Approximate Analysis of Tandem Blocking Queueing Networks with Correlated Arrivals and Services

Guidance

Professor Masao FUKUSHIMA
Professor Tetsuya TAKINE

Kentaro OKAZAKI

2003 Graduate Course
in
Department of Applied Mathematics and Physics
Graduate School of Informatics
Kyoto University

February 2005

Abstract

Many real systems such as production lines are modeled as queueing networks with finite buffers. In such a queueing model, blocking occurs if a customer attempts to enter the next queue whose buffer is full and not available. In this case, the customer is forced to wait until the next queue can be entered. General queueing networks with blocking do not have a product-form solution. Therefore, it is very difficult to obtain the exact solution because of the explosion of state space. Accordingly, many approximation methods have been proposed so far. Most of them are based on the node decomposition method, where the queueing network is decomposed into several subsystems, and each subsystem is analyzed separately. Also, most of approximation methods proposed in the past consider renewal arrivals and services in each node. However, they can be correlated in real systems. It is known that the correlations in arrivals and services have a great impact on the performance of queueing networks.

In this thesis, we develop an approximate method for tandem queueing networks with finite buffers and blocking, taking account of correlations in arrivals and services. We assume that the arrival process is a two-state Markov Arrival Process (MAP), which can represent correlation in interarrival times. In addition, we apply MAP to the service process in each node. In our method, a tandem queueing network with n nodes is decomposed into $n-2$ subsystems, each of which consists of three nodes in tandem. We develop an efficient algorithmic procedure for analyzing each subsystem and propose an iterative procedure for computing the steady-state probabilities of the number of customers in each node. Through numerical experiments, we examine the accuracy of the approximation and confirm that our method well approximates the performance of tandem queueing networks with correlated arrivals and services.

Contents

1 Introduction **1**

2 Mathematical model **2**
2.1 Blocking mechanism 2
2.2 Arrival and service processes 2
2.3 Tandem configuration 4

3 Decomposition **5**

4 Analysis of subsystems **6**
4.1 The generator of the underlying Markov chain 6
4.2 Stationary distribution 7
4.3 The statistics of interarrival and service times in the second node 8
4.3.1 Service times in the second node 8
4.3.2 Interarrival times in the second node 10
4.4 Two moments and lag-1 autocorrelation approximation method 11

5 Algorithm **13**
5.1 The first node has an infinite capacity 13
5.2 The first node has a finite capacity 14

6 Numerical experiments **15**

7 Conclusion **20**

A The generator of the underlying Markov chain of a subsystem **21**

1 Introduction

Many real systems such as production lines are modeled as queueing networks with finite buffers and blocking. A tandem queueing network with blocking is shown in Figure 1. External arrivals join the first node. Customers receive services successively from the first node to the last node. In such a queueing model, blocking occurs if a customer attempts to enter the next queue whose buffer is full and not available. In this case, the customer is forced to wait until the next queue can be entered.

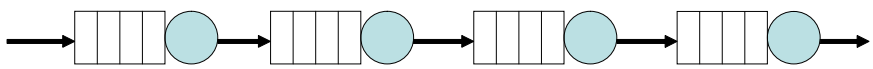

Figure 1: Tandem queueing network.

To design production lines, it is important to reduce the lead-time and in-process inventory. Small buffer reduces the in-process inventory, while it increases the lead-time. Hence, to design production lines, it is necessary to estimate these measures in advance. However, it is very difficult to obtain the exact solution of the queueing network because of the explosion of state space. Accordingly, many approximation methods have been proposed [3, 5, 8]. Most of them are based on the node decomposition method, where the queueing network is decomposed into several subsystems, and each subsystem is analyzed separately. Upstream subsystems influence downstream subsystems, while downstream subsystems also influence upstream subsystems by blocking. Therefore, iterative algorithms are employed.

Most of approximation methods proposed in the past consider renewal arrivals and services in each node, and approximate them by the method of moment matching. It is known that the correlations in arrivals and services have a great impact on the performance of queueing networks. Recently some works propose approximation methods considering the correlation in the arrival process for queueing networks with infinite buffers or customer loss [6, 7, 11]. They report that the correlation in the arrival process makes considerable difference in the average number of customers. It is conceivable that the same is true in the case there is blocking. In [1], Altiok and Melamed analyze the influence of the correlations on performance of the queueing networks. In addition, they give examples of real manufacturing systems with correlated arrivals or services. They alert users to potential deleterious implications stemming from unfounded independence assumptions.

In this thesis, we develop an approximate method for tandem queueing networks with finite buffers and blocking taking account of the correlations. The method estimates the distributions of the queue length of the nodes in tandem queueing networks. The method is based on the node decomposition. In our method, a tandem queueing network with N nodes is decomposed into $N-2$ subsystems, each of which consists of three nodes in tandem. We focus our attention on the autocorrelation of interarrival times in each nodes, as well as the mean and variance. This makes it possible to take account of the effect caused by the correlations in arrivals and services. In our method, we assume that the arrival process is a two-state Markov Arrival Process (MAP), which can represent

correlation in interarrival times. In addition, we apply MAP to the service process in each node.

The rest of this thesis is organized as follows. In section 2, we explain the mathematical model. In section 3, we describe the decomposition of the tandem queueing network. In section 4, we give the method for analyzing each subsystem. We describe the iterative algorithms in section 5. The result of numerical experiments is given in section 6. Finally, the conclusion is given in section 7.

2 Mathematical model

In this section, we introduce blocking mechanisms. After that, we present the arrival and service processes used in this thesis. Finally, we describe the tandem queueing network with blocking.

2.1 Blocking mechanism

In a queueing network with blocking, a customer who attempts to go into the next queue whose buffer is full has to wait. This phenomenon is referred to as blocking. Various blocking mechanisms are propounded to model real systems [4]. We introduce three of the most commonly used blocking mechanisms.

Blocking after service (*BAS*): when a customer attempts to go into a downstream node, if the downstream node is full, the customer has to wait in the former node. While the customer is waiting, the upstream node is blocked, so that other customers can not get service. As soon as the number of customers in the downstream node decreases, the waiting customer moves, and the upstream node is unblocked.

Blocking before service (*BBS*): before a customer receives service in a node, the next node is checked. If the next node is full, the customer can not start receiving service and the former node is blocked. As soon as a departure occurs in the next node, the former node is unblocked. BBS is divided into two classes according to the behavior of waiting customers: waiting customers can be in the server (*BBS-SO* (server occupied)) and not (*BBS-SNO* (server not occupied)).

Repetitive service (*RS*): when a customer finishes the service, if the downstream node is full, the customer can not proceed and it receives service again.

In this thesis, we assume BAS. BAS is used to model production systems etc. The tandem queueing network with BBS-SNO can be converted to the tandem queuing network with BAS. For example, we consider the tandem queueing network with BBS-SNO including three nodes whose capacities are m_1, m_2, and m_3. The system is equivalent to the tandem queueing network with BAS whose capacities are m_1, $m_2 - 1$, and $m_3 - 1$.

2.2 Arrival and service processes

In this thesis, we assume that the arrival process is Markovian Arrival Process (MAP). MAP is an arrival process modulated by an underlying Markov chain with a finite state space $\Omega = \{1, \ldots, M\}$ [10]. We assume the underlying Markov chain is irreducible.

MAP is characterized by two $M \times M$ matrices $\boldsymbol{C}$, and $\boldsymbol{D}$. Diagonal elements of $\boldsymbol{C}$ are negative, and other elements are nonnegative. Elements of $\boldsymbol{D}$ are nonnegative. The sojourn time in state i follows an exponential distribution with mean $(-\boldsymbol{C}_{ii})^{-1}$. The underlying Markov chain changes its state from i to j with rate $\boldsymbol{C}_{i,j}(i \neq j)$ and no customer arrives. With rate $\boldsymbol{D}_{i,j}$, the Markov chain changes its state from i to j and a customer arrives. Therefore, the elements of $\boldsymbol{C}$ and $\boldsymbol{D}$ satisfy the following equations.

$$\sum_{\substack{j=1 \\ j \neq i}}^{M} \boldsymbol{C}_{i,j} + \sum_{j=1}^{M} \boldsymbol{D}_{i,j} = -\boldsymbol{C}_{i,i}, \qquad \text{for all } i \in \Omega.$$

Let MAP$(\boldsymbol{C}, \boldsymbol{D})$ denote MAP with representation $(\boldsymbol{C}, \boldsymbol{D})$.

We explain about the statistical property of MAP$(\boldsymbol{C}, \boldsymbol{D})$ below. $(\boldsymbol{C}+\boldsymbol{D})$ is the generator of the underlying Markov chain. By the assumption that the underlying Markov chain is irreducible, the steady-state probability vector $\boldsymbol{\pi}$ is the unique nonnegative solution of

$$\boldsymbol{\pi}\left(\boldsymbol{C}+\boldsymbol{D}\right) = \boldsymbol{0}, \quad \boldsymbol{\pi}\boldsymbol{e} = 1,$$

where $\boldsymbol{e}$ denotes a column vector with an appropriate dimension, whose elements are all equal to one. Let λ denote the arrival rate of MAP$(\boldsymbol{C}, \boldsymbol{D})$:

$$\lambda = \boldsymbol{\pi}\boldsymbol{D}\boldsymbol{e}.$$

We define $\boldsymbol{\pi}_e$ as the steady-state probability vector of the underlying Markov chain immediately after arrivals. Thus we have

$$\boldsymbol{\pi}_e = \frac{\boldsymbol{\pi}\boldsymbol{D}}{\boldsymbol{\pi}\boldsymbol{D}\boldsymbol{e}}.$$

Let X be the interarrival time in steady-state of MAP$(\boldsymbol{C}, \boldsymbol{D})$. The probability density function of X is shown as

$$f(x) = \boldsymbol{\pi}_e \exp(\boldsymbol{C}x)\boldsymbol{D}\boldsymbol{e}.$$

The Laplace-Stieltjes transform is represented as

$$f(s) = \mathrm{E}[e^{-sX}] = \boldsymbol{\pi}_e \int_0^\infty e^{-sx}\exp(\boldsymbol{C}x)\mathrm{d}x\boldsymbol{D}\boldsymbol{e} = \boldsymbol{\pi}_e\left(s\boldsymbol{I}-\boldsymbol{C}\right)^{-1}\boldsymbol{D}\boldsymbol{e}, \quad \mathrm{Re}(s) > 0.$$

Let $(X_1, X_2, \ldots)$ denote a sequence of interarrival times in steady state of MAP$(\boldsymbol{C}, \boldsymbol{D})$. The first and second moments of X_l are expressed as follows.

$$\begin{aligned} \mathrm{E}[X_l] &= \frac{\boldsymbol{\pi}\boldsymbol{D}}{\boldsymbol{\pi}\boldsymbol{D}\boldsymbol{e}}\left(-\boldsymbol{C}\right)^{-1}\boldsymbol{e} = \frac{1}{\boldsymbol{\pi}\boldsymbol{D}\boldsymbol{e}}, \\ \mathrm{E}[X_l^2] &= 2\frac{\boldsymbol{\pi}\boldsymbol{D}}{\boldsymbol{\pi}\boldsymbol{D}\boldsymbol{e}}\left(\left(-\boldsymbol{C}\right)^{-1}\right)^2\boldsymbol{e} = \frac{2}{\boldsymbol{\pi}\boldsymbol{D}\boldsymbol{e}}\boldsymbol{\pi}\left(-\boldsymbol{C}\right)^{-1}\boldsymbol{e}. \end{aligned}$$

The state of the underlying Markov chain immediately before an arrival correlates with that of immediately after the arrival. Therefore, there are correlations among interarrival times. The joint density function of $(X_1, X_2, \ldots, X_n)$ is given by

$$g(x_1, x_2, \ldots, x_n) = \boldsymbol{\pi}_e\exp(\boldsymbol{C}x_1)\boldsymbol{D}\exp(\boldsymbol{C}x_2)\boldsymbol{D}\ldots\exp(\boldsymbol{C}x_n)\boldsymbol{D}\boldsymbol{e}.$$

Hence, the lag-k autocorrelation of X_l is given by

$$\mathrm{E}[X_l X_{l+k}] = \frac{\boldsymbol{\pi D}}{\boldsymbol{\pi D e}} (-\boldsymbol{C})^{-1} \left((-\boldsymbol{C})^{-1} \boldsymbol{D}\right)^k (-\boldsymbol{C})^{-1} \boldsymbol{e}. \quad (1)$$

It is known that MAP can approximate any stationary point process with arbitrary accuracy [2]. The more states of the MAP, the more accurately the approximation is. However, the computational time in analysis grows with the number of states. Therefore, we assume that MAP has two states in this thesis.

In what follows, we describe the service process employed in this thesis. We apply MAP to the service process. Let $\Omega_s = \{1, \ldots, M_s\}$ be the state space of the service process. The service process is expressed by two $M_s \times M_s$ matrices $\boldsymbol{C}_s$ and $\boldsymbol{D}_s$. Among elements of $\boldsymbol{C}_s$ and $\boldsymbol{D}_s$, the same relations are applied as in $\boldsymbol{C}$ and $\boldsymbol{D}$ of MAP$(\boldsymbol{C}, \boldsymbol{D})$. One customer receives the service at once. While there is no customer in the server, the state of the service process never changes. If we observe the process only for periods during which there are some customers in the server, the service process is identical with MAP$(\boldsymbol{C}_s, \boldsymbol{D}_s)$. We assume that the underlying Markov chain is irreducible. The service time of the lth customer is denoted by Y_l. The sequence of service times $(Y_1, Y_2, \ldots, Y_n)$ has the joint probability density function

$$g(y_1, y_2, \ldots, y_n) = \boldsymbol{\pi}_s \exp(\boldsymbol{C}_s y_1) \boldsymbol{D}_s \exp(\boldsymbol{C}_s y_2) \boldsymbol{D}_s \ldots \exp(\boldsymbol{C}_s x_n) \boldsymbol{D}_s \boldsymbol{e},$$

where $\boldsymbol{\pi}_s$ denotes the steady-state probability vector of the underlying Markov chain immediately after services. This is similar to the joint probability density function of $(X_1, X_2, \ldots, X_n)$. Hence, the statistics of service times are expressed in the same way as those of interarrival times of MAP$(\boldsymbol{C}_s, \boldsymbol{D}_s)$.

We name this process Markovian Service Process (MSP). We denote by MSP$(\boldsymbol{C}_s, \boldsymbol{D}_s)$ MSP with representation $(\boldsymbol{C}_s, \boldsymbol{D}_s)$.

2.3 Tandem configuration

We consider open queueing networks consisting of N queues in series. We number the nodes from 1 to N. The first node has a finite or infinite capacity buffer, and other nodes have finite capacity buffers. Let m_k denote the capacity of node-k. We assume the blocking mechanism is BAS. The arrival process to the first node is MAP$(\boldsymbol{C}_a, \boldsymbol{D}_a)$ and the service process in node-k is MSP$(\boldsymbol{C}_k, \boldsymbol{D}_k)$. The arrival rate of MAP$(\boldsymbol{C}_a, \boldsymbol{D}_a)$ is denoted by λ_a. We define $n_k(t)$, and $p_k(t)$ as the number of customers and the state of the service process, respectively, in node-k at time t. In addition, $n_k(t) = m_k + 1$ means that there are m_k customers in node-k and node-k is blocking the upstream node at time t. Then the process $\{(n_1(t), \ldots, n_N(t), p_1(t), \ldots, p_N(t)), t \geq 0\}$ is the continuous time Markov chain.

General tandem configurations do not have a closed-form solution. One way of analyzing such queueing networks is to solve the Markov chain and gain the stationary probability vector. However, it is not practical since the number of states in the Markov chain explosively increases with the number of nodes.

3 Decomposition

In this section, we present our approximate approach to tandem queueing networks with blocking. The algorithm approximates the stationary distribution of the queue length in each node.

We assume that the first node has an infinite capacity. Therefore, if the system is stable, all arrival customers can go into the system, and the throughput of the queueing network is equal to λ_a.

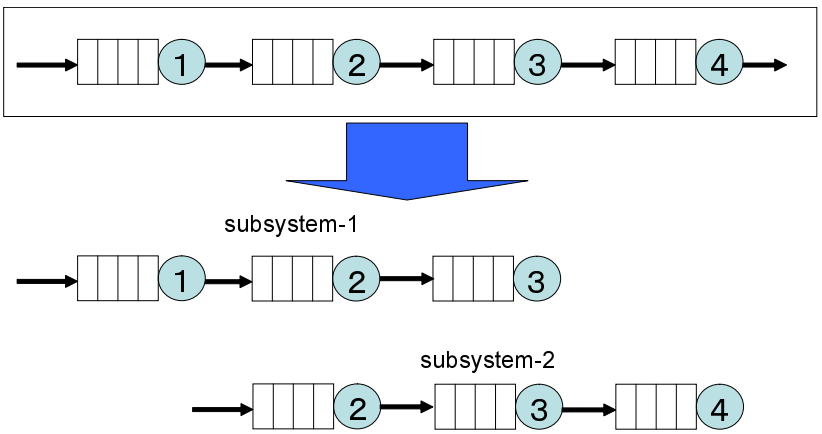

Figure 2: Decomposition to subsystems.

First, the tandem queueing network is decomposed into $N-2$ subsystems as in Figure 2. Each subsystem consists of three nodes. Subsystem-1 includes node-1, node-2, and node-3. Subsystem-2 includes node-2, node-3, and node-4, etc. Let $m_{k,i}(k = 1, 2, 3)$ be the maximum capacities of the ith node in subsystem-k, and the capacities of these nodes are identical with those of the corresponding nodes in the original queueing network, that is, $m_{k,i} = m_{k+i-1}$.

We define a two-state MAP($\boldsymbol{C}_{k,a}, \boldsymbol{D}_{k,a}$) as the arrival process to subsystem-k and MSP($\boldsymbol{C}_{k,d}, \boldsymbol{D}_{k,d}$) as the service process in the third node of subsystem-k. The service processes in the first and the second nodes are identical with those of the corresponding nodes in the original tandem queueing network. We define $a_k(t)$ as the state of the arrival process at time t. Let $p_{k,i}(t)$ be the state of service process in the ith node of subsystem-k at time t. Also let $n_{k,i}(t)$ denote the number of customers in the ith node in subsystem-k at time t. In addition, let $n_{k,i}(t) = m_{k,i} + 1(i = 2, 3)$ represent that the ith node in subsystem-k is full and the $i-1$st node is blocked.

Our intent is that the behavior of $n_{k,2}(t)$ closely matches that of $n_{k+1}(t)$. For that purpose, MAP($\boldsymbol{C}_{k,a}, \boldsymbol{D}_{k,a}$) emulates the arrival process to node-k in the original network, and MSP($\boldsymbol{C}_{k,d}, \boldsymbol{D}_{k,d}$) emulates the service process of node-$k+2$ including the effect of blocking. These are unknown processes at first, except the arrival process to the first node and service process in the last node. Therefore, we take an iterative algorithm.

The first node of each subsystem has a finite buffer with the exception of subsystem-1. Therefore, there are customer losses in subsystems. However, in the actual queueing

network, all customers go through the network without being lost. When the downstream node is full, if upstream server completes the service, blocking occurs. As soon as the downstream server completes the service, the blocked customer enters the downstream node. To simulate this behavior we assume that when the first node is full, if the service in the first node is finished, new arrival occurs with the probability q. The throughput of subsystem-k is a function of q. Therefore, let $\lambda_k(q)$ denote the throughput of subsystem-k. We choose a suitable value of q so that $\lambda_k(q) = \lambda_a$. $\lambda_k(q)$ is monotonically increasing with q. If $q = 0$, $\lambda_k(q) \leq \lambda_a$ because of customer losses. $q = 1$ means that there are always customers in the first node of subsystem. Hence, $\lambda_k(q) \geq \lambda_a$ unless the system is unstable. Therefore, if the original queueing network is stable, $\lambda_k(q) = \lambda_a, 0 \leq q \leq 1$ has a solution.

Let $S(t) = \{n_{k,1}(t), n_{k,3}(t), a_k(t), p_{k,1}(t), p_{k,2}(t), p_{k,3}(t)\}$. The state of subsystem-k at time t is expressed as $(n_{k,2}(t), S(t))$. The process $\{(n_{k,2}(t), S(t)), t \geq 0\}$ is a continuous time Markov chain. Let $\boldsymbol{Q}_k$ denote the infinitesimal generator of the Markov chain. Then we can solve the balance equations,

$$\boldsymbol{\pi}_k \boldsymbol{Q}_k = \boldsymbol{0}, \quad \boldsymbol{\pi}_k \boldsymbol{e} = 0,$$

where $\boldsymbol{\pi}_k$ denotes the steady-state probability vector of the Markov chain. In the next section, we describe the analytical method in details. The steady-state queue length distribution of the second node in subsystem-k approximately represents that of node-$k+1$ in the original queueing network. In addition, the arrival process and the service process in the second node correspond to the arrival process and the service process, respectively, in node-$k+1$ in the original network.

Once the stationary state distribution of the Markov chain is given, we can evaluate the first two moments and the lag-1 autocorrelation of interarrival times and service times in the second node of subsystem-k. Thus we can construct MAP($\boldsymbol{C}_{k+1,a}, \boldsymbol{D}_{k+1,a}$) and MAP($\boldsymbol{C}_{k-1,d}, \boldsymbol{D}_{k-1,d}$) to emulate these statistics. In the next section, we describe how to evaluate these statistics and to fit statistics of a MAP to those.

4 Analysis of subsystems

In this section, we obtain the stationary state probability vector of subsystem-k. Next, we obtain the statistics of the arrival and service processes in the second node of subsystem-k. Finally, we describe the method for constructing the arrival and service processes of the adjacent subsystems.

4.1 The generator of the underlying Markov chain

In subsystem-k, the underlying Markov chain can move from (i, S) to $(i+1, S')$ for $0 \leq i \leq m_{k,2}$ and from (i, S) to $(i-1, S')$ for $1 \leq i \leq m_{k,2}+1$. Hence, the infinitesimal

generator of subsystem-k has the $(m_2+2)\times(m_2+2)$ block tridiagonal structure as follows.

$$\boldsymbol{Q}_k = \begin{bmatrix} \boldsymbol{B}_1 & \boldsymbol{B}_2 & & & & \\ \boldsymbol{B}_0 & \boldsymbol{A}_1 & \boldsymbol{A}_2 & & \boldsymbol{O} & \\ & \boldsymbol{A}_0 & \boldsymbol{A}_1 & \ddots & & \\ & & \ddots & \ddots & \boldsymbol{A}_2 & \\ & \boldsymbol{O} & & \boldsymbol{A}_0 & \boldsymbol{A}_1 & \boldsymbol{H}_2 \\ & & & & \boldsymbol{H}_0 & \boldsymbol{P}_2 \end{bmatrix}.$$

Each element of the (i,j)th block corresponds to the rate with which the state of the Markov chain moves from (i,S) to (j,S'), except the diagonal elements of $\boldsymbol{Q}_k$. The structure of each block is given in Appendix.

4.2 Stationary distribution

We analyze the underlying Markov chain to obtain the steady-state probability vector $\boldsymbol{\pi_k}$. For that purpose, we use the uniformization technique. We define θ_k and $\boldsymbol{Q}_{ku}$ as

$$\theta_k = \max_i |(\boldsymbol{Q}_k)_{ii}|,$$

and

$$\boldsymbol{Q}_{ku} = \boldsymbol{I} + \frac{1}{\theta}\boldsymbol{Q}_k,$$

respectively. Note that $\boldsymbol{Q}_{ku}$ is a nonnegative matrix. In addition,

$$\boldsymbol{Q}_{ku}\boldsymbol{e} = \left(\boldsymbol{I} + \frac{1}{\theta}\boldsymbol{Q}_k\right)\boldsymbol{e} = 1.$$

Therefore, $\boldsymbol{Q}_{ku}$ is considered as the transition matrix of a discrete time Markov chain $Y(t)$. Since $\boldsymbol{\pi}_k\boldsymbol{Q}_k = 0$,

$$\boldsymbol{\pi}_k\boldsymbol{Q}_{ku} = \boldsymbol{\pi}_k\left(\boldsymbol{I} + \frac{1}{\theta}\boldsymbol{Q}_k\right) = \boldsymbol{\pi}_k.$$

Therefore, $\boldsymbol{\pi}_k$ is identical with the steady-state probability vector of the discrete time Markov chain $Y(t)$. We then solve the following balance equations to obtain $\boldsymbol{\pi}_k$.

$$\boldsymbol{\pi}_k\boldsymbol{Q}_{ku} = \boldsymbol{\pi}_k, \quad \boldsymbol{\pi}_k\boldsymbol{e} = 1. \tag{2}$$

$\boldsymbol{Q}_{ku}$ has the same structure as $\boldsymbol{Q}_k$, that is, if $(\boldsymbol{Q}_k)_{i,j} = 0$, then $(\boldsymbol{Q}_{ku})_{i,j} = 0$. $\boldsymbol{Q}_k$ is a sparse matrix. There are at most nine nonzero elements in each low. Therefore, we can utilize the sparse structure to compute $\boldsymbol{\pi}_k$.

We use the Gauss-Seidel Iteration to solve (2). The method converges for ordinary Markov chains. In case that the method does not converge, we employ the power method. The power method converges for any Markov cahins that have steady-state probability vector. However, the convergence of the power method is slower than that of the Gauss-Seidel Iteration. In the algorithm, we assume that $m_{k,2} \geq 2$.

Step 1. Initialization:

- Define $\boldsymbol{\pi}_k^{(0)}$ arbitrary, set $n = 0$.

Step 2. $n := n + 1$.

- Solve

$$\begin{aligned}
\boldsymbol{\pi}_{k,0}^{(n)}\boldsymbol{B}_{1u} + \boldsymbol{\pi}_{k,1}^{(n-1)}\boldsymbol{B}_{0u} &= \boldsymbol{\pi}_{k,0}^{(n)}, \\
\boldsymbol{\pi}_{k,0}^{(n)}\boldsymbol{B}_{2u} + \boldsymbol{\pi}_{k,1}^{(n)}\boldsymbol{A}_{1u} + \boldsymbol{\pi}_{k,2}^{(n-1)}\boldsymbol{A}_{0u} &= \boldsymbol{\pi}_{k,1}^{(n)}, \\
\boldsymbol{\pi}_{k,l-1}^{(n)}\boldsymbol{A}_{2u} + \boldsymbol{\pi}_{k,l}^{(n)}\boldsymbol{A}_{1u} + \boldsymbol{\pi}_{k,l+1}^{(n-1)}\boldsymbol{A}_{0u} &= \boldsymbol{\pi}_{k,l}^{(n)}, \quad l = 2, \ldots, m_{k,2} - 1, \\
\boldsymbol{\pi}_{k,m_{k,2}-1}^{(n)}\boldsymbol{A}_{2u} + \boldsymbol{\pi}_{k,m_{k,2}}^{(n)}\boldsymbol{A}_{1u} + \boldsymbol{\pi}_{k,m_{k,2}+1}^{(n-1)}\boldsymbol{H}_{0u} &= \boldsymbol{\pi}_{k,m_{k,2}}^{(n)}, \\
\boldsymbol{\pi}_{k,m_{k,2}}^{(n)}\boldsymbol{H}_{2u} + \boldsymbol{\pi}_{k,m_{k,2}+1}^{(n)}\boldsymbol{P}_{2u} &= \boldsymbol{\pi}_{k,m_{k,2}+1}^{(n)}.
\end{aligned}$$

- Normalize

$$\boldsymbol{\pi}_k^{(n)} = \frac{\boldsymbol{\pi}_k^{(n)}}{\boldsymbol{\pi}_k^{(n)}\boldsymbol{e}}.$$

If $\|\boldsymbol{\pi}_k^{(n)} - \boldsymbol{\pi}_k^{(n)}\boldsymbol{Q}_{ku}\| \leq \epsilon$, then stop. Otherwise, goto step2.

4.3 The statistics of interarrival and service times in the second node

4.3.1 Service times in the second node

We consider the departure process from the second node under the condition that there are some customers in the second node. Therefore, we construct the censored Markov chain under the condition $n_2(t) \geq 1$ so that we can obtain MSP($\boldsymbol{C}_{2s}, \boldsymbol{D}_{2s}$), which represents the service process in the second node.

The generator of the censored Markov chain $\boldsymbol{Q}_{\text{cens}}$ is given by

$$\boldsymbol{Q}_{\text{cens}} = \begin{bmatrix}
\boldsymbol{A}_1 + \boldsymbol{B}_0(-\boldsymbol{B}_1)^{-1}\boldsymbol{B}_2 & \boldsymbol{A}_2 & & & & \\
\boldsymbol{A}_0 & \boldsymbol{A}_1 & \boldsymbol{A}_2 & & & \\
 & \boldsymbol{A}_0 & \ddots & \ddots & & \\
 & & \ddots & \boldsymbol{A}_1 & \boldsymbol{A}_2 & \\
 & & & \boldsymbol{A}_0 & \boldsymbol{A}_1 & \boldsymbol{H}_2 \\
 & & & & \boldsymbol{H}_0 & \boldsymbol{P}_2
\end{bmatrix}.$$

We extract the elements of $\boldsymbol{Q}_{\text{cens}}$ corresponding to the rates for departure from the second

node, and construct $\boldsymbol{D}_{2s}$. $\boldsymbol{C}_{2s}$ is composed of other elements.

$$\boldsymbol{C}_{2s} = \begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{A}_2 & & & \\ & \boldsymbol{A}_1 & \ddots & & \\ & & \ddots & \boldsymbol{A}_2 & \\ & & & \boldsymbol{A}_1 & \boldsymbol{H}_2 \\ & & & & \boldsymbol{P}_2 \end{bmatrix}, \quad \boldsymbol{D}_{2s} = \begin{bmatrix} \boldsymbol{B}_0(-\boldsymbol{B}_1)^{-1}\boldsymbol{B}_2 & & & & & \\ & \boldsymbol{A}_0 & \boldsymbol{O} & & & \\ & & \boldsymbol{A}_0 & \ddots & & \\ & & & \ddots & \boldsymbol{O} & \\ & & & & \boldsymbol{A}_0 & \boldsymbol{O} \\ & & & & & \boldsymbol{H}_0 \quad \boldsymbol{O} \end{bmatrix}.$$

Let T_+ denote the subset of the all states corresponding to the positive numbers of customers in the second node, and $\boldsymbol{\pi}_{k+}$ denote the sub-vector of $\boldsymbol{\pi}_k$ over T_+. The stationary probability vector $\boldsymbol{\pi}_{\text{cens}}$ of the censored process is proportional to $\boldsymbol{\pi}_{k+}$ [9], i.e.,

$$\boldsymbol{\pi}_{\text{cens}} = (\boldsymbol{\pi}_{k+}\boldsymbol{e})^{-1}\,\boldsymbol{\pi}_{k+}.$$

Let $(V_1, V_2, \ldots)$ be a sequence of service times of MSP$(\boldsymbol{C}_{2s}, \boldsymbol{D}_{2s})$. The statistics of $(V_1, V_2, \ldots)$ are

$$\begin{aligned} \mathrm{E}[V_l] &= \frac{1}{\boldsymbol{\pi}_{\text{cens}}\boldsymbol{D}_{2s}\boldsymbol{e}}, \\ \mathrm{E}[V_l^2] &= \frac{2}{\boldsymbol{\pi}_{\text{cens}}\boldsymbol{D}_{2s}\boldsymbol{e}}\boldsymbol{\pi}_{\text{cens}}\,(-\boldsymbol{C}_{2s})^{-1}\,\boldsymbol{e}, \\ \mathrm{E}[V_l V_{l+1}] &= \frac{1}{\mathrm{E}[V_l]}\boldsymbol{\pi}_{\text{cens}}(-\boldsymbol{C}_{2s})^{-1}\boldsymbol{D}_{2s}(-\boldsymbol{C}_{2s})^{-1}\boldsymbol{e}. \end{aligned}$$

The inversion of $\boldsymbol{C}_{2s}$ is the most expensive operation. Since $\boldsymbol{C}_{2s}$ is a block upper triangular matrix, we can calculate it efficiently as follows.

We define $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{m_{k,2}+1})$ so that

$$(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{m_{k,2}+1}) = \boldsymbol{\pi}_{\text{cens}}(-\boldsymbol{C}_{2s})^{-1}.$$

Using the block matrices, $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{m_{k,2}+1})$ can be written as

$$\boldsymbol{x}_1 = \boldsymbol{\pi}_0(-\boldsymbol{A}_1)^{-1}, \tag{3}$$

$$\boldsymbol{x}_l = (\boldsymbol{x}_{l-1}\boldsymbol{A}_2 + \boldsymbol{\pi}_l)(-\boldsymbol{A}_1)^{-1}, \quad l = 2, \ldots, m_{k,2}, \tag{4}$$

$$\boldsymbol{x}_{m_{k,2}+1} = (\boldsymbol{x}_{m_{k,2}}\boldsymbol{H}_2 + \boldsymbol{\pi}_{m_{k,2}+1})(-\boldsymbol{P}_2)^{-1}. \tag{5}$$

We described the system of equations from (3) to (5) using the block elements of the generator. However, we can write the equations by the uniformized generator.

The block elements of the uniformized generator are derived as follows. For a diagonal block $\boldsymbol{A}$, uniformized block $\boldsymbol{A}_u$ is given by

$$\boldsymbol{A}_u = \boldsymbol{I} + \frac{1}{\theta}\boldsymbol{A}.$$

Therefore,

$$(-\boldsymbol{A})^{-1} = \frac{1}{\theta}\,(I - \boldsymbol{A}_u)^{-1}\,.$$

For a non-diagonal block $\boldsymbol{B}$, we define $\boldsymbol{B}_u$ as

$$\boldsymbol{B}_u = \frac{1}{\theta}\boldsymbol{B}.$$

We define $\boldsymbol{u} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{m_{k,2}+1})$ as $\theta^{-1}\boldsymbol{x}$. Using the uniformized generator, we can rewrite the system of equations from (3) to (5) as below,

$$\begin{aligned}
\boldsymbol{u}_1 &= \boldsymbol{\pi}_{cens,0}(\boldsymbol{I} - \boldsymbol{A}_{1u})^{-1}, \\
\boldsymbol{u}_l &= (\boldsymbol{u}_{l-1}\boldsymbol{A}_{2u} + \boldsymbol{\pi}_{cens,l})(\boldsymbol{I} - \boldsymbol{A}_{1u})^{-1}, \quad l = 2, \ldots, m_{k,2}, \\
\boldsymbol{u}_{m_{k,2}+1} &= (\boldsymbol{x}_{m_{k,2}}\boldsymbol{H}_{2u} + \boldsymbol{\pi}_{cens,m_{k,2}+1})(\boldsymbol{I} - \boldsymbol{P}_{2u})^{-1}.
\end{aligned}$$

The inverse matrix in each equation has already been obtained in analysis of the stationary state distribution.

We define $\boldsymbol{y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{m2+1})$ as

$$\boldsymbol{y} = \boldsymbol{\pi}_{\text{cens}}(-\boldsymbol{C}_{2s})^{-1}\boldsymbol{D}_{2s} = \boldsymbol{x}\boldsymbol{D}_{2s}.$$

Consequently,

$$\begin{aligned}
\boldsymbol{y}_1 &= \boldsymbol{u}_1\boldsymbol{B}_{0u}(\boldsymbol{I} - \boldsymbol{B}_{1u})^{-1}\boldsymbol{B}_{2u} + \boldsymbol{u}_2\boldsymbol{A}_{0u}, \\
\boldsymbol{y}_l &= \boldsymbol{u}_{l+1}\boldsymbol{A}_{0u}, \qquad l = 2, \ldots, m_{k,2} - 1, \\
\boldsymbol{y}_{m_{k,2}} &= \boldsymbol{u}_{m_{k,2}+1}\boldsymbol{H}_0, \\
\boldsymbol{y}_{m_{k,2}+1} &= \boldsymbol{O}.
\end{aligned}$$

Now we can compute $\boldsymbol{\pi}_{\text{cens}}(-\boldsymbol{C}_{2s})^{-1}\boldsymbol{D}_{2s}(-\boldsymbol{C}_{2s})^{-1}$ in a similar way to compute $\boldsymbol{\pi}_{\text{cens}}(-\boldsymbol{C}_{2s})^{-1}$ replacing $\boldsymbol{\pi}_{\text{cens}}$ by $\boldsymbol{y}$.

4.3.2 Interarrival times in the second node

The statistics of interarrival times in the second node can be obtained as the preceding subsection. The arrival process is MAP $(\boldsymbol{C}_{2a}, \boldsymbol{D}_{2a})$ described as follows.

$$\boldsymbol{C}_{2a} = \begin{bmatrix}
\boldsymbol{B}_1 & \boldsymbol{O} & & & & \\
\boldsymbol{B}_0 & \boldsymbol{A}_1 & \ddots & & & \\
 & \boldsymbol{A}_0 & \ddots & \ddots & & \\
 & & \ddots & \ddots & \boldsymbol{O} & \\
 & & & \boldsymbol{A}_0 & \boldsymbol{A}_1 & \boldsymbol{O} \\
 & & & & \boldsymbol{H}_0 & \boldsymbol{P}_2
\end{bmatrix}, \quad
\boldsymbol{D}_{2a} = \begin{bmatrix}
\boldsymbol{O} & \boldsymbol{B}_2 & & & & \\
\boldsymbol{O} & \boldsymbol{O} & \boldsymbol{A}_2 & & & \\
 & \ddots & \ddots & \ddots & & \\
 & & \ddots & \boldsymbol{O} & \boldsymbol{A}_2 & \\
 & & & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{H}_2 \\
 & & & & \boldsymbol{O} & \boldsymbol{O}
\end{bmatrix}.$$

The elements of $\boldsymbol{D}_{2a}$ correspond to the rates for service completions in the first node. Note that the elements of $\boldsymbol{H}_2$ do not correspond to the arrival to the second node in fact.

We denote by $(W_1, W_2, \ldots)$ a sequence of interarrival times of MAP $(\boldsymbol{C}_{2a}, \boldsymbol{D}_{2a})$. We then have

$$\begin{aligned}
\mathrm{E}[W_l] &= \frac{1}{\boldsymbol{\pi}_k \boldsymbol{D}_{2a} \boldsymbol{e}}, \\
\mathrm{E}[W_l^2] &= \frac{2}{\boldsymbol{\pi}_k \boldsymbol{D}_{2a} \boldsymbol{e}} \boldsymbol{\pi}_k \left(-\boldsymbol{C}_{2a}\right)^{-1} \boldsymbol{e}, \\
\mathrm{E}[W_l W_{l+1}] &= \frac{1}{\mathrm{E}[W_l]} \boldsymbol{\pi}_k (-\boldsymbol{C}_{2a})^{-1} \boldsymbol{D}_{2a} (-\boldsymbol{C}_{2a})^{-1} \boldsymbol{e}.
\end{aligned}$$

We define $\boldsymbol{v} = (\boldsymbol{v}_0, \ldots, \boldsymbol{v}_{m_{k,2}+1})$ as $\theta^{-1}(-\boldsymbol{C}_{2a})^{-1}\boldsymbol{e}$. Since $\boldsymbol{C}_{2a}$ is a block lower triangular matrix, we can compute $\boldsymbol{v}$ as follows.

$$\begin{aligned}
\boldsymbol{v}_0 &= (\boldsymbol{I} - \boldsymbol{B}_{1u})^{-1} \boldsymbol{e}, \\
\boldsymbol{v}_1 &= (\boldsymbol{I} - \boldsymbol{A}_{1u})^{-1} (\boldsymbol{B}_{0u} \boldsymbol{v}_0 + \boldsymbol{e}), \\
\boldsymbol{v}_l &= (\boldsymbol{I} - \boldsymbol{A}_{1u})^{-1} (\boldsymbol{A}_{0u} \boldsymbol{v}_{l-1} + \boldsymbol{e}), \quad l = 2, \ldots, m_{k,2}, \\
\boldsymbol{v}_{m_{k,2}+1} &= (\boldsymbol{I} - \boldsymbol{P}_{2u})^{-1} (\boldsymbol{H}_{0u} \boldsymbol{v}_{m_{k,2}} + \boldsymbol{e}).
\end{aligned}$$

We define $\boldsymbol{w}$ as

$$\boldsymbol{w} = \begin{bmatrix} \boldsymbol{w}_0 \\ \vdots \\ \boldsymbol{w}_{m_{k,2}+1} \end{bmatrix} = \boldsymbol{D}_{2a}(-\boldsymbol{C}_{2a})^{-1}\boldsymbol{e}.$$

Therefore, we have

$$\begin{aligned}
\boldsymbol{w}_0 &= \boldsymbol{B}_{2u} \boldsymbol{v}_1, \\
\boldsymbol{w}_l &= \boldsymbol{A}_{2u} \boldsymbol{v}_{l+1}, \qquad l = 1, \ldots, m_{k,2} - 1, \\
\boldsymbol{w}_{m_{k,2}} &= \boldsymbol{H}_{2u} \boldsymbol{v}_{m_{k,2}+1}, \\
\boldsymbol{w}_{m_{k,2}+1} &= \boldsymbol{0}.
\end{aligned}$$

Then, we can compute $(-\boldsymbol{C}_{2a})^{-1}\boldsymbol{D}_{2a}(-\boldsymbol{C}_{2a})^{-1}\boldsymbol{e}$ in a similar way to compute $(-\boldsymbol{C}_{2a})^{-1}\boldsymbol{e}$ replacing $\boldsymbol{e}$ by $\boldsymbol{w}$.

4.4 Two moments and lag-1 autocorrelation approximation method

It may be difficult to analyze a queueing model involving a MAP with a large number of phases. Therefore, such a MAP is approximated by a MAP with two phases.

Given the first two moments e_1, e_2 and the lag-1 autocorrelation l_1 of interarrival times of the MAP with a large number of phases, we construct MAP$(\boldsymbol{C}_f, \boldsymbol{D}_f)$ with two phases by an approximation method given below. We assume that

$$\boldsymbol{C}_f = \begin{bmatrix} -\alpha & \delta\alpha \\ 0 & -\beta \end{bmatrix}, \qquad \boldsymbol{D}_f = \begin{bmatrix} \alpha - \gamma & \gamma - \delta\alpha \\ \gamma & \beta - \gamma \end{bmatrix},$$

$$\alpha, \beta > 0, \quad \min(\alpha, \beta) > \gamma > \delta\alpha, \quad 0 \le \delta \le 1.$$

The steady-state probability vector of the underlying Markov chain is $\boldsymbol{\pi}_f = [0.5, 0.5]$. Let $(Z_1, Z_2, \ldots)$ denote a sequence of interarrival times of this MAP. The first two moments and the lag-1 autocorrelation of $(Z_1, Z_2, \ldots)$ are obtained as follows.

$$\begin{aligned}
\mathrm{E}[Z_l] &= \frac{1}{\boldsymbol{\pi}_f \boldsymbol{D}_f \boldsymbol{e}} = \frac{2}{(1-\delta)\alpha + \beta}, \\
\mathrm{E}[Z_l^2] &= 2\frac{\boldsymbol{\pi}_f \boldsymbol{D}_f}{\boldsymbol{\pi}_f \boldsymbol{D}_f \boldsymbol{e}} \left((-\boldsymbol{C}_f)^{-1}\right)^2 \boldsymbol{e} = \frac{2}{\alpha\beta}\frac{(1+\delta)\alpha + \beta}{(1-\delta)\alpha + \beta} = \frac{(1+\delta)\alpha + \beta}{\alpha\beta}\mathrm{E}[Z_l], \\
\mathrm{E}[Z_l Z_{l+1}] &= \frac{\boldsymbol{\pi}_f \boldsymbol{D}_f}{\boldsymbol{\pi}_f \boldsymbol{D}_f \boldsymbol{e}} (-\boldsymbol{C}_f)^{-1} \left((-\boldsymbol{C}_f)^{-1} \boldsymbol{D}_f\right) (-\boldsymbol{C}_f)^{-1} \boldsymbol{e} \\
&= \frac{(1+\delta)\,\alpha + \beta}{((1-\delta)\,\alpha + \beta)\,\alpha\beta} - \frac{\gamma\left((\alpha-\beta)^2 - \alpha^2\delta^2\right)}{((1-\delta)\,\alpha + \beta)\,\alpha^2\beta^2} \\
&= \frac{1}{2}\mathrm{E}[Z_l^2] - \gamma\mathrm{E}[Z_l]\frac{\left((\alpha-\beta)^2 - \alpha^2\delta^2\right)}{\alpha^2\beta^2}.
\end{aligned}$$

We assign values to parameters α, β, δ, and γ so that these statistics approximate e_1, e_2, and l_1. First, we determine values of α, β, and δ satisfying the following equations.

$$\frac{2}{(1-\delta)\alpha + \beta} = e_1, \tag{6}$$

$$\frac{2}{\alpha\beta}\frac{(1+\delta)\alpha + \beta}{(1-\delta)\alpha + \beta} = e_2. \tag{7}$$

After that, we obtain γ that minimizes

$$\tau = \frac{(1+\delta)\,\alpha + \beta}{((1-\delta)\,\alpha + \beta)\,\alpha\beta} - \frac{\gamma\left((\alpha-\beta)^2 - \alpha^2\delta^2\right)}{((1-\delta)\,\alpha + \beta)\,\alpha^2\beta^2} - l_1.$$

If $\frac{e_2 - e_1^2}{e_1^2} \geq 1$, we assign zero to δ, and solve (6) and (7) for α and β. After that, we determine γ that minimizes τ.

However, if $\delta = 0$, then we have

$$\mathrm{cv}^2 = \frac{\mathrm{E}[Z_l^2] - \mathrm{E}[Z_l]^2}{\mathrm{E}[Z_l]^2} \geq 1.$$

Therefore, δ must be greater than zero when $\frac{e_2 - e_1^2}{e_1^2} < 1$. In what follows, we consider the case in which $0.5 < \frac{e_2 - e_1^2}{e_1^2} < 1$.

We solve (6) for β, and we have

$$\beta = \frac{2}{e_1} - (1-\delta)\,\alpha. \tag{8}$$

By replacing β by the right side of (8) in equation (7), this equation becomes

$$\frac{2\delta\alpha + \frac{2}{e_1}}{\frac{2}{e_1}\alpha - \alpha^2 + \delta\alpha^2} e_1 = e_2.$$

We solve this equation for delta. We then have

$$\delta = \frac{\alpha^2 - \frac{2}{e_1}\alpha + \frac{2}{e_2}}{\alpha^2 - \frac{2e_1}{e_2}\alpha} = \frac{\left(\alpha - \frac{1}{e_1}\right)^2 + \left(\frac{2}{e_2} - \frac{1}{e_1^2}\right)}{\alpha\left(\alpha - \frac{2e_1}{e_2}\right)}. \tag{9}$$

This equation means that if $\frac{e_2 - e_1^2}{e_1^2} < 1$, then $\delta > 0$. We obtain the minimum value of δ such that equations (6) and (7) are satisfied. Differentiate (9) with respect to α, we have

$$\frac{\mathrm{d}\delta}{\mathrm{d}\alpha} = \frac{2\left(\left(e_1^2 - e_2\right)\alpha^2 + 2e_1\alpha - \frac{2e_1^2}{e_2}\right)}{e_1 e_2\left(\alpha^2 - \frac{2e_1}{e_2}\alpha\right)^2}.$$

Therefore, the value of α that minimizes δ is obtained as follows.

$$\begin{aligned} \alpha &= \frac{e_1 + \sqrt{2\frac{e_1^4}{e_2} - e_1^2}}{e_2 - e_1^2} \\ &= \frac{1}{\mathrm{cv}^2}\left(\frac{1}{e_1} + \sqrt{\frac{1 - \mathrm{cv}^2}{e_2}}\right). \end{aligned} \tag{10}$$

We replace α by the right side of (10) in equation (6) and (7), and solve those equations for β and δ. After that, we obtain γ that minimizes τ.

5 Algorithm

We present iterative algorithms for performance evaluation of tandem queueing networks with blocking. Let $p_{k,j}^{(n)}$ denote the approximate steady-state probability that the number of customers in node-k is equal to j in the nth iteration. Also let $e_1^a(k)$, $e_2^a(k)$, and $l_1^a(k)$ denote the first two moments and the lag-1 autocorrelation of the sequence of interarrival times, respectively, in the second node of subsystem-k, and $e_1^d(k)$, $e_2^d(k)$, and $l_1^d(k)$ denote those of the sequence of service times, respectively, in the second node of subsystem-k. Our approximate method is described below.

5.1 The first node has an infinite capacity

Step 1. Initialization:

- $(\boldsymbol{C}_{1,a}, \boldsymbol{D}_{1,a}) := (\boldsymbol{C}_a, \boldsymbol{D}_a)$.
- $(\boldsymbol{C}_{k,s}, \boldsymbol{D}_{k,s}) := (\boldsymbol{C}_{k+2}, \boldsymbol{D}_{k+2}), \quad k = 1, \ldots, N-2$.
- $n := 1$.

Step 2. For $k = 1, \ldots, N-1$:

- Analyze subsystem-k.

- Derive $e_1^a(k)$, $e_2^a(k)$, and $l_1^a(k)$.
- Find MAP($\boldsymbol{C}_{k+1,a}, \boldsymbol{D}_{k+1,a}$) using the fitting procedure.

Step 3. For $k = N, \ldots, 2$:

- Analyze subsystem-k.
- Derive $e_1^d(k)$, $e_2^d(k)$, and $l_1^d(k)$.
- Find MAP($\boldsymbol{C}_{k-1,d}, \boldsymbol{D}_{k-1,d}$) using the fitting procedure.

Step 4. Convergence test:

If

$$\max_{i,j} \frac{|p_{i,j}^{(n)} - p_{i,j}^{(n-1)}|}{p_{i,j}^{(n)}} < \epsilon,$$

then stop, otherwise $n := n + 1$, and go to Step 2.

The stationary queue length distribution of node-k is approximated by that of the second node in subsystem-$k-1$ except node-1 and node-N. The stationary queue length distributions of node-1 and node-N are approximated by that of the first node in subsystem-1 and that of the third node in subsystem-$N-2$, respectively.

We do not have a proof of convergence. In our experiences, the algorithm always converged within the reasonable number of iterations unless the original queueing network is unstable. The criterion for determining convergence of the Gauss-Seidel Iteration has to be two orders magnitude less than ϵ. Let $p_{k,j}(x)$ denote the approximate steady-state probability that the number of customers in node-k is equal to j when $\epsilon = x$. In examples we examined, we have

$$\max_{i,j} \frac{|p_{i,j}(10^{-5}) - p_{i,j}(10^{-6})|}{p_{i,j}(10^{-5})} < 0.01.$$

Therefore, it seems reasonable to assume $\epsilon = 10^{-5}$.

5.2 The first node has a finite capacity

We assumed that the first node of the tandem queueing network has an infinite buffer so far. However, we can approximate the performance of tandem queueing networks whose first node has a finite buffer in the similar way. We denote by m_1 the capacity of node-1, and other assumptions do not change. The approximate algorithm for such a queueing network is as follows.

Step 1. Initialization:

- $(\boldsymbol{C}_{1,a}, \boldsymbol{D}_{1,a}) := (\boldsymbol{C}_a, \boldsymbol{D}_a)$.
- $(\boldsymbol{C}_{k,s}, \boldsymbol{D}_{k,s}) := (\boldsymbol{C}_{k+2}, \boldsymbol{D}_{k+2}), \quad k = 1, \ldots, N-2$.
- $n := 1$.

Step 2. Derive the throughput of the network $\lambda_a^{(n)}$:

- Analyze subsystem-1.
- Derive $\lambda_a^{(n)}$.
- Derive $e_1^a(1)$, $e_2^a(1)$, and $l_1^a(1)$.
- Find MAP($\boldsymbol{C}_{2,a}, \boldsymbol{D}_{2,a}$) using the fitting procedure.

Step 3. For $k = 2, \ldots, N-1$:

- Analyze subsystem-k.
- Derive $e_1^a(k)$, $e_2^a(k)$, and $l_1^a(k)$.
- Find MAP($\boldsymbol{C}_{k+1,a}, \boldsymbol{D}_{k+1,a}$) using the fitting procedure.

Step 4. For $k = N, \ldots, 2$:

- Analyze subsystem-k.
- Derive $e_1^d(k)$, $e_2^d(k)$, and $l_1^d(k)$.
- Find MAP($\boldsymbol{C}_{k-1,d}, \boldsymbol{D}_{k-1,d}$) using the fitting procedure.

Step 5. Convergence test:

If

$$\max_{i,j} \frac{|p_{i,j}^{(n)} - p_{i,j}^{(n-1)}|}{p_{i,j}^{(n)}} < \epsilon,$$

then stop, otherwise $n := n + 1$, and go to Step 2.

6 Numerical experiments

In the following experiments, we use two-state MAP as the arrival process. The service process in each node is two-state MSP. The average queue lengths in nodes obtained by our method and simulations are represented in the following figures. The simulation time was greater than 10^7 times the largest mean service time of the system. We simulated 30 times, and displayed 95% confidence interval in the figures. Also for comparison, we plotted the results obtained by Algorithm-4 in [12], which does not consider the correlations in the arrival and service processes.

Example 1

We start with an example with no correlated arrivals and services, where $N = 10$, and $m_k = 4$ for $k = 2, \ldots, N$. The first node has an infinite capacity. The arrival process is MAP($\boldsymbol{C}_a, \boldsymbol{D}_a$):

$$\boldsymbol{C_a} = \begin{bmatrix} -0.02 & 0.00 \\ 0.00 & -0.13 \end{bmatrix}, \quad \boldsymbol{D_a} = \begin{bmatrix} 0.0026 & 0.0173 \\ 0.0173 & 0.1126 \end{bmatrix},$$

and the service process of node-k is MSP($\boldsymbol{C}_k, \boldsymbol{D}_k$):

$$\boldsymbol{C}_k = \begin{bmatrix} -0.15 & 0.00 \\ 0.00 & -0.45 \end{bmatrix}, \quad \boldsymbol{D}_k = \begin{bmatrix} 0.0375 & 0.1125 \\ 0.1125 & 0.3375 \end{bmatrix}, \quad k = 1, \ldots, N.$$

We denote by $(X_1, X_2, \ldots)$ a sequence of interarrival or service times of the process. The statistics of the sequence of interarrival times and that of service times are shown in Table 1.

Table 1: The statistics of the sequence of times.

	$\mathrm{E}[X_l]$	$\mathrm{Var}[X_l]$	$\mathrm{Cov}[X_l, X_{l+1}]$
MAP($\boldsymbol{C}_a, \boldsymbol{D}_a$)	13.33	591.5	0.000
MSP($\boldsymbol{C}_k, \boldsymbol{D}_k$)	3.333	18.52	0.000

The results for Example 1 are shown in Figure 3.

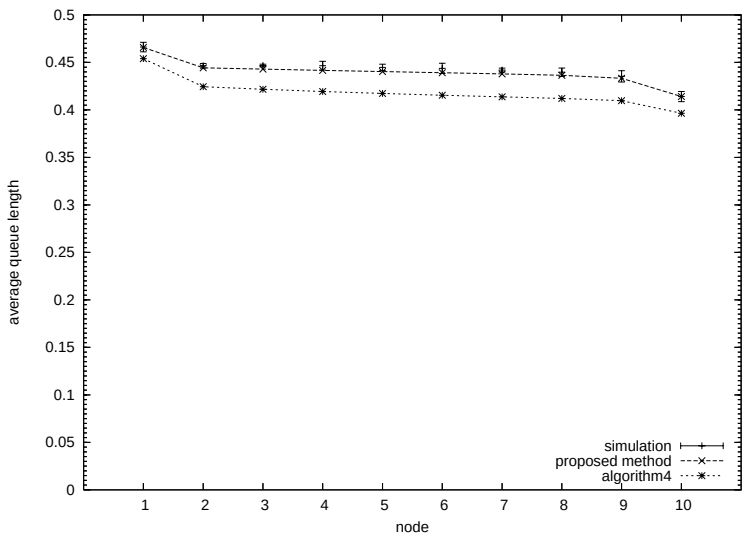

Figure 3: With no correlated arrivals and services.

Figure 3 shows that both the proposed method and Algorithm-4 work well.

Example 2

We now consider a situation where the arrival process is correlated. Example 2 has the following parameters: $N = 10$, and $m_k = 4$ for $k = 2, \ldots, N$. The first node has an infinite capacity. In addition, the service processes in each node is identical with that of Example 1. Example 2 differs from Example 1 in the arrival process. The arrival process is MAP($\boldsymbol{C}_a, \boldsymbol{D}_a$):

$$\boldsymbol{C_a} = \begin{bmatrix} -0.02 & 0.00 \\ 0.00 & -0.13 \end{bmatrix}, \quad \boldsymbol{D_a} = \begin{bmatrix} 0.01827 & 0.001733 \\ 0.001733 & 0.1283 \end{bmatrix}.$$

The statistics of the sequence of interarrival times are shown in Table 2.

Table 2: The statistics of the sequence of interarrival times.

	$\mathrm{E}[X_l]$	$\mathrm{Var}[X_l]$	$\mathrm{Cov}[X_l, X_{l+1}]$
$\mathrm{MAP}(\boldsymbol{C}_a, \boldsymbol{D}_a)$	13.33	591.5	186.2

Note that the mean and variance of interarrival times of Example 2 are identical with those of Example 1, while the covariance differs. Figure 4 shows the results obtained for Example 2.

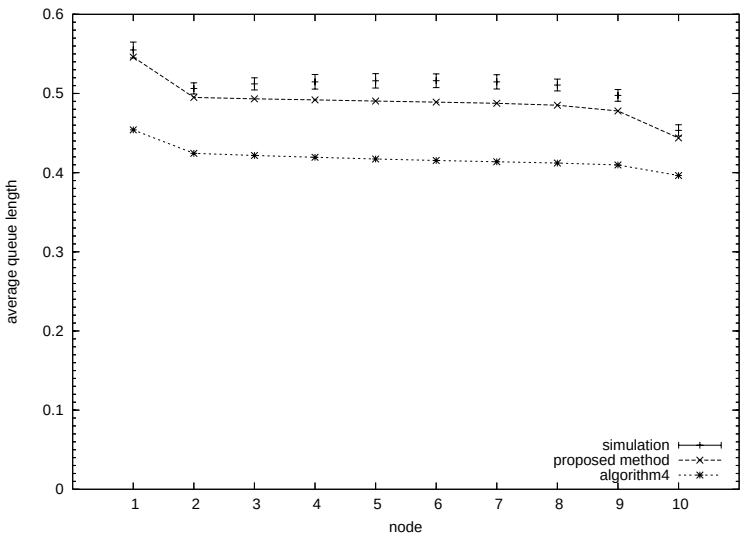

Figure 4: With correlated arrivals.

The data obtained by simulations in Figure 3 and Figure 4 demonstrates the effect of correlation in arrivals. The correlated arrival process raises the average queue lengths of nodes. The proposed method reflects this effect, while Algorithm-4 obtains the same results as Example 1. However, it should be noted that the approximate average queue length obtained by our method are lower than the results of simulations. We believe that it is because of the correlation between arrivals and services in each node. In a node of the tandem queueing network with blocking, if many arrivals occur in a short time, the service times tend to be long because of the blocking. In our method, we assumed that the service process of the third node in each subsystem is independent of arrivals.

Example 3

Here, we examine the effect of the correlations in the service processes of nodes. In Example 3, there are $N = 10$ servers. The first node has an infinite capacity, and the capacity of node-k is $m_k = 4$ for $k = 2, \ldots, N$. The arrival process is identical with that

of Example 1, and the service process in node-k is MSP$(\boldsymbol{C}_k, \boldsymbol{D}_k)$:

$$\boldsymbol{C}_k = \begin{bmatrix} -0.15 & 0.00 \\ 0.00 & -0.45 \end{bmatrix}, \quad \boldsymbol{D}_k = \begin{bmatrix} 0.1388 & 0.01125 \\ 0.01125 & 0.4388 \end{bmatrix}, \quad k = 1, \ldots, N.$$

The mean, variance and covariance of the sequence of service times in node-k are shown in Table 3.

Table 3: The statistics of the sequence of service times.

	$\mathrm{E}[X_l]$	$\mathrm{Var}[X_l]$	$\mathrm{Cov}[X_l, X_{l+1}]$
MAP$(\boldsymbol{C}_k, \boldsymbol{D}_k)(k = 1, \ldots, N)$	3.333	18.52	3.333

Note that the mean and variance of the sequence of service times in each node are identical with those of Example 1.

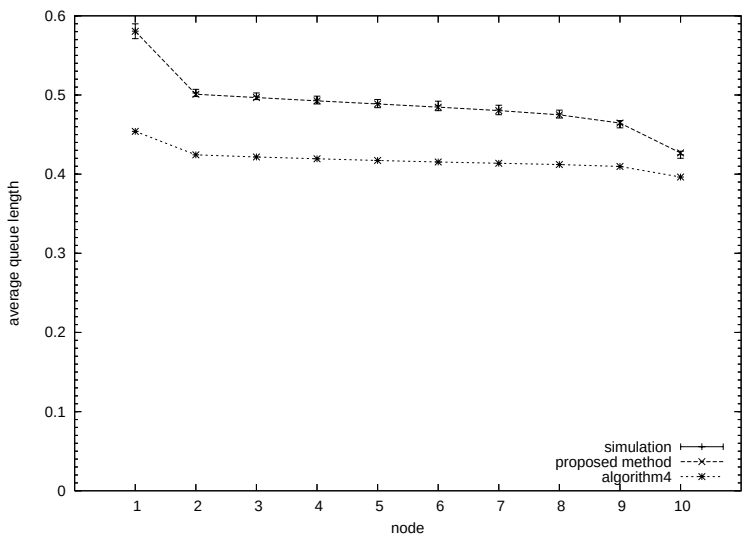

Figure 5: With correlations in services of all nodes.

The results of simulations in Example 1 and Example 3 show that the correlations in service processes have the effect on the expected queue lengths in nodes. The results obtained by the proposed method are very close to those obtained by simulations.

Example 4

There are $N = 10$ nodes in the network. The first node has an infinite capacity, and $m_k = 4$ for $k = 2, \ldots, N$. The moments of service times are identical among all nodes. However, only the service process in node-5 is correlated. The service processes are MSPs represented as follows.

$$\boldsymbol{C}_5 = \begin{bmatrix} -0.15 & 0.00 \\ 0.00 & -0.45 \end{bmatrix}, \quad \boldsymbol{D}_5 = \begin{bmatrix} 0.1388 & 0.01125 \\ 0.01125 & 0.4388 \end{bmatrix},$$

$$\boldsymbol{C}_k = \begin{bmatrix} -0.15 & 0.00 \\ 0.00 & -0.45 \end{bmatrix}, \quad \boldsymbol{D}_k = \begin{bmatrix} 0.0375 & 0.01125 \\ 0.01125 & 0.3375 \end{bmatrix}, \quad k = 1, \ldots, 4, 6, \ldots, N.$$

The arrival process is identical with that of Example 1. The statistics of service times are shown in Table 4. The average queue length in each node is shown in Figure 6.

Table 4: The statistics of the sequence of service times.

	$\mathrm{E}[X_l]$	$\mathrm{Var}[X_l]$	$\mathrm{Cov}[X_l, X_{l+1}]$
$\mathrm{MSP}(\boldsymbol{C}_5, \boldsymbol{D}_5)$	3.333	18.52	3.333
$\mathrm{MSP}(\boldsymbol{C}_k, \boldsymbol{D}_k)(k = 1, \ldots, 4, 6, \ldots, N)$	3.333	18.52	0.000

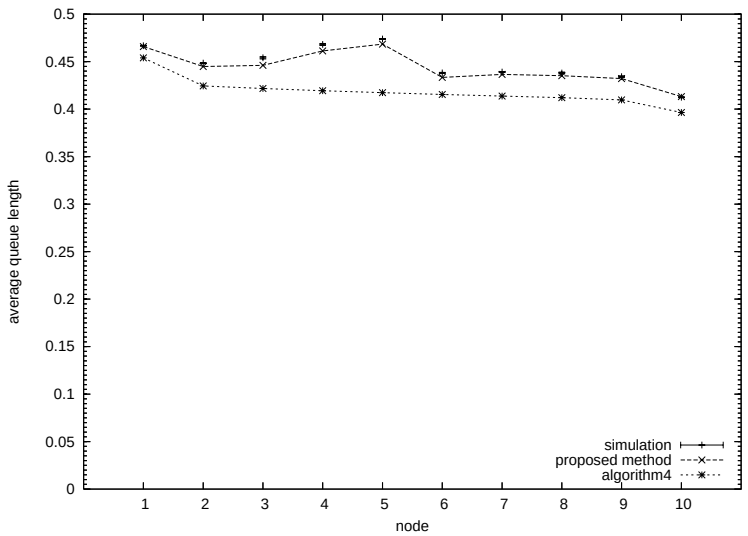

Figure 6: With a correlation in services of node-5.

Figure 6 shows that the correlation in the service process of node-5 raises the average queue lengths of node-5 and its upstream nodes. The proposed method reflects the effect of the correlation.

Example 5

We have presented examples in which the capacity of the first node is infinite. Recall that, we also developed the algorithm for tandem queueing networks whose first node has a finite capacity. In Example 5, the first node has a capacity of $m_1 = 4$, and other conditions are identical with those of Example 2. The results for Example 5 are shown in Figure 7.

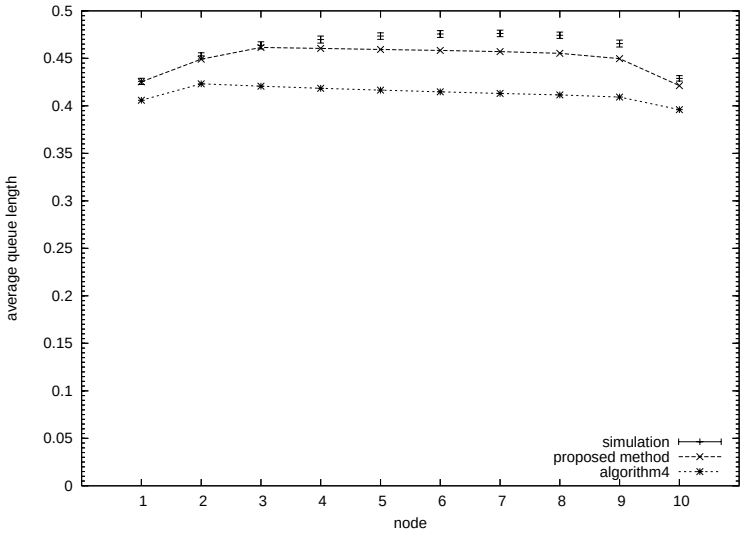

Figure 7: The first node has a finite capacity.

Figure 7 suggests that our method for tandem queueing networks whose first node has a finite capacity performs accurately.

7 Conclusion

We proposed an approximation method for performance evaluation of tandem blocking queueing networks with correlated arrivals and services. The method is based on the method of decomposition. The algorithm decomposes the tandem queueing network into several subsystems and separately analyzes each subsystem. Also, we developed an efficient algorithmic procedure for analyzing each subsystem.

The results of simulation suggested that correlations in arrivals and services had a great impact on the performance of tandem queueing networks with blocking. Through numerical experiments, we examined the accuracy of the approximation and confirmed that our method well approximated the performance of tandem queueing networks with correlated arrivals and services.

Acknowledgement

First of all, I wish to thank Professor Tetsuya Takine for his invaluable guidance, support and encouragement throughout my studies. I would like to thank Professor Masao Fukushima who has given me careful advice and guidance. I am also very grateful to members of Professor Fukushima's laboratory for their camaraderie.

References

[1] Tayfur Altiok and Benfamin Melamed (2001) "The case for modeling correlation in manufacturing systems," IIE Transactions, 33, 779–791.

[2] Søren Asmussen and Ger Koole (1993) "Marked point processes as limits of Markovian arrival streams," Journal of Applied Probability, 30, 365–372.

[3] Simonetta Balsamo (2000) "Closed queueing networks with finite capacity queues: approximate analysis," Proceedings of the 14th European Simulation Multiconference on Simulation and Modelling: Enablers for a Better Quality of Life, 593–600.

[4] Simonetta Balsamo (2003) "A review on queueing network models with finite capacity queues for software architectures performance prediction," Performance Evaluation, 51, 269–288.

[5] Alexandre Brandwajn and Yung-Li Lily Jow (1988) "An approximation method for tandem queues with blocking," Operations Research, 36, 73–83.

[6] Muckai K. Girish and Jian-Qian Hu (2000) "Higher order approximation for the single server queue with splitting, merging and feedback," European Journal of Operational Research, 124, 447–467.

[7] Armin Heindl (2003) "Decomposition of general tandem queueing networks with MMPP inputs and customer losses," Performance Evaluation, 51, 117–136.

[8] Kyung P. Jun and Harry G. Perros (1990) "An approximate analysis of open tandem queueing networks with blocking and general service times," European Journal of Operational Research, 46, 123–135.

[9] G. Latouche and V. Ramaswami (1999) "Introduction to Matrix Analytic Methods in Stochastic Modeling," American Statistical Association, Alexandria, Virginia.

[10] David M. Lucantoni, Kathleen S. Meier-Hellstern, and Marcel F. Neuts (1990) "A single-server queue with server vacations and a class of non-renewal arrival processes," Advances in Aplied Probability, 22, 676–705.

[11] K. Mitchell and A. van de Liefvoort (2003) "Approximation models of feed-forward G/G/1/N queueing networks with correlated arrivals," Performance Evaluation, 51, 137–152.

[12] Harry G. Perros (1994) "Queueing Networks With Blocking," Oxford University Press, New York, Oxford.

A The generator of the underlying Markov chain of a subsystem

For the analysis of subsystem-k, we describe the generator of the underlying Markov chain. In what follows, we omit the subscript k. Let MAP$(\boldsymbol{C}_a, \boldsymbol{D}_a)$ denote the arrival process to the subsystem, and MSP$(\boldsymbol{C}_i, \boldsymbol{D}_i)$ denote the service process in the ith node for $i = 1, 2, 3$. We assume that when the first node is full, as soon as a departure occurs

in the first node, a customer arrives at the first node with probability q. The value of q is chosen such that the throughput of the subsystem is equal to that of the original network. Other parameters are as follows.

m_i : The capacity of the ith node.
n_i : The number of customers in the ith node.
a : The state of the arrival process.
p_i : The state of the service process in the ith node.

We denote by $(n_2, n_1, n_3, P) = (n_2, n_1, n_3, a, p_1, p_2, p_3)$ the state of the subsystem. Let Ξ denote the whole state set of (n_2, n_1, n_3, P), and let Φ denote the Cartesian product of the state sets of the arrival and service processes. Ξ is defined as follows.

$$
\begin{aligned}
&1 \le n_1 \le m_1, 1 \le n_2 \le m_2 + 1, 0 \le n_3 \le m_3 + 1, P \in \Phi, \\
&1 \le n_1 \le m_1, n_2 = 0, 0 \le n_3 \le m_3, P \in \Phi, \\
&n_1 = 0, 1 \le n_2 \le m_2, 0 \le n_3 \le m_3 + 1, P \in \Phi, \\
&n_1 = 0, n_2 = 0, 0 \le n_3 \le m_3, P \in \Phi.
\end{aligned}
$$

Note that (n_2, n_1, n_3, S) is the continuous time Markov chain on Ξ. We denote by $\boldsymbol{Q}$ the generator of the underlying Markov chain. $\boldsymbol{Q}$ is an $(m_2 + 2) \times (m_2 + 2)$ block matrix that takes a form:

$$
\boldsymbol{Q} = \begin{bmatrix}
\boldsymbol{B}_1 & \boldsymbol{B}_2 & & & & \\
\boldsymbol{B}_0 & \boldsymbol{A}_1 & \boldsymbol{A}_2 & & \boldsymbol{O} & \\
 & \boldsymbol{A}_0 & \boldsymbol{A}_1 & \ddots & & \\
 & & \ddots & \ddots & \boldsymbol{A}_2 & \\
 & \boldsymbol{O} & & \boldsymbol{A}_0 & \boldsymbol{A}_1 & \boldsymbol{H}_2 \\
 & & & & \boldsymbol{H}_0 & \boldsymbol{P}_2
\end{bmatrix}.
$$

We describe the structure of each block element matrix below.

$\boldsymbol{A}_2$ includes the rates corresponding to the increase of the number of customers in the second node, that is, completions of services in the first node. $\boldsymbol{A}_2$ is an $(m_1+1)\times(m_1+1)$ block matrix given below.

$$
\boldsymbol{A}_2 = \begin{bmatrix}
\boldsymbol{O} & & & \boldsymbol{O} & \\
\boldsymbol{A}_0^{(1)} & \ddots & & & \\
 & \ddots & \ddots & & \\
 & & \boldsymbol{A}_0^{(1)} & \boldsymbol{O} & \\
 & \boldsymbol{O} & & (1-q)\boldsymbol{A}_0^{(1)} & q\boldsymbol{A}_0^{(1)}
\end{bmatrix},
$$

with

$$
\boldsymbol{A}_0^{(1)} = \boldsymbol{I}_{n_3} \otimes \boldsymbol{I}_a \otimes \boldsymbol{D}_1 \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_{p_3},
$$

where $\boldsymbol{I}_\cdot$ denotes an identity matrix with appropriate dimension, and $\otimes$ denotes a Kronecker product [9].

$\boldsymbol{A}_0$ includes the rates corresponding to the decrease of the number of customers in the second node, that is, completions of services in the second node. $\boldsymbol{A}_0$ is an $(m_1+1)\times(m_1+1)$ block matrix that is given by

$$\boldsymbol{A}_0 = \begin{bmatrix} \boldsymbol{A}'_0 & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & \boldsymbol{A}'_0 \end{bmatrix},$$

where $\boldsymbol{A}'_0$ is an $(m_3+2)\times(m_3+2)$ block matrix.

$$\boldsymbol{A}'_0 = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{A}_2^{(3)} & & & \boldsymbol{O} \\ & \ddots & \ddots & & \\ & & \ddots & \boldsymbol{A}_2^{(3)} & \\ & & & \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{O} & & & \boldsymbol{A}_0^{(3)} & \boldsymbol{O} \end{bmatrix},$$

where

$$\begin{aligned} \boldsymbol{A}_2^{(3)} &= \boldsymbol{I}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{D}_2 \otimes \boldsymbol{I}_{p_3}, \\ \boldsymbol{A}_0^{(3)} &= \boldsymbol{I}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{D}_3. \end{aligned}$$

$\boldsymbol{A}_0^{(3)}$ represents unblocking of the second node, that is, departures from the third node.

$\boldsymbol{A}_1$ is an $(m_1+1)\times(m_1+1)$ block matrix that takes a form:

$$\boldsymbol{A}_1 = \begin{bmatrix} \boldsymbol{B}_1^{(1)} & \boldsymbol{A}_2^{(1)} & & & \\ & \boldsymbol{A}_1^{(1)} & \boldsymbol{A}_2^{(1)} & \boldsymbol{O} & \\ & & \ddots & \ddots & \\ & \boldsymbol{O} & & \boldsymbol{A}_1^{(1)} & \boldsymbol{A}_2^{(1)} \\ & & & & \boldsymbol{P}_1^{(1)} \end{bmatrix}.$$

$\boldsymbol{A}_2^{(1)}$ includes the rates corresponding to arrivals to the first node. We then have

$$\boldsymbol{A}_2^{(1)} = \boldsymbol{I}_{n_3} \otimes \boldsymbol{D}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_{p_3}.$$

$\boldsymbol{A}_1^{(1)}$ is an $(m_3+2)\times(m_3+2)$ block matrix.

$$\boldsymbol{A}_1^{(1)} = \begin{bmatrix} \boldsymbol{B}_1^{(3)} & & & & \\ \boldsymbol{A}_0^{(3)} & \boldsymbol{A}_1^{(3)} & & \boldsymbol{O} & \\ & \ddots & \ddots & & \\ & & \boldsymbol{A}_0^{(3)} & \boldsymbol{A}_1^{(3)} & \boldsymbol{A}_2^{(3)} \\ & \boldsymbol{O} & & \boldsymbol{O} & \boldsymbol{P}_3^{(3)} \end{bmatrix}.$$

Each diagonal block in $\boldsymbol{A}_1^{(1)}$ corresponds to transitions in states of the arrival process and service processes. We then have

$$\begin{aligned}
\boldsymbol{A}_1^{(3)} &= \boldsymbol{C}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_{p_3} + \boldsymbol{I}_a \otimes \boldsymbol{C}_1 \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_{p_3} \\
&\quad + \boldsymbol{I}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{C}_2 \otimes \boldsymbol{I}_d + \boldsymbol{I}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{C}_3, \\
\boldsymbol{P}_3^{(3)} &= \boldsymbol{A}_1^{(3)} - \boldsymbol{I}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{C}_2 \otimes \boldsymbol{I}_{p_3}, \\
\boldsymbol{B}_1^{(3)} &= \boldsymbol{A}_1^{(3)} - \boldsymbol{I}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{C}_3.
\end{aligned}$$

If $n_1 = m_1 + 1$, then arriving customers are lost. If $n_1 = 0$, then the state of the service process in the first node does not change. Therefore,

$$\begin{aligned}
\boldsymbol{P}_1^{(1)} &= \boldsymbol{A}_1^{(1)} + \boldsymbol{A}_2^{(1)}, \\
\boldsymbol{B}_1^{(1)} &= \boldsymbol{A}_1^{(1)} - \boldsymbol{I}_{n_3} \otimes \boldsymbol{I}_a \otimes \boldsymbol{C}_1 \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_{p_3}.
\end{aligned}$$

Note that if $n_1 = 0$, then n_2 can not be $m_2 + 1$ and that if $n_2 = m_2 + 1$, then the first node is blocked. Therefore, $\boldsymbol{P}_2$ is an $m_1 \times m_1$ block matrix given by

$$\boldsymbol{P}_2 = \begin{bmatrix}
\boldsymbol{A}_1^{(1)\prime\prime} & \boldsymbol{A}_2^{(1)} & & \\
& \ddots & \ddots & \\
& & \boldsymbol{A}_1^{(1)\prime\prime} & \boldsymbol{A}_2^{(1)} \\
& & & \boldsymbol{P}_1^{(1)\prime\prime}
\end{bmatrix},$$

where

$$\begin{aligned}
\boldsymbol{A}_1^{(1)\prime\prime} &= \boldsymbol{A}_1^{(1)} - \boldsymbol{I}_{n_3} \otimes \boldsymbol{I}_a \otimes \boldsymbol{C}_1 \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_{p_3}, \\
\boldsymbol{P}_1^{(1)\prime\prime} &= \boldsymbol{P}_1^{(1)} - \boldsymbol{I}_{n_3} \otimes \boldsymbol{I}_a \otimes \boldsymbol{C}_1 \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_{p_3}.
\end{aligned}$$

If $n_1 = 0$, then n_2 can not be $m_2 + 1$. Therefore, $\boldsymbol{H}_2$ is an $(m_1 + 1) \times m_1$ block matrix and $\boldsymbol{H}_0$ is an $m_1 \times (m_1 + 1)$ block matrix:

$$\boldsymbol{H}_2 = \begin{bmatrix}
\boldsymbol{O} & \cdots & \boldsymbol{O} \\
\boldsymbol{A}_0^{(1)} & & \boldsymbol{O} \\
& \ddots & \\
\boldsymbol{O} & & \boldsymbol{A}_0^{(1)}
\end{bmatrix}, \quad \boldsymbol{H}_0 = \begin{bmatrix}
\boldsymbol{A}_0' & \boldsymbol{O} & & \\
& \ddots & \ddots & \\
& & \boldsymbol{A}_0' & \boldsymbol{O} \\
& \boldsymbol{O} & & (1-q)\boldsymbol{A}_0' \quad q\boldsymbol{A}_0'
\end{bmatrix}.$$

$\boldsymbol{B}_1$ is an $(m_1 + 1) \times (m_1 + 1)$ block matrix given by

$$\boldsymbol{B}_1 = \begin{bmatrix}
\boldsymbol{B}_1^{(1)} & \boldsymbol{A}_2^{(1)\prime} & & & \boldsymbol{O} \\
& \boldsymbol{A}_1^{(1)\prime} & \boldsymbol{A}_2^{(1)\prime} & & \\
& & \ddots & \ddots & \\
& & & \boldsymbol{A}_1^{(1)\prime} & \boldsymbol{A}_2^{(1)\prime} \\
\boldsymbol{O} & & & & \boldsymbol{P}_1^{(1)\prime}
\end{bmatrix}.$$

If $n_2 = 0$, then n_3 can not be $m_3 + 1$ and the state of the service process in the second node does not change. Therefore, we have

$$\boldsymbol{A}_1^{(1)'} = \begin{bmatrix} \boldsymbol{B}_1^{(3)} & & & \boldsymbol{O} \\ \boldsymbol{A}_0^{(3)} & \boldsymbol{A}_1^{(3)} & & \\ & \ddots & \ddots & \\ \boldsymbol{O} & & \boldsymbol{A}_0^{(3)} & \boldsymbol{A}_1^{(3)} \end{bmatrix} - \boldsymbol{I}_{n_3-1} \otimes \boldsymbol{I}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_d,$$

where $\boldsymbol{I}_{n_3-1}$ denotes an identity matrix with $m_3 + 1$ dimension. Furthermore,

$$\begin{aligned} \boldsymbol{B}_1^{(1)'} &= \boldsymbol{A}_1^{(1)'} - \boldsymbol{I}_{n_3-1} \otimes \boldsymbol{I}_a \otimes \boldsymbol{C}_1 \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_d, \\ \boldsymbol{P}_1^{(1)'} &= \boldsymbol{A}_1^{(1)'} + \boldsymbol{I}_{n_3-1} \otimes \boldsymbol{D}_a \otimes \boldsymbol{I}_{p_1} \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_d. \end{aligned}$$

$\boldsymbol{B}_2$ and $\boldsymbol{B}_0$ are similar to $\boldsymbol{A}_2$ and $\boldsymbol{A}_0$, respectively, and described as follows.

$$\boldsymbol{B}_2 = \begin{bmatrix} \boldsymbol{O} & & & \boldsymbol{O} & & \\ \boldsymbol{A}_0^{(1)'} & \ddots & & & & \\ & \ddots & \ddots & & & \\ & & \boldsymbol{A}_0^{(1)'} & \boldsymbol{O} & & \\ & \boldsymbol{O} & & (1-q)\boldsymbol{A}_0^{(1)'} & q\boldsymbol{A}_0^{(1)'} \end{bmatrix},$$

where

$$\boldsymbol{A}_0^{(1)'} = \begin{bmatrix} 1 & & \boldsymbol{O} & 0 \\ & \ddots & & \vdots \\ \boldsymbol{O} & & 1 & 0 \end{bmatrix} \otimes \boldsymbol{I}_a \otimes \boldsymbol{D}_1 \otimes \boldsymbol{I}_{p_2} \otimes \boldsymbol{I}_{p_3},$$

and

$$\boldsymbol{B}_0 = \begin{bmatrix} \boldsymbol{A}_0'' & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & \boldsymbol{A}_0'' \end{bmatrix}, \quad \boldsymbol{A}_0'' = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{A}_2^{(3)} & & \boldsymbol{O} \\ & \ddots & \ddots & \\ & & \ddots & \boldsymbol{A}_2^{(3)} \\ & & & \boldsymbol{O} \\ \boldsymbol{O} & & & \boldsymbol{A}_0^{(3)} \end{bmatrix}.$$