a type-m customer to move through queues in the lower layer without changing his station and the matrix $V(m)$ expresses transition probabilities for the customer to choose next queue with changing his station in the upper layer. Let

$$\boldsymbol{\theta}(m) = (\boldsymbol{\theta}(m,1)\,, \boldsymbol{\theta}(m,2),\ \cdots,\ \boldsymbol{\theta}(m,R))$$

where, for $j \in I_N$, $m \in I_M$, $i \in I_R$,

$$\boldsymbol{\theta}(m,i) = (\boldsymbol{\theta}_1(m,i), \boldsymbol{\theta}_2(m,i),\ \cdots,\ \boldsymbol{\theta}_N(m,i))$$

and $\boldsymbol{\theta}_j(m,i) = (\theta_j(m,i,1),\ \theta_j(m,i,2),\ \cdots,\ \theta_j(m,i,K))$.

Assuming that the Markovian transition matrix $T(m)$ $(m \in I_M)$ is irreducible and positive, in which case there exists an unique stationary distribution, we can formulate the traffic equation for the two-layer network in the following form:

$$(\,1,\ \boldsymbol{\theta}(m)\,) = (\,1,\ \boldsymbol{\theta}(m)\,)\,T(m) \quad \text{for} \quad m \in I_M\ . \tag{3.2}$$

Note that the vector $(1, \boldsymbol{\theta}(m))$ is not a probability vector but the vector normalized such that the first element should be one. Thus equation (3.2) can be solved uniquely. The solution $\theta_j(m,i,k)$ represents the relative frequency with which a type-m and station- i customer of class k visits queue j when $j \in I_N,\ m \in I_M,\ i \in I_R,\ k \in I_K$. Similarly, the traffic equation for a type-m customer to change his station in the upper layer is written in the form

$$(\,1,\ \boldsymbol{v}(m)\,) = (\,1,\ \boldsymbol{v}(m)\,)\ S(m)\ , \tag{3.3}$$

where $\boldsymbol{v}(m) = (v_{i_1}(m),\ v_{i_2}(m),\ \cdots,\ v_{i_r}(m)\,)$ and $i_j \in G_m$ for $j = 1, 2, \cdots, r$. The term $v_i(m)$ $(i \in G_m)$ is the relative frequency with which a type-m customer visits station i.

The traffic equation for a station i customer to visit queues in the lower layer is written in the form

$$(\,1,\ \boldsymbol{w}(i)\,) = (\,1,\ \boldsymbol{w}(i)\,)\,P(i)\ , \tag{3.4}$$

where $\boldsymbol{w}(i) = ((w_{(j,k)}(i))_{k=1}^{K})_{j=1}^{N}$. The term $w_{(j,k)}(i)$ is the relative frequency with which a station-i customer visits queue j as a class-k customer. Note that

$$(\boldsymbol{v}(m),\ \boldsymbol{s}_{(-)}(m)) = 1 \quad \text{and} \quad (\boldsymbol{w}(i),\ \boldsymbol{q}(i)) = 1\ , \tag{3.5}$$

where $(\boldsymbol{x},\ \boldsymbol{y})$ denotes the inner product of vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. The following proposition can be derived straightforwardly from the definitions of traffic equations (3.2), (3.4), and (3.5).

Proposition 3.1 *For* $j \in I_N$, $m \in I_M$, $i \in I_R$,

$$\theta_j(m, i, k) = v_i(m) w_{(j,k)}(i)\ . \tag{3.6}$$

Traffic intensity: The traffic intensity due to a type-m and station-i customer of class k at queue j is defined by

$$\sigma_j(m, i, k) = \frac{\theta_j(m, i, k)}{\mu_k}\ . \tag{3.7}$$

Define the traffic-intensity function at queue j in the form

$$\tau_j(\boldsymbol{c}_j) \ = \ \prod_{l=1}^{|\boldsymbol{c}_j|} \sigma_j(\boldsymbol{c}_{jl}) \ = \ \prod_{l=1}^{|\boldsymbol{c}_j|} \frac{\theta_j(u_{jl},\ v_{jl},\ w_{jl})}{\mu_{w_{jl}}}\ . \tag{3.8}$$

Definition 3.1 (Symmetric queue) *Queue j is referred to as a symmetric queue if the queueing discipline satisfies the following relation for feasible* $\mathbf{c}$.

$$\beta_j(l, \mathbf{c}) = \delta_j(l, \mathrm{R}_{(j,l)}(\mathbf{c})) \quad \textit{for} \quad l = 1, 2, \cdots, |\boldsymbol{c}_j|\ ,$$

where $\mathrm{R}_{(j,l)}(\boldsymbol{c}) = (\boldsymbol{c}_1, \cdots, \boldsymbol{c}_{j-1}, (\boldsymbol{c}_{j1}, \cdots, \boldsymbol{c}_{j,l-1}, \boldsymbol{c}_{j,l+1}, \cdots, \boldsymbol{c}_{jn}), \boldsymbol{c}_{j+1}, \cdots, \boldsymbol{c}_N)$.

Definition 3.2 (Local balance queue) *Queue j is referred to as a local balance queue if each service class at the queue is described by a negative exponential distribution and queueing discipline satisfies the relation*

$$\sum_{l=1}^{|\boldsymbol{c}_j|} \mu_{w_{jl}}\, \Phi(\boldsymbol{x}(\boldsymbol{c}) - \boldsymbol{e}_j(u_{jl}, v_{jl}))\, (\beta_j(l, \boldsymbol{c}) - \delta_j(l, \mathrm{R}_{(j,l)}(\boldsymbol{c})) = 0\ . \tag{3.9}$$

Note that the Definition 3.2 is a generalization of the local balance queue defined by Chandy et al. [10].

Theorem 3.1 (Product form distribution) *If each queue in the lower layer of the network is a symmetric queue or a local balance queue, then the equilibrium state probability for the network state $(\boldsymbol{c}, \boldsymbol{y})$ is given in the form*

$$P(\boldsymbol{c},\ \boldsymbol{y}) = P(\boldsymbol{c}) \prod_{j=1}^{N} P_j(\boldsymbol{y}_j | \boldsymbol{c}_j)\ , \tag{3.10}$$

where

$$P_j(\boldsymbol{y}_j | \boldsymbol{c}_j) \ = \ \prod_{l=1}^{|\boldsymbol{c}_j|} \mu_{w_{jl}} \{1 - F_{w_{jl}}(y_{jl})\}\ , \tag{3.11}$$

$$P(\boldsymbol{c}) \ = \ C\, \Lambda(\boldsymbol{m}(\boldsymbol{c}))\, \Phi(\boldsymbol{x}(\boldsymbol{c})) \prod_{j=1}^{N} \tau_j(\boldsymbol{c}_j)\ , \tag{3.12}$$

and C is a normalization constant.

See Appendix 1 for the proof of the theorem.

4. Marginal distribution and throughput

Marginal distribution for network occupancy $\boldsymbol{x}$: The product form distribution (3.10) for the equilibrium network state has more information than is usually required for performance evaluation in actual applications. For convenience, therefore, marginal distribution of the product form distribution is derived with respect to the aggregate network states. The marginal distribution for the network occupancy $\boldsymbol{c}$ is derived from equations (3.10) and (3.11) in the form

$$\int_0^\infty P(\boldsymbol{c}, \boldsymbol{y}) d\boldsymbol{y} = P(\boldsymbol{c}) . \tag{4.1}$$

Let $x_{j(m,i)}$ be an aggregate number of type-m and station-i customers at queue j. Denote

$$\boldsymbol{x}_{jm} = (x_{j(m,1)}, x_{j(m,2)}, \cdots, x_{j(m,R)}) , \quad \boldsymbol{x}_j = (\boldsymbol{x}_{j1}, \boldsymbol{x}_{j2}, \cdots, \boldsymbol{x}_{jM}) , \quad \boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N) ,$$

$$m_s = \sum_{j=1}^{N} \sum_{i=1}^{R} x_{j(s,i)} , \; n_i = \sum_{j=1}^{N} \sum_{m=1}^{M} x_{j(i,m)} , \; \boldsymbol{m} = (m_1, m_2, \cdots, m_M), \; \boldsymbol{n} = (n_1, n_2, \cdots, n_R)$$

and

$$\begin{aligned} P(\boldsymbol{x}) &= P(\boldsymbol{x}(\boldsymbol{c}) = \boldsymbol{x}) \\ &= P\Big((((x_{j(m,i)}(\boldsymbol{c}_j) = x_{j(m,i)})_{i=1}^{R})_{m=1}^{M})_{j=1}^{N} \Big) . \end{aligned} \tag{4.2}$$

Aggregate traffic intensity for network occupancy $\boldsymbol{x}$: Define the aggregate traffic intensity associated with aggregate state $x_{j(m,i)}$ by

$$\rho_{j(m,i)} = \sum_{k=1}^{K} \sigma_j(m, i, k)$$

for $j \in I_N$, $m \in I_M$, $i \in I_R$. Denote

$$\boldsymbol{\rho}_{jm} = (\rho_{j(m,1)}, \rho_{j(m,2)}, \cdots, \rho_{j(m,R)}) \quad \text{and} \quad \boldsymbol{\rho}_j = (\boldsymbol{\rho}_{j1}, \boldsymbol{\rho}_{j2}, \cdots, \boldsymbol{\rho}_{jM}) .$$

Proposition 4.1 (Product form distribution for aggregate state $\boldsymbol{x}$) *The equilibrium marginal distribution for the aggregate state $\boldsymbol{x}$ is given in the form*

$$P(\boldsymbol{x}) = C \, \Lambda(\boldsymbol{m}) \, \Phi(\boldsymbol{x}) \prod_{j=1}^{N} \pi_j(\boldsymbol{x}_j) , \tag{4.3}$$

where

$$\pi_j(\boldsymbol{x}_j) = \frac{\| \boldsymbol{x}_j \| !}{\boldsymbol{x}_j !} \boldsymbol{\rho}_j^{\boldsymbol{x}_j} . \tag{4.4}$$

Proof: From the definition of the marginal distribution,

$$\begin{aligned} P(\boldsymbol{x}) &= \sum_{\boldsymbol{c} \in B(\boldsymbol{x})} P(\boldsymbol{c}) \\ &= C \Lambda(\boldsymbol{m}) \, \Phi(\boldsymbol{x}) \prod_{j=1}^{N} \{ \sum_{\boldsymbol{c}_j \in B_j(\boldsymbol{x}_j)} \prod_{l=1}^{|\boldsymbol{c}_j|} \sigma_j(c_{jl}) \; \} , \end{aligned} \tag{4.5}$$

where

$$B_j(\boldsymbol{x}_j) = \{\boldsymbol{c}_j | \boldsymbol{x}_j(\boldsymbol{c}_j) = \boldsymbol{x}_j\} \quad \text{for} \quad j \in I_N \quad \text{and} \quad B(\boldsymbol{x}) = \bigotimes_{j=1}^{N} B_j(\boldsymbol{x}_j) .$$

The statement follows by applying the relation

$$\sum_{\boldsymbol{c}_j \in B_j(\boldsymbol{x}_j)} \prod_{l=1}^{|\boldsymbol{c}_j|} \sigma_j(\boldsymbol{c}_{jl}) = \pi_j(\boldsymbol{x}_j)$$

to equation (4.5). ▮

Throughput for network occupancy $\boldsymbol{x}$: Let $\varphi_{j(m,i)}(\boldsymbol{x})$ be the rate, for the aggregate network occupancy $\boldsymbol{x}$, with which a type-m and station-i customer at queue j moves from the lower layer to the upper layer or from the upper layer to the lower layer. That is,

$$\varphi_{j(m,i)}(\boldsymbol{x}) = \sum_{\boldsymbol{c} \in B(\boldsymbol{x})} \sum_{l=1}^{|\boldsymbol{c}_j|} P(\boldsymbol{c})\, \mu_{w_{jl}}\, \gamma_j(l, \boldsymbol{c})\, r_{(j,w_{jl}),*}(v_{jl}) \mathrm{I}(u_{jl} = m, v_{jl} = i) . \tag{4.6}$$

Assumption 4.1 *For $j \in I_N$ and for $l = 1, 2, \cdots |\boldsymbol{c}_j|$, the $\beta_j(j, \boldsymbol{c})$ depends only on type and station indexes (u_{jl}, v_{jl}) of a customer at position l in queue j, that is,*

$$\beta_j(l, \boldsymbol{c}) = \beta_j(u_{jl}, v_{jl}) . \tag{4.7}$$

Let

$$w_{j(i)} = \sum_{k=1}^{K} w_{(j,k)}(i) r_{(j,k),*}(i) .$$

Proposition 4.2 (Throughput for aggregate state $\boldsymbol{x}$) *Under the assumption 4.1, the $\varphi_{j(m,i)}(\boldsymbol{x})$ is given in the form*

$$\varphi_{j(m,i)}(\boldsymbol{x}) = C\, v_i(m) w_{j(i)} \Lambda(\boldsymbol{m}) \Phi(\boldsymbol{x} - \boldsymbol{e}_j(m,i)) \pi_j(\boldsymbol{x}_j - \boldsymbol{e}_j(m,i)) \prod_{\substack{s=1 \\ s \neq j}}^{N} \pi_s(\boldsymbol{x}_s) . \tag{4.8}$$

Proof: From the definition (4.6), product form distribution (3.12), and assumption (4.1)

$$\begin{aligned}
\varphi_{j(m,i)}(\boldsymbol{x}) &= \sum_{\boldsymbol{c} \in B(\boldsymbol{x})} C\Lambda(\boldsymbol{m}(\boldsymbol{c})) \prod_{\substack{s=1 \\ s \neq j}}^{N} \tau_s(\boldsymbol{c}_s) \sum_{l=1}^{|\boldsymbol{c}_j|} \Phi(\boldsymbol{x}(\boldsymbol{c}) - \boldsymbol{e}_j(u_{jl}, v_{jl})) \beta_j(u_{jl}, v_{jl}) \\
&\quad \cdot \ \theta_j(u_{jl}, v_{jl}, w_{jl}) r_{(j,w_{jl}),*}(i) \prod_{\substack{s=1 \\ s \neq j}}^{|\boldsymbol{c}_j|} \sigma_s(\boldsymbol{c}_{js}) \mathrm{I}(u_{jl} = m, v_{jl} = i) \\
&= C\Lambda(\boldsymbol{m}) \prod_{\substack{s=1 \\ s \neq j}}^{N} \pi_s(\boldsymbol{x}_s) \sum_{l=1}^{|\boldsymbol{c}_j|} \Phi(\boldsymbol{x} - \boldsymbol{e}_j(m,i)) \beta_j(u_{jl}, v_{jl}) \\
&\quad \cdot \ \sum_{w_{jl}=1}^{K} \theta_j(m, i, w_{jl}) r_{(j,w_{jl}),*}(i) \pi_j(\boldsymbol{x}_j - \boldsymbol{e}_j(m,i)) .
\end{aligned} \tag{4.9}$$

Using the proposition 3.1, we can obtain (4.8) after some calculation. ▮

Marginal distribution for aggregate state z and station occupancy n: Let

$$z_{ji} = \sum_{m=1}^{M} x_{j(m,i)} , \quad \boldsymbol{z}_j = (z_{j1}, z_{j2}, \cdots, z_{jR}), \quad \boldsymbol{z} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_N) .$$

Note that

$$\boldsymbol{z}_j = \boldsymbol{x}_{j1} + \boldsymbol{x}_{j2} + \cdots + \boldsymbol{x}_{jM} , \quad \|\boldsymbol{x}_j\| = \|\boldsymbol{z}_j\| \quad \text{and} \quad \boldsymbol{n} = \boldsymbol{z}_1 + \boldsymbol{z}_2 \cdots + \boldsymbol{z}_N .$$

We assume following two additional assumptions to derive marginal distribution associated with the aggregate state z_{ji}.

Assumption 4.2 *The arrival function $\Lambda(\boldsymbol{m})$ can be written in the form*

$$\Lambda(\boldsymbol{m}) = \prod_{s=1}^{M} \Lambda_s(m_s) \quad \text{and} \quad \Lambda_s(m_s) = \lambda_s^{m_s} . \tag{4.10}$$

If a routing subchain of a type-s customer in the upper layer is a closed subchain, then assume that $\lambda_s = 1$ for the given population m_s of the subchain s; otherwise assume that $\lambda_s = 0$.

Assumption 4.3 *The service rate function Φ depends only on variables $\boldsymbol{z}$ which have no information associated with types of customers in the network. We rewrite the service rate function by Ψ, that is*

$$\Phi(\boldsymbol{x}) \Rightarrow \Phi(\boldsymbol{x}_{11} + \cdots + \boldsymbol{x}_{1M}, \cdots, \boldsymbol{x}_{N1} + \cdots + \boldsymbol{x}_{NM}) = \Psi(\boldsymbol{z}) .$$

Define the aggregate traffic intensities associated with aggregate state z_{ji} and $\boldsymbol{z}_j$ by

$$a_{ji} = \sum_{m=1}^{M} \lambda_m \rho_{j(m,i)} \quad \text{and} \quad \boldsymbol{a}_j = (a_{j1},\, a_{j2},\, \cdots,\, a_{jR}) \ , j \in I_N .$$

Define

$$\nu_j(\boldsymbol{z}_j) = \frac{\| \boldsymbol{z}_j \| !}{\boldsymbol{z}_j\,!} \boldsymbol{a}_j^{\boldsymbol{z}_j} , \quad j \in I_N .$$

Proposition 4.3 (Marginal distribution for aggregate state $\boldsymbol{z}$) *Under the assumptions 4.1, 4.2, and 4.3, the equilibrium marginal distribution for the aggregate state $\boldsymbol{z}$ is given in the form*

$$P(\boldsymbol{z}) = C\Psi(\boldsymbol{z}) \prod_{j=1}^{N} \nu_j(\boldsymbol{z}_j) . \tag{4.11}$$

Proof: Let $H_j(\boldsymbol{z}_j) = \{\boldsymbol{x}_j \mid \boldsymbol{x}_{j1} + \boldsymbol{x}_{j2} + \cdots + \boldsymbol{x}_{jM} = \boldsymbol{z}_j\}$ and $H(\boldsymbol{z}) = \otimes_{j=1}^{N} H_j(\boldsymbol{x}_j)$. Then

$$\begin{aligned} P(\boldsymbol{z}) &= \sum_{\boldsymbol{x} \in H(\boldsymbol{z})} P(\boldsymbol{x}) = C \sum_{\boldsymbol{x} \in H(\boldsymbol{z})} \Psi(\boldsymbol{x}) \prod_{j=1}^{N} \prod_{m=1}^{M} \prod_{i=1}^{R} \frac{(\lambda_m \rho_{j(m,i)})^{x_{j(m,i)}}}{x_{j(m,i)}\,!} \\ &= C\Psi(\boldsymbol{z}) \prod_{j=1}^{N} \frac{\| \boldsymbol{z}_j \| !}{\boldsymbol{z}_j\,!} \boldsymbol{a}_j^{\boldsymbol{z}_j} \qquad \blacksquare \end{aligned}$$

Proposition 4.4 (Marginal distribution for station occupancy $\boldsymbol{n}$) *Under the assumptions 4.1, 4.2, and 4.3, the marginal distribution for station occupancy $\boldsymbol{n}$ is given in the form*

$$P(\boldsymbol{n}) = C\, G(\boldsymbol{n}) \,, \tag{4.12}$$

where

$$G(\boldsymbol{n}) = \sum_{\boldsymbol{z} \in K(\boldsymbol{n})} \Psi(\boldsymbol{z}) \prod_{j=1}^{N} \nu_j(\boldsymbol{z}_j) \tag{4.13}$$

and $K(\boldsymbol{n}) = \{\boldsymbol{z} | \boldsymbol{z}_1 + \boldsymbol{z}_2 + \cdots + \boldsymbol{z}_N = \boldsymbol{n}\}$.

Proof: Summation $\sum_{\boldsymbol{z} \in K(\boldsymbol{n})} P(\boldsymbol{z})$ yields the statement. ∎

Note that the $G(\boldsymbol{n})$ is equivalent to the normalization constant of the closed network with the population vector $\boldsymbol{n}$.

Throughput for aggregate state $\boldsymbol{z}$: Let $\varphi^{*}_{j(m,i)}(\boldsymbol{z})$ be the rate, given the aggregate network occupancy $\boldsymbol{z}$, with which a type-m and station-i customer at queue j moves from the lower layer to the upper layer or from the upper layer to the lower layer.

Proposition 4.5 (Throughput for aggregate state $\boldsymbol{z}$) *Under the assumptions 4.1, 4.2, and 4.3, $\varphi_{j(m,i)}(\boldsymbol{z})$ is given in the form*

$$\varphi^{*}_{j(m,i)}(\boldsymbol{z}) = C \lambda_m v_i(m) w_{j(i)} \Psi(\boldsymbol{z} - \boldsymbol{e}_j(i)) \nu_j(\boldsymbol{z}_j - \boldsymbol{e}_i) \prod_{\substack{s=1 \\ s \neq j}}^{N} \nu_s(\boldsymbol{z}_s)$$

Proof: Calculation for $\varphi^{*}_{j(m,i)}(\boldsymbol{z}) = \sum_{\boldsymbol{x} \in H(\boldsymbol{z})} \varphi_{j(m,i)}(\boldsymbol{x})$ yields the statement. ∎

Throughput for station occupancy $\boldsymbol{n}$: For station occupancy $\boldsymbol{n}$, define $\varphi^{**}_{j(m,i)}(\boldsymbol{n})$ as the throughput for a type-m and station-i customer to move from the lower layer to the upper layer at queue j, and define $\varphi^{***}_{(i)}(\boldsymbol{n})$ as the throughput for a station-i customer to move from the lower layer to the upper layer. That is,

$$\varphi^{**}_{j(m,i)}(\boldsymbol{n}) = \sum_{\boldsymbol{z} \in K(\boldsymbol{n})} \varphi^{*}_{j(m,i)}(\boldsymbol{z}) \quad \text{and} \quad \varphi^{***}_{(i)}(\boldsymbol{n}) = \sum_{j=1}^{N} \sum_{m=1}^{M} \varphi^{**}_{j(m,i)}(\boldsymbol{n}) \,.$$

Let

$$\eta_i = \sum_{m=1}^{M} \lambda_m v_i(m) \,.$$

Proposition 4.6 (Throughput for station occupancy $\boldsymbol{n}$) *Under the assumptions 4.1, 4.2, and 4.3,*

$$\varphi^{**}_{j(m,i)}(\boldsymbol{n}) = C \lambda_m v_i(m) w_{j(i)} G(\boldsymbol{n} - \boldsymbol{e}_i) \tag{4.14}$$

and

$$\varphi^{***}_{(i)}(\boldsymbol{n}) = C\, \eta_i\, G(\boldsymbol{n} - \boldsymbol{e}_i) \,. \tag{4.15}$$

Proof: This formulas can be derived by straightforward calculation. ∎

Let $\mu_i(\boldsymbol{n})$ be the throughput for a station-i customer to move from the lower layer to the upper layer or from the upper layer to the lower layer.

Proposition 4.7 (Throughput at station i given station occupancy $\boldsymbol{n}$) *Under the assumptions 4.1, 4.2, and 4.3,*

$$\mu_i(\boldsymbol{n}) = \eta_i \frac{G(\boldsymbol{n} - \boldsymbol{e}_i)}{G(\boldsymbol{n})} . \tag{4.16}$$

Proof: Since $\varphi^{***}_{(i)}(\boldsymbol{n}) = \mu_i(\boldsymbol{n})P(\boldsymbol{n})$, the equation (4.16) is derived directly from equations (4.12) and (4.15). ∎

5. Two-layer networks with outstanding queues

Model description: We have so far assumed that each station in the upper layer consists of infinite servers for the two layer networks without outstanding queues. In this section, we instead assume that each station in the upper layer has a capacity constraint for the two-layer networks with outstanding queues. In the networks with outstanding queues, if a station has a constraint on capacity, then a number of customers less than or equal to the constraint number may enter the station and travel through queues in the lower layer according to the routing chain associated with the station. Customers in excess of the constraint number have to wait in the outstanding queue in front of the station until the number of customers in the station become less than the constraint number of the station. Formally, when k_i $(0 < k_i < \infty)$ is a capacity constraint at station i $(i \in I_R)$, a number of customers less than or equal to k_i may travel through the queues in the lower layer according to routing chain $P(i)$. Customers in excess k_i have to wait in the outstanding queue i associated with station i until the number of traveling customers becomes less than k_i. The two-layer queueing networks with outstanding queues do not have the product form distribution which is available in the case of networks without outstanding queues. Approximate techniques are required to derive characteristics for the networks with outstanding queues. Figure 3 shows an example of a two-layer network with outstanding queues.

Approximation technique for the networks with outstanding queues:

Let $\boldsymbol{k} = (k_1, k_2, \cdots, k_R)$. Denote $\boldsymbol{x}^+ = (\max(0, x_1), \max(0, x_2), \cdots, \max(0, x_n))$ and $\boldsymbol{x} \wedge \boldsymbol{y} = (\min(x_1, y_1), \min(x_2, y_2), \cdots, \min(x_n, y_n),)$.

In case of the networks without outstanding queues, one may consider that station i behaves as if there was an infinite queue with state-dependent service rate $\mu_i(\boldsymbol{n})$ in the upper layer, and the $\mu_i(\boldsymbol{n})$ is equivalent to the throughput at station i between the upper layer and the lower layer where $\mu_i(\boldsymbol{n})$ is given in Proposition 4.7. In case of the networks with outstanding queues, on the other hand, one may consider that station i behaves as if there were an queue with a number k_i of servers in the upper layer. We cannot, however, obtain the service rate at the station in a closed form. If we could obtain service rate at each station in the upper layer, then we could obtain approximate solution for the network with outstanding queues by using the standard Markovian technique.

A key observation for the approximation of the service rate is as follows. If station occupancy is given by $\boldsymbol{n}$, then one may find that a population $(\boldsymbol{n} \wedge \boldsymbol{k})$ of customers are traveling through the lower layer and a population $(\boldsymbol{n} - \boldsymbol{k})^+$ of customers are waiting in

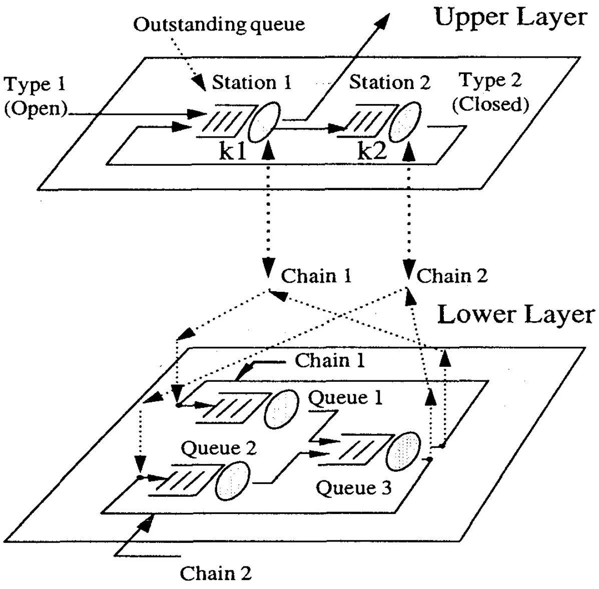

Figure 3: Two-layer network with outstanding queues.

outstanding queues in the upper layer layer. Thus the approximate throughput $\mu_i^*(\boldsymbol{n})$ at station i may, given the station occupancy $\boldsymbol{n}$, be written in the form

$$\mu_i^*(\boldsymbol{n}) = \eta_i \frac{G(\boldsymbol{n} \wedge \boldsymbol{k} - \boldsymbol{e}_i)}{G(\boldsymbol{n} \wedge \boldsymbol{k})}, \tag{5.1}$$

where $G(\boldsymbol{n})$ is given in the equation (4.13). Assuming that the service-time distribution of station i ($i \in I_R$) is an exponential distribution with rate $\mu_i^*(\boldsymbol{n})$, we can formulate balance equations with respect to state $\boldsymbol{n}$ and obtain approximate solution by solving the equations.

6. Conclusion

This paper has proposed a two-layer queueing network paradigm for the performance evaluation of computer and communication systems. The main result establishes a product form solution for the case in which the upper layer involves no outstanding queue, generalizing the standard results in queueing networks. Marginal distributions have been derived for convenience in practical applications. For the case in which the upper layer dose include outstanding queues, an approximate method is proposed, which generalizes the standard flow-equivalent methods. The quantitative evaluation of the precision of the proposed approximate method is remained for the future study. The proposed approximation technique requires that a system of equations be solved, with is formulated regarding the upper layer configuration under the Markovian assumptions. As the number of states in the upper layer increases, the system of equations quickly becomes intractable. Thus another immediate research direction would be on how to deal with the explosion of the number of states in the system of equations.

Acknowledgment

The author expresses his sincere thanks to two anonymous referees whose comments were valuable in revising the original manuscript.

Appendix 1. Proof of Theorem 3.1

Insert operator and remove operator: For description of transition caused by arrivals and departures of customers, the following operators are defined.

Remove operators $\mathrm{R}_{(j,l)}(\boldsymbol{c})$ and $\mathrm{R}_{(j,l)}(\boldsymbol{y})$ are defined by

$$\mathrm{R}_{(j,l)}(\boldsymbol{c}) = (\boldsymbol{c}_1, \cdots, \boldsymbol{c}_{j-1}, (\boldsymbol{c}_{j1}, \cdots, \boldsymbol{c}_{j,l-1}, \boldsymbol{c}_{j,l+1}, \cdots, \boldsymbol{c}_{jn}), \boldsymbol{c}_{j+1}, \cdots, \boldsymbol{c}_N)$$

$$\text{and} \quad \mathrm{R}_{(j,l)}(\boldsymbol{y}) = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_{j-1}, (\boldsymbol{y}_{j1}, \cdots, \boldsymbol{y}_{j,l-1}, \boldsymbol{y}_{j,l+1}, \cdots, \boldsymbol{y}_{jn}), \boldsymbol{y}_{j+1}, \cdots, \boldsymbol{y}_N) .$$

Insert operator $\mathrm{I}_{(j,l)[m,i,k]}(\boldsymbol{c})$ inserts an element of (m,i,k) at the position (j,l) which corresponds to the position l at queue j in the vector $\boldsymbol{c}$, and insert operator $\mathrm{I}_{(j,l)[u]}(\boldsymbol{y})$ inserts an element u at the position (j,l) of vector $\boldsymbol{y}$. That is,

$$\mathrm{I}_{(j,l)[m,i,k]}(\boldsymbol{c}) = (\boldsymbol{c}_1, \cdots, \boldsymbol{c}_{j-1}, (\boldsymbol{c}_{j1}, \cdots, \boldsymbol{c}_{j,l-1}, (m,i,k), \boldsymbol{c}_{jl}, \cdots, \boldsymbol{c}_{jn}), \boldsymbol{c}_{j+1}, \cdots, \boldsymbol{c}_N)$$

$$\text{and} \quad \mathrm{I}_{(j,l)[u]}(\boldsymbol{y}) = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_{j-1}, (\boldsymbol{y}_{j1}, \cdots, \boldsymbol{y}_{j,l-1}, u, \boldsymbol{y}_{jl}, \cdots, \boldsymbol{y}_{jn}), \boldsymbol{y}_{j+1}, \cdots, \boldsymbol{y}_N) .$$

Global balance equation: Let $P_t(\boldsymbol{c},\boldsymbol{y})$ be the state probability density function of the network at time t. The balance equation can be obtained by describing the probability $P_{t+dt}(\boldsymbol{c},\boldsymbol{y})$ in terms of $P_t(\boldsymbol{c},\boldsymbol{y})$, functions of $(\boldsymbol{c},\boldsymbol{y})$, and dt; applying the definition of the derivative; and equating the result to zero.

Although each supplementary random variable y_{jl} is strictly greater than zero, the description $y_{jl} = 0$ is used in this paper to describe the state of a customer whose service is complete and thus is ready to depart. Define a service rate vector in the form

$$\xi(\boldsymbol{c}) = ((\gamma_j(l, \boldsymbol{c}))_{l=1}^{|\boldsymbol{c}_j|})_{j=1}^{N} .$$

We write $\xi_{(g,h)}(\boldsymbol{c})$ to denote the associated service rate vector in which the element $\gamma_g(h,\boldsymbol{c})$ is replaced by 0, that is,

$$\xi_{(g,h)}(\boldsymbol{c}) = (\gamma_1^*(\boldsymbol{c}), \cdots, \gamma_{g-1}^*(\boldsymbol{c}),\ \gamma_{(g,h)}^*(\boldsymbol{c})\ , \gamma_{g+1}^*(\boldsymbol{c}), \cdots, \gamma_N^*(\boldsymbol{c}))$$

where $\gamma_j^*(\boldsymbol{c}) = (\gamma_j(l,\boldsymbol{c}))_{l=1}^{|\boldsymbol{c}_j|}$ for $j \neq g$, and

$$\gamma_{(g,h)}^*(\boldsymbol{c}) = (\gamma_g(1,\boldsymbol{c}), \cdots, \gamma_g(h-1,\boldsymbol{c}),\ 0\ , \gamma_g(h+1,\boldsymbol{c}), \cdots, \gamma_g(|\boldsymbol{c}_g|,\boldsymbol{c})) .$$

The standard technique of the supplementary variable method yields

$$\begin{aligned} P_{t+dt}(\boldsymbol{c},\boldsymbol{y}) &= P_t(\ \boldsymbol{c},\ \boldsymbol{y} + \xi(\boldsymbol{c})dt)\,(1-\lambda(\boldsymbol{c})dt) \\ &\quad + \sum_{j=1}^{N}\sum_{l=1}^{|\boldsymbol{c}_j|}\sum_{g=1}^{N}\sum_{h=1}^{|\boldsymbol{c}_g|+1}\sum_{k=1}^{K}\int_0^{\gamma_g(h,\boldsymbol{c}_L)dt} P_t(\ \boldsymbol{c}_L, \boldsymbol{y}_L(u) + \xi_L\, dt\,)\, du \end{aligned}$$

$$
\begin{aligned}
&\cdot \quad r_{(g,k),(j,w_{jl})}(v_{jl}) \cdot \delta_j(l,\, \boldsymbol{c}_A) \cdot f_{w_{jl}}(y_{jl}) \cdot (1 - \lambda(\boldsymbol{c}_L)dt) \\
&+ \quad \sum_{j=1}^{N} \sum_{l=1}^{|\boldsymbol{C}_j|} \sum_{g=1}^{N} \sum_{h=1}^{|\boldsymbol{C}_g|+1} \sum_{k=1}^{K} \sum_{i=1}^{R} \int_0^{\gamma_g(h,\boldsymbol{C}_U)dt} P_t(\, \boldsymbol{c}_U,\, \boldsymbol{y}_U(u) + \xi_U\, dt\,)\, du \\
&\cdot \quad r_{(g,k),*}(i) \cdot s_{i,v_{jl}}(u_{jl}) \cdot r_{*,(j,w_{jl})}(v_{jl}) \cdot \delta_j(l,\, \boldsymbol{c}_A) \\
&\cdot \quad f_{w_{jl}}(y_{jl}) \cdot (1 - \lambda(\boldsymbol{c}_U)dt) \\
&+ \quad \sum_{j=1}^{N} \sum_{l=1}^{|\boldsymbol{C}_j|} P_t(\, \boldsymbol{c}_A,\, \boldsymbol{y}_A + \xi_A\, dt\,) \cdot \lambda_{u_{jl}}(\boldsymbol{c}_A)\, dt \\
&\cdot \quad s_{0,v_{jl}}(u_{jl}) \cdot r_{*,(j,w_{jl})}(v_{jl}) \cdot \delta_j(l,\, \boldsymbol{c}_A) \cdot f_{w_{jl}}(y_{jl}) \\
&+ \quad \sum_{g=1}^{N} \sum_{h=1}^{|\boldsymbol{C}_g|+1} \sum_{k=1}^{K} \sum_{i=1}^{R} \sum_{m=1}^{M} \int_0^{\gamma_g(h,\boldsymbol{C}_D)dt} P_t(\, \boldsymbol{c}_D,\, \boldsymbol{y}_D(u) + \xi_D\, dt\,)\, du \\
&\cdot \quad r_{(g,k),*}(i) \cdot s_{i0}(m) \cdot (1 - \lambda(\boldsymbol{c}_D)dt) \;+ o(dt)\,, \qquad \text{(a.1)}
\end{aligned}
$$

where

$$\xi_L = \xi_U = \xi_{(g,h)}(\mathrm{R}_{(j,l)}(\boldsymbol{c}))\,,\ \xi_D = \xi_{(g,h)}(\boldsymbol{c})\,,\ \xi_A = \xi(\mathrm{R}_{(j,l)}(\boldsymbol{c}))\,, \quad \boldsymbol{c}_A = \mathrm{R}_{(j,l)}(\boldsymbol{c})\,,$$

$$\boldsymbol{c}_L = \mathrm{I}_{(g,h)[u_{jl},\, v_{jl},\, k]}(\mathrm{R}_{(j,l)}(\boldsymbol{c}))\,, \quad \boldsymbol{c}_U = \mathrm{I}_{(g,h)[u_{jl},\, i,\, k]}(\mathrm{R}_{(j,l)}(\boldsymbol{c}))\,, \quad \boldsymbol{c}_D = \mathrm{I}_{(g,h)[m,\, i,\, k]}(\boldsymbol{c})\,,$$

$$\boldsymbol{y}_D(u) = \mathrm{I}_{(g,h)[u]}(\boldsymbol{y})\,, \quad \boldsymbol{y}_A = \mathrm{R}_{(j,l)}(\boldsymbol{y})\,, \quad \text{and} \quad \boldsymbol{y}_U(u) = \boldsymbol{y}_L(u) = \mathrm{I}_{(g,h)[u]}(\mathrm{R}_{(j,l)}(\boldsymbol{y}))\,.$$

The first term on the right-hand side of equation (a.1) corresponds to the event that there is no arrival and no departure in the time interval dt . The second term expresses the internal transition of customers without change of stations. The third term expresses the transition of customers with change of stations. The fourth term is the contribution due to the arrival of a customer from outside the network. The last term implies the event that a customer depart from the network to outside eventually.

For each integral factor in equation (a.1), expanding the integrand in a Taylor series around $u = 0$ and carrying out the integration yields

$$
\begin{aligned}
\int_0^{\gamma_g(h,\boldsymbol{C}_L)dt} P_t(\, \boldsymbol{c}_L, \boldsymbol{y}_L(u) + \xi_L\, dt\,)\, du &= \gamma_g(h, \boldsymbol{c}_L) P_t(\boldsymbol{c}_L\,, \boldsymbol{y}_L(0))dt + o(dt)\,, \\
\int_0^{\gamma_g(h,\boldsymbol{C}_U)dt} P_t(\, \boldsymbol{c}_U,\, \boldsymbol{y}_U(u) + \xi_U\, dt\,)\, du &= \gamma_g(h, \boldsymbol{c}_U) P_t(\boldsymbol{c}_U,\, \boldsymbol{y}_U(0))dt + o(dt)\,, \\
\int_0^{\gamma_g(h,\boldsymbol{C}_D)dt} P_t(\, \boldsymbol{c}_D,\, \boldsymbol{y}_D(u) + \xi_D\, dt\,)\, du &= \gamma_g(h, \boldsymbol{c}_D) P_t(\boldsymbol{c}_D,\, \boldsymbol{y}_D(0))dt + o(dt)\,.
\end{aligned}
$$

Similarly, the Taylor expansion yields

$$
\begin{aligned}
P_t(\, \boldsymbol{c},\, \boldsymbol{y} + \xi(\boldsymbol{c})dt) &= P_t(\boldsymbol{c},\, \boldsymbol{y}) + (\xi(\boldsymbol{c}), J_t(\boldsymbol{c}, \boldsymbol{y}))dt + o(dt), \\
P_t(\, \boldsymbol{c}_A,\, \boldsymbol{y}_A + \xi_A\, dt\,) &= P_t(\boldsymbol{c}_A, \boldsymbol{y}_A) + (\xi_A\,, J_t(\boldsymbol{c}_A, \boldsymbol{y}_A))dt + o(dt)
\end{aligned}
$$

where $J_t(\boldsymbol{c}, \boldsymbol{y})$ is given by

$$J_t(\boldsymbol{c}, \boldsymbol{y}) = (\,(\,\frac{\partial P_t(\boldsymbol{c}, \boldsymbol{y})}{\partial y_{jl}}\,)_{l=1}^{|\boldsymbol{C}_j|}\,)_{j=1}^{N}\,.$$

Substituting these relations into equation (a.1), applying the definition of the derivative, equating the derivative to zero, dropping the time parameter, and rearranging terms yields the global balance equation:

$$\begin{aligned}
\lambda(\boldsymbol{c})P(\boldsymbol{c},\boldsymbol{y}) &= (\xi(\boldsymbol{c}), J(\boldsymbol{c},\boldsymbol{y})) \\
&+ \sum_{j=1}^{N}\sum_{l=1}^{|\boldsymbol{C}_j|}\sum_{g=1}^{N}\sum_{h=1}^{|\boldsymbol{C}_g|+1}\sum_{k=1}^{K}\gamma_g(h,\boldsymbol{c}_L)P(\,\boldsymbol{c}_L,\boldsymbol{y}_L(0)) \\
&\cdot\ r_{(g,k),(j,w_{jl})}(\,v_{jl}\,)\cdot\delta_j(\,l,\ \boldsymbol{c}_A\,)\cdot f_{w_{jl}}(\,y_{jl}\,) \\
&+ \sum_{j=1}^{N}\sum_{l=1}^{|\boldsymbol{C}_j|}\sum_{g=1}^{N}\sum_{h=1}^{|\boldsymbol{C}_g|+1}\sum_{k=1}^{K}\sum_{i=1}^{R}\gamma_g(h,\boldsymbol{c}_U)P(\,\boldsymbol{c}_U,\ \boldsymbol{y}_U(0)) \\
&\cdot\ r_{(g,k),*}(i)\cdot s_{i,v_{jl}}(u_{jl})\cdot r_{*,(j,w_{jl})}(\,v_{jl}\,)\ \cdot\delta_j(l,\ \boldsymbol{c}_A)\cdot f_{w_{jl}}(\,y_{jl}\,) \\
&+ \sum_{j=1}^{N}\sum_{l=1}^{|\boldsymbol{C}_j|} P(\,\boldsymbol{c}_A,\ \boldsymbol{y}_A)\lambda_{u_{jl}}(\boldsymbol{c}_A) \\
&\cdot\ s_{0,v_{jl}}(u_{jl})\cdot\ r_{*,(j,w_{jl})}(v_{jl})\ \cdot\delta_j(l,\ \boldsymbol{c}_A)\cdot f_{w_{jl}}(\,y_{jl}\,) \\
&+ \sum_{g=1}^{N}\sum_{h=1}^{|\boldsymbol{C}_g|+1}\sum_{k=1}^{K}\sum_{i=1}^{R}\sum_{m=1}^{M}\gamma_g(h,\boldsymbol{c}_D)P(\,\boldsymbol{c}_D,\ \boldsymbol{y}_D(0)) \\
&\cdot\ r_{(g,k),*}(i)\cdot s_{i\,0}(m)\ . \qquad\qquad (a.2)
\end{aligned}$$

Substituting the product form of equations (3.10), (3.11) , and (3.12) into the right-hand side of equation (a.2) , and using the relation given in the traffic equation (3.2) and the following relations

$$\boldsymbol{x}(\boldsymbol{c}_L) = \boldsymbol{x}(\boldsymbol{c}) + \boldsymbol{e}_g(u_{jl},v_{jl}) - \boldsymbol{e}_j(u_{jl},v_{jl})\,,\quad \boldsymbol{x}(\boldsymbol{c}_U) = \boldsymbol{x}(\boldsymbol{c}) + \boldsymbol{e}_g(u_{jl},i) - \boldsymbol{e}_j(u_{jl},v_{jl})\,,$$

$$\boldsymbol{x}(\boldsymbol{c}_A) = \boldsymbol{x}(\boldsymbol{c}) - \boldsymbol{e}_j(u_{jl},v_{jl})\,,\quad \text{and}\quad \boldsymbol{x}(\boldsymbol{c}_D) = \boldsymbol{x}(\boldsymbol{c}) + \boldsymbol{e}_g(m,i)\,,$$

we find that carrying out the calculation yields that

$$\begin{aligned}
&\text{the right hand-side of equation (a.2)} \\
&= P(\boldsymbol{c},\boldsymbol{y})\Big[\lambda(\boldsymbol{c}) - \sum_{j=1}^{N}\sum_{l=1}^{|\boldsymbol{C}_j|}\frac{\Phi(\boldsymbol{x}(\boldsymbol{c}) - \boldsymbol{e}_j(u_{jl},v_{jl}))}{\Phi(\boldsymbol{x}(\boldsymbol{c}))}\frac{f_{w_{jl}}(y_{jl})}{1-F_{w_{jl}}(y_{jl})}\{\beta_j(l,\boldsymbol{c}) - \delta_j(l,\boldsymbol{c}_A)\}\Big]. \\
&\qquad\qquad (a.3)
\end{aligned}$$

Since the global equation (a.2) is a linear equation system, the product form distribution (3.10) is a unique solution of equation (a.2), if it exists. One can verify that if each queue in the network is either a symmetric queue or a local balance queue, then the value of (a.3) is equal to the left-hand side of the equation. Consequently, the global balance equation (a.2) has a unique solution of product form (3.10). ▮

Appendix 2. Notations

R: the number of stations in the upper layer

N: the number of queues in the lower layer

M: the number of customer types

G_m : the set of stations that a type-m customer may visit

$s_{0i}(m)$: the probability for a type-m customer to join station i when he arrives at the network

$s_{ij}(m)$: the probability for a type-m customer to join station j after completion of his service time at station i

$S(m)$: the routing chain of a type-m customer in the upper layer; i.e., $S(m) = \{s_{ij}(m)\}$

K: the number of classes of customers

S_k: the service time of a class-k customer

F_k, f_k: the distribution function of the S_k, density function of the S_k

$\mu_k = 1/E(S_k)$

$r_{*,(j,k)}(i)$: the probability for a station-i customer to join queue j of the lower layer as a class-k customer immediately after his arrival at station i of the upper layer

$r_{(j,k),(h,l)}(i)$; the probability for a station-i customer to join queue h as a class l customer after service time completion at queue j as a class-k customer

$P(i)$: the routing chain of station-i customer in the lower layer; i.e., $P(i) = \{r_{(j,k),(h,l)}(i)\}$

u_{jl}: the type index of a customer in position l at queue j

v_{jl}: the station index of a customer in position l at queue j

w_{jl}: the class index of a customer in position l at queue j

$\boldsymbol{c}_{jl} = (u_{jl}, v_{jl}, w_{jl})$, $\boldsymbol{c}_j = (\boldsymbol{c}_{j1}, \boldsymbol{c}_{j2}, \cdots, \boldsymbol{c}_{jn})$ where $n = |\boldsymbol{c}_j|$, and $\boldsymbol{c} = (\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_N)$

y_{jl} : the remaining service time of a customer in position l at queue j

$\boldsymbol{y}_j = (y_{j1}, y_{j2}, \cdots, y_{jn})$, $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_N)$

$\delta_j(l, \boldsymbol{c})$: the probability for a customer who finds state $\boldsymbol{c}$ at his arrival instant to choose position l at queue j where he enters.

$\gamma_j(l, \boldsymbol{c})$: the network-occupancy $(\boldsymbol{c})$-dependent service rate for a customer in position l at queue j; i.e., $\gamma_j(l, \boldsymbol{c}) = -\frac{d}{dt} y_{jl}$

$x_{j(m,i)}(\boldsymbol{c}_j)$: the total number of type-m and station- i customers at queue j given queue occupancy $\boldsymbol{c}_j$; i.e., $x_{j(m,i)}(\boldsymbol{c}_j) = \sum_{l=1}^{|\boldsymbol{c}_j|} \mathrm{I}(u_{jl} = m, v_{jl} = i)$

$\boldsymbol{x}_{jm}(\boldsymbol{c}_j) = (x_{j(m,1)}(\boldsymbol{c}_j),\ x_{j(m,2)}(\boldsymbol{c}_j),\ \cdots\ ,\ x_{j(m,R)}(\boldsymbol{c}_j))$

$\boldsymbol{x}_j(\boldsymbol{c}_j) = (\boldsymbol{x}_{j1}(\boldsymbol{c}_j),\ \boldsymbol{x}_{j2}(\boldsymbol{c}_j),\ \cdots\ ,\ \boldsymbol{x}_{jM}(\boldsymbol{c}_j))$, $\boldsymbol{x}(\boldsymbol{c}) = (\boldsymbol{x}_1(\boldsymbol{c}_1),\ \boldsymbol{x}_2(\boldsymbol{c}_2),\ \cdots,\ \boldsymbol{x}_N(\boldsymbol{c}_N))$

$z_{ji}(\boldsymbol{c}_j)$: the total number of station i customers at queue j given the queue occupancy $\boldsymbol{c}_j$, i.e $z_{ji}(\boldsymbol{c}_j) = \sum_{m=1}^{M} x_{j(m,i)}(\boldsymbol{c}_j)$

$\boldsymbol{z}_j(\boldsymbol{c}_j) = (z_{j1}(\boldsymbol{c}_j),\ z_{j2}(\boldsymbol{c}_j),\ \cdots,\ z_{jR}(\boldsymbol{c}_j))$, $\boldsymbol{z}(\boldsymbol{c}) = (\boldsymbol{z}_1(\boldsymbol{c}_1),\ \boldsymbol{z}_2(\boldsymbol{c}_2),\ \cdots,\ \boldsymbol{z}_N(\boldsymbol{c}_N))$

$m_s(\boldsymbol{c})$: the total number of type-s customers in the network given network occupancy $\boldsymbol{c}$; i.e., $m_s(\boldsymbol{c}) = \sum_{j=1}^{N} \sum_{i=1}^{R} x_{j(s,i)}(\boldsymbol{c}_j)$

$n_i(\boldsymbol{c})$: the total number of station-i customers in the network given network occupancy $\boldsymbol{c}$;

i.e., $n_i(\boldsymbol{c}) = \sum_{j=1}^{N} \sum_{m=1}^{M} x_{j(m,i)}(\boldsymbol{c}_j)$

$\boldsymbol{m}(\boldsymbol{c}) = (m_1(\boldsymbol{c}), m_2(\boldsymbol{c}), \cdots, m_M(\boldsymbol{c}))$, $\boldsymbol{n}(\boldsymbol{c}) = (n_1(\boldsymbol{c}), n_2(\boldsymbol{c}), \cdots, n_R(\boldsymbol{c}))$

$\Phi(\boldsymbol{x})$: the service rate function of variables $\boldsymbol{x}$

$\gamma_j(l, \boldsymbol{c})$: the service rate to a customer in position l at queue j given network occupancy $\boldsymbol{c}$

$\beta_j(l, \boldsymbol{c})$: the proportion of service rate to a customer in position l at queue j given queue occupancy $\boldsymbol{c}$

$\Lambda(\boldsymbol{m})$: the arrival function

$\lambda_m(\boldsymbol{c})$: the arrival rate for type-m customers from outside the network given network occupancy $\boldsymbol{c}$

$\theta_j(m, i, k)$: the relative frequency for a type-m and station-i customer visit queue j as a class-k customer

$v_i(m)$: the relative frequency for type-m customer to visit station i

$w_{(j,k)}(i)$: the relative frequency for a station-i and class-k customer to visit queue j

$\sigma_j(m, i, k) \ = \theta_j(m, i, k)/\mu_k$: the traffic intensity due to a type-m and station-i customer of class k at queue j

$\tau_j(\boldsymbol{c}_j) = \prod_{l=1}^{|\boldsymbol{c}_j|} \sigma_j(\boldsymbol{c}_{jl})$: the traffic intensity function at queue j

$x_{j(m,i)}$: the aggregate number of type-m and station-i customers at queue j

$\boldsymbol{x}_{jm} = (x_{j(m,1)}, x_{j(m,2)}, \cdots, x_{j(m,R)})$, $\boldsymbol{x}_j = (\boldsymbol{x}_{j1}, \boldsymbol{x}_{j2}, \cdots, \boldsymbol{x}_{jM})$, $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N)$

m_s: the aggregate number of type-s customers in the network; i.e., $m_s = \sum_{j=1}^{N} \sum_{i=1}^{R} x_{j(s,i)}$

n_i: the aggregate number of station-i customers in the network; i.e., $n_i = \sum_{j=1}^{N} \sum_{m=1}^{M} x_{j(m,i)}$

$\boldsymbol{m} = (m_1, m_2, \cdots, m_M)$, $\boldsymbol{n} = (n_1, n_2, \cdots, n_R)$

$\rho_{j(m,i)} = \sum_{k=1}^{K} \sigma_j(m, i, k)$: the aggregate traffic intensity for type-m and station-i customers at queue j

$\boldsymbol{\rho}_{jm} = (\rho_{j(m,1)}, \rho_{j(m,2)}, \cdots, \rho_{j(m,R)})$, $\quad \boldsymbol{\rho}_j = (\boldsymbol{\rho}_{j1}, \boldsymbol{\rho}_{j2}, \cdots, \boldsymbol{\rho}_{jM})$

$B_j(\boldsymbol{x}_j) = \{\boldsymbol{c}_j | \boldsymbol{x}_j(\boldsymbol{c}_j) = \boldsymbol{x}_j\}$, $\quad B(\boldsymbol{x}) = \bigotimes_{j=1}^{N} B_j(\boldsymbol{x}_j)$

z_{ji}: the aggregate number of station-i customers at queue j

$\boldsymbol{z}_j = (z_{j1}, z_{j2}, \cdots, z_{jR})$, $\boldsymbol{z} = (\boldsymbol{z}_1, , \boldsymbol{z}_2, \cdots, \boldsymbol{z}_N)$

$\Psi(\boldsymbol{z})$: the service rate function of variables $\boldsymbol{z}$

$a_{ji} = \sum_{m=1}^{M} \lambda_m \rho_{j(m,i)}$, $\boldsymbol{a}_j = (a_{j1}, a_{j2}, \cdots, a_{jR})$

$\nu_j(\boldsymbol{z}_j) = \frac{\|\boldsymbol{z}_j\|!}{\boldsymbol{z}_j!} \boldsymbol{a}_j^{\boldsymbol{z}_j}$

$H_j(\boldsymbol{z}_j) = \{ \boldsymbol{x}_j \mid \boldsymbol{x}_{j1} + \boldsymbol{x}_{j2} + \cdots + \boldsymbol{x}_{jM} = \boldsymbol{z}_j\}$, $\quad H(\boldsymbol{z}) = \bigotimes_{j=1}^{N} H_j(\boldsymbol{z}_j)$

$K(\boldsymbol{n}) = \{\boldsymbol{z} \mid \boldsymbol{z}_1 + \boldsymbol{z}_2 + \cdots + \boldsymbol{z}_N = \boldsymbol{n}\}$

$w_{j(i)} = \sum_{k=1}^{K} w_{(j,k)}(i) r_{(j,k),*}(i)$

$\eta_i = \sum_{m=1}^{M} \lambda_m v_i(m)$

$\varphi_{j(m,i)}(\boldsymbol{x})$, $\varphi^{*}_{j(m,i)}(\boldsymbol{z})$, $\varphi^{**}_{j(m,i)}(\boldsymbol{n})$: the throughput for a type-m and station-i customer at queue j to move from the lower layer to the upper layer at states $\boldsymbol{x}$, $\boldsymbol{z}$, and $\boldsymbol{n}$

$\varphi^{***}_{(i)}(\boldsymbol{n})$: the throughput for a station-i customer to move from the lower layer to the upper layer at station occupancy $\boldsymbol{n}$

$\mu_i(\boldsymbol{n})$: the throughput for station-i customer to move from the lower layer to the upper layer given station occupancy $\boldsymbol{n}$

References

[1] S. Balsamo and G. Iazeolla, An extension of Norton's theorem for queueing networks, *IEEE Tras.Software Eng.* **SE-8** (1982) 298-305.

[2] F. Baskett, K.M. Chandy, R.R Muntz and F.G. Palacios, Open, closed, and mixed networks of queues with different classes of customers, *J.ACM* **22** (1975) 248-260.

[3] B. Baynat and Y. Dallery, A unified view of product-form approximation techniques for general queueing networks, *Perf. Eval.* **18** (1993) 205-224.

[4] R.J. Boucherie and N.M. Van Dijk, Product forms for queueing networks with state-dependent multiple job transitions, *Adv. Appl. Prob.* **23** (1991) 152-187.

[5] R.J. Boucherie and N.M Van Dijk, A generalization of Norton's theorem for queueing networks, *Queueing Systems* **13** (1993) 251-289.

[6] P. Buchhold, A class of hierarchical queueing networks and their analysis, *Queueing Systems* **15** (1994) 59-80.

[7] J.P. Buzen, R.P. Goldberg, A.M. Langer, E. Lentz, H.S. Schwenk, D.A. Sheetz, and A. Shum, BEST/1-Design of a tool for computer system capacity planning, *Proc. AFIP NCC* (1978) 447-455.

[8] K.M. Chandy, U. Herzog, and L. Woo, Approximate analysis of general queueing networks, *IBM J. Res. Dev.* **19** (1975) 43-49.

[9] K.M. Chandy, U. Herzog, and L.Woo, Parametric analysis of queueing networks,*IBM J. Res. Dev.* **19** (1975) 36-42.

[10] K.M. Chandy, J.H. Howard, and D.F Towsley, Product form and local balance in queueing networks,*J.ACM* **27** (1977) 250-263.

[11] K.M. Chandy and A.J. Martin, A characterization of product form queueing networks, *J.ACM* **30** (1983) 286-299.

[12] J.W. Cohen, *The Single Server Queue* (North-Holland, 1982).

[13] P.J. Courtois, *Decomposability: Queueing and computer system applications* (Academic Press, 1977).

[14] R.L. Disney and D. König, Queueing networks: A survey of their random processes, *SIAM Review* **27** (1985) 335-403.

[15] W. Henderson and P.G. Taylor, Product form in networks of queues with batch arrivals and batch services, *Queueing Systems* **6** (1990) 71-88.

[16] J.R. Jackson, Job-shop like queueing systems, *Manag. Sci.* **10** (1963) 131-142.

[17] P.A. Jacobson and E.D. Lazowska, Analyzing queueing networks with simultaneous resource possession, *C.ACM* **25** (1982).

[18] F.P. Kelly, *Reversibility and Stochastic Networks* (Wiley, New York, 1979).

[19] F.P. Kelly, Networks of Queues, *Adv. Appl. Prob.* **8** (1976) 416-432.

[20] I. Kino and S. Morita, PERFORMS - A support system for computer system performance evaluation, *Proc. Modelling Techniques and Tools for Performance Analysis*, Paris (North-Holland, 1984) 119-138.

[21] A.G. Konheim and M. Reiser, Finite capacity queueing systems with applications in computer modeling, *SIAM J. Computing* **7** (1978) 210-229.

[22] P.J. Kuehn, Approximate analysis of general queueing networks by decomposition, *IEEE Trans. Comm.* **27**(6) (1979) 113-126.

[23] S.S. Lam, Queueing networks with population size constraints, *IBM J. Res. Dev.* **41**(4) (1977) 370-378.

[24] S.S. Lavenberg (Ed.), *Computer Performance Modeling Handbook*, (Academic Press, 1983).

[25] M. Miyazawa, Insensitivity and product form decomposability of reallocatable GSMP, *Adv. Apl. Prob.* **25** (1993) 415-437.

[26] C.H. Sauer, Approximate solution of queueing networks with simultaneous resource possession, *IBM J. Res. Dev.* **25**(6) (1981) 894-903.

[27] C.H. Sauer, M. Reiser, and MacNair: RESQ - A package for solution of generalized queueing networks, *Proc. NCC* (1977) 977-986.

[28] R.F. Serfozo, Markovian network processes: Congestion-dependent routing and processing, *Queueing Systems* **5** (1989) 5-36.

[29] R.F. Serfozo, *Reversibility and compound birth-death and Migration Processes, Queueing and Related Models* (Oxford Univ. Press, 1992) 65-90.

[30] R.F. Serfozo, Queueing networks with dependent nodes and concurrent movements, *Queueing Systems* **13** (1993) 143-182.

[31] N.M. Van Dijk, Stop=recirculate for exponential product form queueing networks with departure blocking, *Oper. Res. Letters* **10** (1991) 345-351.

[32] N.M. Van Dijk, *Queueing networks and product form* (John Wiley, Chicheter, 1993).

[33] M. Veran and D. Potier, QNAP2: A portable environment for queueing network modelling, *Proc. Modelling Techniques and Tools for Performance Analysis*, Paris (1984).

[34] J. Walrand, *An Introduction to Queueing Networks* (Prentice Hall, 1988).

[35] J. Walrand, A note on Norton's theorem for queueing networks,*J. Appl. Prob.* **20** (1983) 442-444.

[36] P. Whittle, *Systems in Stochastic Equilibrium* (John Wiley, 1986).

Issei Kino
C&C Research Laboratories NEC
Miyazaki Miyamae-ku Kawasaki 216 Japan
kino@sbl.cl.nec.co.jp