Queueing Models and
Service Management
Vol. 8, No. 3, page 59–90, 2025

QMSM
©AU 2023

A Note on Computing Approach Toward Two-tier Service Models

Hsing Paul Luh[1,*] and Zhe George Zhang[2]

[1]Department of Mathematical Science
Center for Computational Research and Applications
National Chengchi University,
Taipei 116, Taiwan

[2]Department of Decision Sciences
Western Washington University
Bellingham, WA, USA

(*Received April 2025; accepted July 2025*)

Abstract: This paper presents a new algorithm for computing the performance measure of a two-tier service queueing model. In such a system, one service provider offers service with unlimited waiting space and the other offers a finite waiting space. Due to the two queue feature, the system is formulated as a state dependent quasi-birth-and-death (QBD) process. The customer choice behavior and the observable queues make the QBD process to have a large number of boundary states. Such a structure motivates us to develop a more efficient algorithm than the classical rate matrix iteration algorithms. With the special structure of the infinitesimal generator matrix for the two-tier service system, we propose a more efficient and innovative K-matrix based algorithm for computing the stationary distribution. As the buffer size increases, the improved accuracy and computational efficiency of the K-matrix method become significant compared with the classical Geometric-Matrix method. We demonstrate the advantages of the new algorithm with numerical examples.

Keywords: Finite buffer queue, quasi-birth-death process, two service channels.

1. Introduction

In this paper, we consider a queueing system with two service providers (SPs) for customers to choose. One SP offers free service with unlimited waiting space and the other SP offers a toll service and only admits a finite maximum number of customers to wait. There are waiting costs for customers in both lines. We develop a computational model for evaluating the impact of the customer choice on the performance measures. We formulate a state dependent quasi-birth-and-death (QBD) process for such a two-tier service system. By exploring the structure of the infinite generator of the QBD process, a new K-matrix

[*] Corresponding author
Email: slu@nccu.edu.tw

based algorithm is proposed for computing the stationary distribution more efficiently and accurately compared with the classical rate matrix iteration algorithms.

The model considered in this paper belongs to a class of service systems with customer choice and delay. For a comprehensive survey in this area, see Hassin[11]. Most of the past studies focused on single-tier free or toll service system with customer choice under different information scenarios (observable or unobservable queue). There are some recent works on the two-tier service. Guo et al. [10] considered a two tier service system where customers make their joining decisions on toll or free channel based on long term statistics without real-time information of queue lengths. Hua et al. [13] studied the competition and coordination in a two-tier service system with customer choice behavior. Again they assumed that the customers make their decisions based on long-term statistics rather than real-time queue length information. Chen et al. [2] conducted an empirical analysis on the two-tier system based on the real data set verified by a two-queue model without real-time delay information for customers. The real-time queue length information will make the arrival process depend on the state of the system. Such a dependence will greatly complicate the analysis and make some analytical methods mathematically intractable. However, there do exist some practical situations where customers make their service selection according to real-time queue lengths of the two service providers. Unfortunately, to the best of our knowledge, there is very few studies in the literature focusing on a more realistic two-tier service system with *heterogeneous delay sensitive* customers and real-time queue length information. Formulating the system as an quasi-birth-and-death (QBD) process is possible, but the computing the stationary distribution with a realistic buffer size and traffic intensity (usually quite high) can become an issue. Our focus is on developing a more efficient algorithm to compute the stationary performance measures for the two-tier service system with customer choice and the real time queue length information (observable queues). Specifically, we consider the case where heterogeneous customers, after joining the queue, are not allowed to switch between the two service providers. This system has been modeled as a two-dimensional state space QBD process. Since the infinitesimal generator matrix of the QBD process has a special structure due to customer choice, an innovative K-matrix based algorithm is proposed to solve for the stationary distribution. With such an algorithm, we can compute the performance measures more efficiently and accurately compared with the classical rate matrix iteration algorithms.

In literature, a number of methods were suggested to solve QBD models with special structures. For example, Latouche and Neuts [15] used matrix analytic methods and Konheim and Reiser [14] proposed generating function methods. In particular, Grassmann [6] applied generalized eigenvalues to analyze certain tridiagonal matrix polynomials. Such a method is used for counting the number of sign changes in the Sturm sequence involving eigenvalues. Solving QBD models by a generalized eigenvalue method has been used successfully in related models in Grassmann and Drekic [8], Drekic and Grassmann [4], and Grassmann [7]. These studies showed the instances where eigenvalue methods are much more efficient than other methods. In addition, Grassmann and Tavakoli [9] also provided a survey paper comparing different approaches dealing with QBD models with low-rank sub-

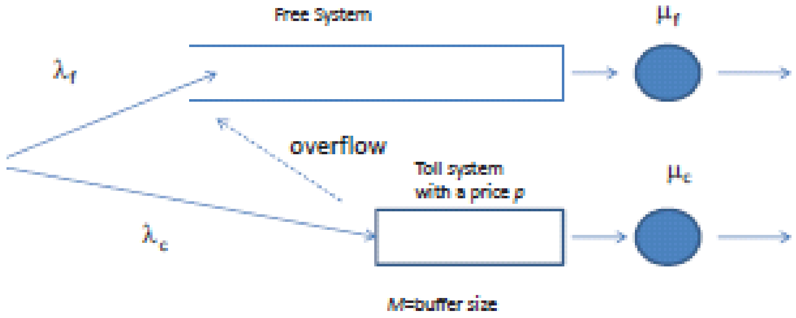

Figure 1. A Two-tier Service System with Strategic Customers

matrices. Investigating the property of low-rank matrices, we derive in this paper an explicit equation for solving the eigenvalue which is independent of the number of boundary states, implying that the computational effort is reduced significantly to an order of a linear function of the size of a submatrix.

The paper is organized as follows. In Section 2, we formulate a QBD model for a two-tier service system, a class of service systems with customer options. In Section 3 we present a new and efficient approach with an eigenvalue to solve the stationary probability distribution of the queue sizes for the two-tier service system. In Section 4, we compare the computational efficiencies of the two solution schemes that solves the model numerically. The paper is concluded with a summary in Section 5.

2. A Two-tier Service Model

Consider a two-tier service system offering both fast toll service and regular free service to customers arriving according to a Poisson process with rate Λ as shown in Figure 1. When a customer arrives at the system, he or she is provided with the real-time queue length information for both queues (observable queues). To ensure that the expected waiting time of a customer choosing the toll service is no more than an upper bound, we assume that the toll queue has a finite buffer of size M. For a two-tier healthcare system, the toll system may represent a private hospital which usually emphasizes the fast service delivery with a guaranteed maximum delay. In practice, limiting the number of waiting customers can also improve the service efficiency as avoiding overly long wait list usually reduces or eliminates the ”no-show” rate. We assume that whenever the buffer is full, the customer has to join the free lane. This assumption is reasonable if the customer’s service utility is large enough. We also assume that the service times are exponentially distributed with rates μ_f and μ_c for the free and toll SPs, respectively. The customer time value (waiting cost) parameter, denoted by θ, is randomly distributed with a cumulative distribution function of F_θ. A customer makes the SP selection based on the expected costs of the two queues and becomes indifferent if the following condition holds at an arrival instant t:

$$\theta X_f(t)(1/\mu_f) = p + \theta X_c(t)(1/\mu_c), \tag{1}$$

where $X_f(t)$ and $X_c(t)$ are the queue lengths (including the customer in service) of the free and toll systems, respectively, and p is the toll price. We assume that an arriving customer makes the following decisions: if $\theta X_f(t)(1/\mu_f) \le p + \theta X_c(t)(1/\mu_c)$, he chooses the free system; if $\theta X_f(t)(1/\mu_f) > p + \theta X_c(t)(1/\mu_c)$ and $X_c(t) < M$, he joins the toll system; otherwise, he will join the free system. Note that here we assume that all customers will get service or there is no balking. The system state is defined as $(X_f(t), X_c(t))$ on the state space

$$\Omega = \{(n, m) : n = 0, 1, ...; m = 0, 1, ..., M\}$$

Under the stability condition (Proposition 2.1), the system reaches the steady-state. That is $\lim_{t\to\infty} P\{X_f(t) = n, X_c(t) = m) = p_{nm}$. Denote the equilibrium arrival rates to the free and the toll systems as $\lambda_f(n, m)$ and $\lambda_c(n, m)$, respectively, at state (n, m). Obviously, based on customer choice, we have

$$\begin{aligned}
\lambda_f(n,m) &= \Lambda F_\theta(\frac{p}{n/\mu_f - m/\mu_c}), \\
\lambda_c(n,m) &= \Lambda\left(1 - F_\theta(\frac{p}{n/\mu_f - m/\mu_c})\right),
\end{aligned}$$

for states with $m < M$. In this model, we assume that both free and toll services are identical in service quality. The only competitive advantage for the toll system is that the expected waiting time is upper bounded compared with the free system. This advantage can be, as mentioned above, modeled as a queue with a finite buffer size M. Note that since m is bounded by M, as n becomes very large or the free queue is very long, $\lambda_f(n, m)$ will approach to zero as long as the toll system buffer is not full ($m < M$). Thus due to the customer's self-interest choice behavior, there exists a threshold value for the free queue, denoted by n_0 such that $\lambda_f(n, m) < \varepsilon$ whenever $n > n_0$ and $m < M$, where ε is a small positive value. The lower bound n_0 can be determined by

$$n_0 > \frac{p\mu_f}{F_\theta^{-1}(\varepsilon/\Lambda)} + \frac{\mu_f}{\mu_c}M.$$

We use the uniform distribution F_θ over $(0, U)$ to develop a Quasi-Birth and Death (QBD) model (other distributions can be used at the expense of more complicated formulas). It is easy to find that in this case

$$n_0 = \text{int}\left(\frac{p}{U}\frac{\Lambda\mu_f}{\varepsilon} + (M-1)\frac{\mu_f}{\mu_c}\right). \tag{2}$$

It is thus reasonable to assume that if the free system is highly congested and the toll system is still not full (i.e., $n > n_0, m < M$), all customers will join the toll system. Using this

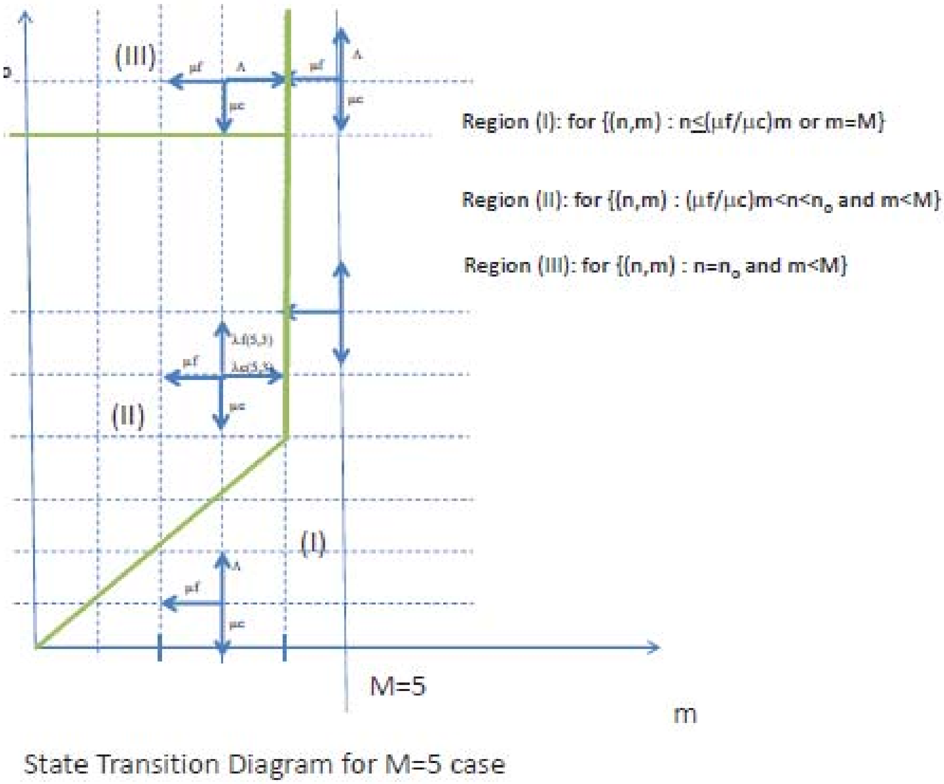
(III)
μf
Λ
μc
Λ
μf
μc
Region (I): for {(n,m) : n≤(μf/μc)m or m=M}
Region (II): for {(n,m) : (μf/μc)m<n<n₀ and m<M}
Region (III): for {(n,m) : n=n₀ and m<M}
λf(5,3)
μf
λc(5,3)
μc
(II)
(I)
Λ
μf
μc
M=5
m
State Transition Diagram for M=5 case

Figure 2. State Transition Diagram for a Two-tier Service System with Real-time Delay Information

property, we can develop a level independent QBD process to model the two-tier service system with any desired accuracy (ε value). The buffer size of the toll system, M, determines the number of phases for each state and n_0 determine the number of boundary states of the QBD process. The arrival rates to free and toll systems, respectively, can be written as

$$\begin{aligned}
\lambda_f(n,m) &= \begin{cases} \Lambda & \text{if } n \leq m \times (\mu_f/\mu_c) \quad \text{or } m = M, \\ \frac{\Lambda}{U}\left(\frac{p\mu_f\mu_c}{n\mu_c - m\mu_f}\right) & \text{if } m \times (\mu_f/\mu_c) < n < n_0 \text{ and } m < M, \\ 0 & \text{if } n = n_0 \text{ and } m < M. \end{cases} \\
\lambda_c(n,m) &= \Lambda - \lambda_f(n,m).
\end{aligned}$$

The states of the system can be classified into three categories based on the arriving customer's choice behavior. From the state transition diagram in Figure 2, the states in region I are ”all join the free system” states; the states in region II are ”join either the free or the toll system” states; and the states in region III are ”all join the toll system”.

Based on the classification of the states, we specify the infinitesimal generator matrix **Q** for the QBD process as follows:

$$\mathbf{Q}=\begin{bmatrix}
\mathbf{C}_{00} & \mathbf{A}_{01} \\
\mathbf{D}_{10} & \mathbf{C}_{11} & \mathbf{A}_{12} \\
 & \mathbf{D}_{21} & \mathbf{C}_{22} & \mathbf{A}_{23} \\
 & & \ddots & \ddots & \ddots \\
 & & & \mathbf{D}_{M,M-1} & \mathbf{C}_{M,M} & \mathbf{A}_{M,M+1} \\
 & & & & \ddots & \ddots & \ddots \\
 & & & & & \mathbf{D}_{n_0-1,n_0-2} & \mathbf{C}_{n_0-1,n_0-1} & \mathbf{A}_{n_0-1,n_0} \\
 & & & & & & \mathbf{D} & \mathbf{C} & \mathbf{A} \\
 & & & & & & & \mathbf{D} & \mathbf{C} & \mathbf{A} \\
 & & & & & & & & \ddots & \ddots & \ddots
\end{bmatrix}, \tag{3}$$

where all elements are $(M+1)\times(M+1)$ matrices. For $0 \le n \le M$,

$$\begin{aligned}
\mathbf{C}_{00} &= \begin{bmatrix}
-\Lambda \\
\mu_c & -(\Lambda+\mu_c) \\
 & \ddots & \ddots \\
 & & \mu_c & -(\Lambda+\mu_c)
\end{bmatrix}, \\
\mathbf{A}_{01} &= \begin{bmatrix}
\Lambda \\
 & \ddots \\
 & & \Lambda
\end{bmatrix} = \Lambda I
\end{aligned}$$

$$\begin{aligned}
\mathbf{D}_{10} &= \begin{bmatrix}
\mu_f \\
 & \ddots \\
 & & \mu_f
\end{bmatrix} = \mu_f I, \\
\mathbf{C}_{11} &= \begin{bmatrix}
-(\mu_f+\Lambda) & \lambda_c(1,0) \\
\mu_c & -(\Lambda+\mu_c+\mu_f) \\
 & \ddots & \ddots \\
 & & \mu_c & -(\Lambda+\mu_c+\mu_f)
\end{bmatrix}, \\
\mathbf{A}_{12} &= \begin{bmatrix}
\lambda_f(1,0) \\
 & \Lambda \\
 & & \ddots \\
 & & & \Lambda
\end{bmatrix}
\end{aligned}$$

$$\mathbf{D}_{21} = \begin{bmatrix}
\mu_f \\
 & \ddots \\
 & & \mu_f
\end{bmatrix} = \mu_f I,$$

$$
\mathbf{C}_{22} = \begin{bmatrix}
-(\mu_f+\Lambda) & \lambda_c(2,0) & & & \\
\mu_c & -(\Lambda+\mu_c+\mu_f) & \lambda_c(2,1) & & \\
 & \mu_c & -(\Lambda+\mu_c+\mu_f) & & \\
 & & \ddots & \ddots & \\
 & & & \mu_c & -(\Lambda+\mu_c+\mu_f)
\end{bmatrix},
$$

$$
\mathbf{A}_{23} = \begin{bmatrix}
\lambda_f(2,0) & & & & \\
 & \lambda_f(2,1) & & & \\
 & & \Lambda & & \\
 & & & \ddots & \\
 & & & & \Lambda
\end{bmatrix}
$$

$$
\mathbf{D}_{M,M-1} = \begin{bmatrix}
\mu_f & & \\
 & \ddots & \\
 & & \mu_f
\end{bmatrix} = \mu_f I,
$$

$$
\mathbf{C}_{M,M} = \begin{bmatrix}
-(\mu_f+\Lambda) & \lambda_c(M,0) & & & \\
\mu_c & -(\Lambda+\mu_c+\mu_f) & \lambda_c(M,1) & & \\
 & \ddots & \ddots & \ddots & \\
 & & \mu_c & -(\Lambda+\mu_c+\mu_f) & \lambda_c(M,M-1) \\
 & & & \mu_c & -(\Lambda+\mu_c+\mu_f)
\end{bmatrix},
$$

$$
\mathbf{A}_{M,M+1} = \begin{bmatrix}
\lambda_f(M,0) & & & \\
 & \ddots & & \\
 & & \lambda_f(M,M-1) & \\
 & & & \Lambda
\end{bmatrix}.
$$

For $M+1 \leq n \leq n_0 - 1$,

$$
\mathbf{D}_{n,n-1} = \begin{bmatrix}
\mu_f & & \\
 & \ddots & \\
 & & \mu_f
\end{bmatrix} = \mu_f I,
$$

$$
\mathbf{C}_{n,n} = \begin{bmatrix}
-(\mu_f+\Lambda) & \lambda_c(n,0) & & & \\
\mu_c & -(\Lambda+\mu_c+\mu_f) & \lambda_c(n,1) & & \\
 & \ddots & \ddots & \ddots & \\
 & & \mu_c & -(\Lambda+\mu_c+\mu_f) & \lambda_c(n,M-1) \\
 & & & \mu_c & -(\Lambda+\mu_c+\mu_f)
\end{bmatrix},
$$

$$
\mathbf{A}_{n,n+1} = \begin{bmatrix}
\lambda_f(n,0) & & & & \\
 & \lambda_f(n,1) & & & \\
 & & \ddots & & \\
 & & & \lambda_f(n,M-1) & \\
 & & & & \Lambda
\end{bmatrix}.
$$

For $n \geq n_0$,

$$\begin{aligned}
\mathbf{D} &= \begin{bmatrix} \mu_f & & \\ & \ddots & \\ & & \mu_f \end{bmatrix} = \mu_f I = \mathbf{D}_{i,i-1}, \text{ for } i \geq 1. \\
\mathbf{C} &= \begin{bmatrix} -(\mu_f+\Lambda) & \Lambda & & & \\ \mu_c & -(\Lambda+\mu_c+\mu_f) & \Lambda & & \\ & \ddots & \ddots & \ddots & \\ & & & \mu_c & -(\Lambda+\mu_c+\mu_f) \end{bmatrix}, \\
\mathbf{A} &= \begin{bmatrix} & \\ & \Lambda \end{bmatrix}.
\end{aligned}$$

Under the stability condition (12) in Proposition 2.1, the stationary probability vector is defined as

$$\boldsymbol{\pi}_n = [\pi_{n0}, \pi_{n1}, ..., \pi_{nM}],$$

where $\pi_{nm} = \lim_{t\to\infty} P\{X_f(t) = n, X_c(t) = m\}$. We know that from Neuts [17] when $n \geq n_0$, the matrix geometric solution for such a QBD process is given by

$$\boldsymbol{\pi}_{n+1} = \boldsymbol{\pi}_n \mathbf{R}, \tag{4}$$

where **R** is the rate matrix. For $0 \leq n \leq n_0$, the probability vector $\boldsymbol{\pi}_n$ can be obtained by solving a set of equations. Let $\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_{n_0-1}, \boldsymbol{\pi}_{n_0}, \boldsymbol{\pi}_{n_0+1}, \ldots)$. From $\boldsymbol{\pi}\mathbf{Q} = \mathbf{0}$ and (4), the state vectors $\boldsymbol{\pi}_0$, $\boldsymbol{\pi}_1$, . . ., and $\boldsymbol{\pi}_{n_0}$ can be derived from the boundary conditions

$$\boldsymbol{\pi}_0 \mathbf{C}_{00} + \boldsymbol{\pi}_1 \mathbf{D} = \mathbf{0}, \tag{5}$$

$$\begin{aligned}
&\boldsymbol{\pi}_{n-1}\mathbf{A}_{n-1,n} + \boldsymbol{\pi}_n \mathbf{C}_{n,n} + \boldsymbol{\pi}_{n+1}\mathbf{D} = \mathbf{0}, \\
&\qquad 1 \leq n \leq n_0 - 2, \qquad (6) \\
&\boldsymbol{\pi}_{n_0-2}\mathbf{A}_{n_0-2,n_0-1} + \boldsymbol{\pi}_{n_0-1}\mathbf{C}_{n_0-1,n_0-1} + \boldsymbol{\pi}_{n_0}\mathbf{D} = \mathbf{0}, \\
&\boldsymbol{\pi}_{n_0-1}\mathbf{A}_{n_0-1,n_0} + \boldsymbol{\pi}_{n_0}(\mathbf{C} + \mathbf{R}\mathbf{D}) = \mathbf{0}, \qquad (7)
\end{aligned}$$

and the normalization condition

$$\boldsymbol{\pi}_0 \mathbf{1} + \boldsymbol{\pi}_1 \mathbf{1} + \cdots + \boldsymbol{\pi}_{n_0}(\mathbf{I} - \mathbf{R})^{-1}\mathbf{1} = 1. \tag{8}$$

To reduce the number of boundary states to one, we could define the following matrices:

$$\mathbf{C}_0 = \begin{bmatrix} \mathbf{C}_{00} & \mathbf{A}_{01} & & \\ \mathbf{D} & \mathbf{C}_{11} & \mathbf{A}_{12} & \\ & \ddots & \ddots & \\ & & \mathbf{D} & \mathbf{C}_{n_0-1,n_0-1} \end{bmatrix}_{(n_0(M+1))\times(n_0(M+1))}, \tag{9}$$

$$\mathbf{A}_0 = \left[\begin{array}{c} \\ \mathbf{A}_{n_0-1,n_0} \end{array} \right]_{(n_0(M+1))\times(M+1)}, \quad \mathbf{D}_1 = \left[\quad \mathbf{D} \ \right]_{(M+1)\times(n_0(M+1))}. \qquad (10)$$

Thus the **Q** matrix in (3) can be re-written as:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{C}_0 & \mathbf{A}_0 & & & \\ \mathbf{D}_1 & \mathbf{C} & \mathbf{A} & & \\ & \mathbf{D} & \mathbf{C} & \mathbf{A} & \\ & & \ddots & \ddots & \ddots \end{bmatrix}. \qquad (11)$$

By doing this, we have larger boundary level matrices $\mathbf{C}_0, \mathbf{A}_0$,and $\mathbf{D}_1$ which can lead to an alternative computational algorithm for numerical analysis. Using (11), we can also prove the following stability condition.

Proposition 2.1. *With real-time queue length information, the two-tier service system reaches the steady state if*

$$\mu_f > \frac{\left(1-\frac{\Lambda}{\mu_c}\right)\left(\frac{\Lambda}{\mu_c}\right)^M \Lambda}{1-\left(\frac{\Lambda}{\mu_c}\right)^{M+1}}. \qquad (12)$$

Equivalently, it satisfies

$$\frac{\Lambda}{\mu_f} < (\frac{\mu_c}{\Lambda})^M + (\frac{\mu_c}{\Lambda})^{M-1} + \cdots + 1.$$

It implies that for a stable system that includes a free queue and a cost queue, the ratio of the total arrival rates versus the service rate at the free queue must be less than the geometric series of the service rate at the cost queue over the total arrival rate. If the service rates and the total arrival rate are given, a proper buffer size at the cost queues may be estimated by (12). Nevertherless, note that (12) is reduced to more intuitive stability conditions when $M \longrightarrow \infty$. It is stated in the following corollary.

Corollary 2.2. *If $\Lambda/\mu_c \leq 1$, as $M \longrightarrow \infty$, (12) becomes $\mu_f > 0$ or there is no requirement for a positive μ_f. If $\Lambda/\mu_c > 1$, as $M \longrightarrow \infty$, (12) becomes $\mu_f + \mu_c > \Lambda$.*

Like any regular QBD process, the rate matrix **R** should satisfy

$$\mathbf{R}^2\mathbf{D} + \mathbf{R}\mathbf{C} + \mathbf{A} = 0 \qquad (13)$$

and can be solved by using one of many known algorithms (see Neuts [17], Bright and Taylor [1], and Latouche and Ramaswami [16]. The boundary state vector $(\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \cdots, \boldsymbol{\pi}_{n_0-1})$ together with $\boldsymbol{\pi}_{n_0}$ is the unique solution of the equation system of (5)–(8). After the stationary distribution is computed, we can obtain the major performance measures of the two-tier service system. Letting $\pi_{\cdot j} = \sum_{n=0}^{\infty} \pi_{nj}$ be the marginal probability of the toll system queue length and $\pi_{n\cdot} = \sum_{j=0}^{M} \pi_{nj}$ be the marginal probability of the free system queue length, we can compute the expected toll and free queue lengths $E(L^c) = \sum_{j=0}^{M} j\boldsymbol{\pi}_{\cdot j}$ and $E(L^f) = \sum_{n=0}^{\infty} n\boldsymbol{\pi}_{n\cdot}$, respectively. Due to the structure of our model, we can either treat the QBD with a large number of phases (i.e. $n_0(M+1)$ phases) in one boundary state and the generator of (11) or the QBD with a large number of boundary states (i.e n_0) with $M+1$ phases and the generator of (3). However, it follows from (1) that n_0 increases with M. To model a realistic system, M can be quite large and this is particularly true when we analyze a large scale system with heavy traffic intensity. Such a large M results in a large number of boundary states and a large number of phases of the QBD process which greatly increase the computational complexity and may cause the ill-conditioned matrices of the traditional iterative algorithm for the rate matrix. To overcome this challenge, using the special structure of the infinitesimal generator (3), we propose a more efficient and innovative algorithm for computing the stationary distribution. Compared with the traditional matrix geometric solution algorithm, our so-called K-matrix based algorithm is faster, more numerically stable and accurate, and can be applied to solving for the performance measures of large scale two-tier service systems. Since the two-tier service system has wide applications in public service sector, our new algorithm provides practitioners a powerful tool for evaluating the performance of the service systems. The details of the algorithm development and theoretical justification can be found in Section 3.

2.1. Background of QBD with special property

Dayar and Quessette [3] consider a special class of homogeneous continuous-time QBD Markov Chain which posses level-geometric (LG) stationary distribution. They refer to an LG distribution for which L is the smallest possible nonnegative integer that satisfies

$$\boldsymbol{\pi}_{n+1} = x\boldsymbol{\pi}_n \qquad \text{for } n \geq L,$$

where $x \in (0,1)$. In an LG distribution, the level is independent of the phase for level numbers greater than or equal to L. It will be discussed that in the next section $L = n_0 + 1$ in our model. As indicated in the paper of Dayar and Quessette [3], it requires a set of a nonlinear system of equations to solve for x. The next propositions are drawn from the fact that $\mathbf{Q}$ is positive recurrent when $\mathbf{Q}$ and $\overline{\mathbf{D}} = \mathbf{D} + \mathbf{C} + \mathbf{A}$ are both irreducible in which x is described.

Proposition 2.3. *If* $\mathbf{Q}$ *and* $\overline{\mathbf{D}}$ *are irreducible, then* $\mathbf{Q}$ *is positive recurrent if and only if* $\mathbf{p}(\mathbf{A} - \mathbf{D})\mathbf{1} < 0$*, where* $\mathbf{p}$ *satisfies* $\mathbf{p}\overline{\mathbf{D}} = \mathbf{0}$ *and* $\mathbf{p}\mathbf{1} = 1$*. The stationary distribution of* $\mathbf{Q}$ *in which* $\mathbf{A}_0 = \mathbf{A}$ *and* $\mathbf{D}_1 = \mathbf{D}$ *is LG with parameter* $L = 0$ *if and only if there exists a positive vector* $\mathbf{a}$ *with* $\mathbf{a}\mathbf{1} = 1$ *and a positive scalar* $0 < x < 1$ *which is the spectral radius of* $\mathbf{R}$ *such that* $\mathbf{a}(x^2\mathbf{D} + x\mathbf{C} + \mathbf{A}) = \mathbf{0}$ *and* $\mathbf{a}(\mathbf{C}_0 + x\mathbf{D}) = \mathbf{0}$*.*

Proposition 2.4. *When* $\mathbf{A}$ *is of rank-1 then* $\mathbf{R}$ *is also rank-1 and* $\mathbf{R} = \mathbf{c}\boldsymbol{\xi}^T$ *where* $\mathbf{A} = \mathbf{c}\mathbf{b}^T$*,* $\mathbf{b} = \mathbf{e}_j$*,* $\mathbf{c}^T = \mathbf{e}_j\Lambda$*,* $\mathbf{e}_j$ *is a unit column vector and* $j = M+1$*. Then* x *satisfies the following equations.* $\boldsymbol{\xi}^T = -\mathbf{b}^T(\mathbf{C} + x\mathbf{D})^{-1}$*,* $x = \boldsymbol{\xi}^T\mathbf{c}$ *,* $x \in (0,1)$ *and* $\mathbf{R}^2 = x\mathbf{R}$*.*

Since $\mathbf{R} = \mathbf{c}\boldsymbol{\xi}^T$, we have $\mathbf{R}^2 = \mathbf{c}\boldsymbol{\xi}^T\mathbf{c}\boldsymbol{\xi}^T = x\mathbf{c}\boldsymbol{\xi}^T$. The details can be found in Dayar and Quessette [3]. Apparently, it involves a system of nonlinear equations of degree more than

$M+1$ to solve for x. In the next section, we propose a new approach to solve the problem with only a simple matrix. After a proper matrix is constructed, it is a routine to attain x^* as an eigenvalue of such a matrix. Most importantly, such an x^* exists uniquely between 0 and 1.

2.2. Constructing the K matrix

Consider the state balance equations where the number of customers in the free queue $n \geq n_0$, i.e.,

$$\boldsymbol{\pi}_n \mathbf{A} + \boldsymbol{\pi}_{n+1}\mathbf{C} + \boldsymbol{\pi}_{n+2}\mathbf{D} = \mathbf{0}, \quad n \geq n_0, \tag{14}$$

Rewriting it, we have

$$\boldsymbol{\pi}_n \mathbf{A1} + \boldsymbol{\pi}_{n+1}(-\mathbf{A}-\mathbf{D})\mathbf{1} + \boldsymbol{\pi}_{n+2}\mathbf{D1} = 0, \quad n \geq n_0 \tag{15}$$

$$\text{since} \qquad (\mathbf{A}+\mathbf{C}+\mathbf{D})\mathbf{1} = \mathbf{0}. \tag{16}$$

Define $\mathbf{d}_n \triangleq \boldsymbol{\pi}_n - \boldsymbol{\pi}_{n+1}$ for $n \geq n_0$. Equation (14) is also written as

$$\mathbf{d}_n\mathbf{A1} = \mathbf{d}_{n+1}\mathbf{D1}.$$

Although there are infinitely many possibilities to connect these two sets of equations, at least it is a hint here to start with construction of a matrix that relates to **R** which is rank-1. In our model, because **A** is a rank-1 matrix and **D** is a full rank matrix, in order to construct a proper matrix that balances $\mathbf{d}_n$ and $\mathbf{d}_{n+1}$, we begin with a simple matrix algebra. Multiplying **1** from the right on (14), we have

$$\boldsymbol{\pi}_n \mathbf{A1} + \boldsymbol{\pi}_{n+1}\mathbf{C1} + \boldsymbol{\pi}_{n+2}\mathbf{D1} = 0, \qquad n \geq n_0$$

and equivalently

$$\boldsymbol{\pi}_n \begin{bmatrix} 0 \\ \vdots \\ \Lambda \end{bmatrix} + \boldsymbol{\pi}_{n+1} \begin{bmatrix} -\mu_f \\ \vdots \\ -\mu_f - \Lambda \end{bmatrix} + \boldsymbol{\pi}_{n+2} \begin{bmatrix} \mu_f \\ \vdots \\ \mu_f \end{bmatrix} = 0, \quad n \geq n_0 \tag{17}$$

We rewrite those equations as

$$\boldsymbol{\pi}_n \begin{bmatrix} 0 & \dots & 0 & 0 \\ & \ddots & 0 & 0 \\ & & 0 & \Lambda \end{bmatrix} + \boldsymbol{\pi}_{n+1} \begin{bmatrix} 0 & \dots & -\mu_f \\ & \ddots & \vdots \\ & & -\mu_f - \Lambda \end{bmatrix} + \boldsymbol{\pi}_{n+2} \begin{bmatrix} 0 & \dots & \mu_f \\ & \ddots & \vdots \\ & & \mu_f \end{bmatrix} = \mathbf{0}, \quad n \geq n_0.$$

Define

$$\boldsymbol{\mathcal{P}} \triangleq \begin{bmatrix} 0 & \cdots & 0 & \mu_f \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & \mu_f \end{bmatrix}.$$

We have

$$\begin{aligned} \boldsymbol{\pi}_n\mathbf{A} + \boldsymbol{\pi}_{n+1}[-\boldsymbol{\mathcal{P}} - \mathbf{A}] + \boldsymbol{\pi}_{n+2}\boldsymbol{\mathcal{P}} = &\quad \mathbf{0} \qquad (18)\\ \text{and}\quad [\boldsymbol{\pi}_n - \boldsymbol{\pi}_{n+1}]\mathbf{A} - [\boldsymbol{\pi}_{n+1} - \boldsymbol{\pi}_{n+2}]\boldsymbol{\mathcal{P}} = &\quad \mathbf{0}\\ \text{implying}\quad \mathbf{d}_n\mathbf{A} = \mathbf{d}_{n+1}\boldsymbol{\mathcal{P}}, &\qquad n \geq n_0. \qquad (19) \end{aligned}$$

Note if equations (14) and (18) hold for $\boldsymbol{\pi}_n$, and they hold for $\mathbf{d}_n$, too. Replace $\boldsymbol{\pi}_n$ by $\mathbf{d}_n$ in equations (14) and (18). Subtracting equation (14) from (18), we easily obtain

$$\mathbf{d}_n[\mathbf{C} + \boldsymbol{\mathcal{P}} + \mathbf{A}] + \mathbf{d}_{n+1}[\mathbf{D} - \boldsymbol{\mathcal{P}}] = \mathbf{0}, \quad n \geq n_0 + 1.$$

By equation (19), it produces $\mathbf{d}_n[\mathbf{C} + \boldsymbol{\mathcal{P}}] + \mathbf{d}_{n+1}\mathbf{D} = \mathbf{0}$. Let

$$\mathbf{K} \triangleq -[\mathbf{C} + \boldsymbol{\mathcal{P}}]\mathbf{D}^{-1}, \qquad (20)$$

which is an $(M+1) \times (M+1)$ matrix. Because **D** is invertible, it becomes easy to check the eigenvalue of **K** since the Markovian system is ergodic with the stability condition (2.1). we have
$\mathbf{d}_n(-[\mathbf{C} + \boldsymbol{\mathcal{P}}]\mathbf{D}^{-1}) = \mathbf{d}_{n+1}$ and

$$\mathbf{d}_{n+1} = \mathbf{d}_n\mathbf{K} \quad \text{for} \quad n \geq n_0 + 1. \qquad (21)$$

Lemma 2.5. *The eigenvalues of matrix* **K** *are positive.*

The proof is given in the appendix.

Proposition 2.6. **K** *is constructed under the stability condition (2.1) and there exists a unique eigenvalue of* **K** *between 0 and 1.*

The proof is given in Section 3.

Corollary 2.7. *There exists* $\sigma \in (0, 1)$ *and a corresponding eigenvector* $\boldsymbol{\theta}$ *where* $\boldsymbol{\theta} > \mathbf{0}$ *such that*

$$\mathbf{K}\boldsymbol{\theta} = \sigma\boldsymbol{\theta}.$$

The proof can be shown by applying Proposition 2.6 and Perron's theorem with eigenvalues in Horn and Johnson [12].

From (21), we have

$$\mathbf{d}_{n+1}\boldsymbol{\theta} = \mathbf{d}_n\mathbf{K}\boldsymbol{\theta} = \sigma\mathbf{d}_n\boldsymbol{\theta}, \quad \text{for } n \geq n_0 + 1.$$

By induction on n, it implies that $\mathbf{d}_{n+t}\boldsymbol{\theta} = \sigma^t\mathbf{d}_n\boldsymbol{\theta}$, $t = 1, 2, \cdots$.

By the definition of $\mathbf{d}_n$ we have

$$\begin{aligned} \boldsymbol{\pi}_{n_0+1} = &\quad \boldsymbol{\pi}_{n_0+2} + \mathbf{d}_{n_0+1}\\ \boldsymbol{\pi}_{n_0+2} = &\quad \boldsymbol{\pi}_{n_0+3} + \mathbf{d}_{n_0+2} \end{aligned}$$

$$\begin{aligned} \boldsymbol{\pi}_{n_0+3} = &\ \boldsymbol{\pi}_{n_0+4} + \mathbf{d}_{n_0+3} \\ \vdots \quad &\ \vdots \end{aligned}$$

Summing up the above equations, it results in $\boldsymbol{\pi}_{n_0+1} = (\mathbf{d}_{n_0+1} + \mathbf{d}_{n_0+2} + \cdots)$. By the stability assumption, $\boldsymbol{\pi}_n$ approaches to 0 as n approaches to infinity, i.e., $\boldsymbol{\pi}_{n_0+1} = (\mathbf{d}_{n_0+1} + \mathbf{d}_{n_0+2} + \cdots) < \infty$. With the relation of equation (21) and an eigenvector $\boldsymbol{\theta}$, we have

$$\begin{aligned} \boldsymbol{\pi}_{n_0+1}\boldsymbol{\theta} &= (\mathbf{d}_{n_0+1} + \mathbf{d}_{n_0+2} + \cdots)\boldsymbol{\theta} \\ &= \mathbf{d}_{n_0+1}(1 + \sigma + \sigma^2 + \ldots)\boldsymbol{\theta} \\ \boldsymbol{\pi}_{n_0+2}\boldsymbol{\theta} &= (\mathbf{d}_{n_0+2} + \mathbf{d}_{n_0+3} + \cdots)\boldsymbol{\theta} \\ &= \sigma\mathbf{d}_{n_0+1}(1 + \sigma + \sigma^2 + \ldots)\boldsymbol{\theta} \\ &= \sigma\boldsymbol{\pi}_{n_0+1}\boldsymbol{\theta} \end{aligned}$$

Since $\boldsymbol{\pi}_n$ is finite and nonnegative for all n, and $\boldsymbol{\theta}$ is a positive vector that is independent of n, it yields the following lemma.

Lemma 2.8. *Under the stability assumption and $\sigma \in (0,1)$, we have*

$$\boldsymbol{\pi}_{n_0+t} = \sigma_k^{t-1}\boldsymbol{\pi}_{n_0+1}, \;\; t \geq 1.$$

The proof is straightforward. Because of $(\boldsymbol{\pi}_{n_0+2} - \sigma\boldsymbol{\pi}_{n_0+1})\boldsymbol{\theta} = 0$ and $\boldsymbol{\pi}_n$ and $\mathbf{d}_n$, $n > n_0$ belonging to a subspace of solving (14), it implies $\boldsymbol{\pi}_{n_0+2} - \sigma\boldsymbol{\pi}_{n_0+1} = \mathbf{0}$ since $\boldsymbol{\theta} > 0$. By induction on t, we have it proved for all $t \geq 1$.

Theorem 2.9. *Suppose σ is an eigenvalue of* $\mathbf{K}$ *and $\sigma \in (0,1)$, then* $\mathbf{R}^2 = \sigma\mathbf{R}$.

The proof is straightforward from Lemma 2.8 and Proposition 2.4.

$\mathbf{R}$ can be derived from (13), that is

$$\mathbf{A} + \mathbf{R}\{\mathbf{C} + \sigma\mathbf{D}\} = \mathbf{0},$$

$$\mathbf{R} = -\mathbf{A}(\mathbf{C} + \sigma\mathbf{D})^{-1}. \tag{22}$$

Because $(\mathbf{C} + \sigma\mathbf{D})$ is a tridiagonal matrix, the inverse of it may be determined by El-Mikkawy and Karawia [5]. Moreover, $\mathbf{A}$ has only a positive element where Λ is at the southeastern corner, and $\mathbf{R}$ has the following form

$$\mathbf{R} = -\Lambda \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \\ r_1 & r_2 & \cdots & r_{M+1} \end{bmatrix}.$$

An illustrative example

Consider a two-tier system where there is a paid service but no waiting space for the toll queue, that is $M = 1$. To enter the system, any new arrival is forced to trade off the price for a waiting position at the free queue with the paid service when the free queue length is sufficiently long, i.e., $n > n_0$. From (11), set

$$\mathbf{D} = \begin{bmatrix} \mu_f & 0 \\ 0 & \mu_f \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} -(\mu_f + \Lambda) & \Lambda \\ \mu_c & -(\Lambda + \mu_c + \mu_f) \end{bmatrix},$$

$$\mathbf{A} = \begin{bmatrix} 0 & 0 \\ 0 & \Lambda \end{bmatrix}.$$

By (20), set

$$\mathbf{K} = \begin{bmatrix} \frac{\Lambda+\mu_f}{\mu_f} & \frac{-(\Lambda+\mu_f)}{\mu_f} \\ \frac{-\mu_c}{\mu_f} & \frac{\Lambda+\mu_c}{\mu_f} \end{bmatrix}.$$

Lemma 2.10. *The eigenvalue of* **K** *is*

$$\sigma = \frac{(\frac{2\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f} + 1) - ((\frac{\mu_c}{\mu_f})^2 + \frac{4\Lambda\mu_c}{\mu_f^2} + \frac{2\mu_c}{\mu_f} + 1)^{\frac{1}{2}}}{2}$$

which is less than 1 and greater than 0.

The proof is given in the appendix.

After σ is given, we can easily find **R** and determine the stationary probability $\boldsymbol{\pi}$ by attaining $\boldsymbol{\pi}_0$ from the boundary equations that will be introduced in Section 3.3. Here, we only illustrate the construction of **K** and **R**, i.e.,

$$\mathbf{R} = -\Lambda \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \sigma\mu_f - \Lambda - \mu_f & \Lambda \\ \mu_c & \sigma\mu_f - \Lambda - \mu_f - \mu_c \end{bmatrix}^{-1}$$

$$= \Lambda \begin{bmatrix} 0 & 0 \\ \frac{\mu_c}{(\sigma\mu_f - \Lambda - \mu_f)(\sigma\mu_f - \Lambda - \mu_f - \mu_c) - \Lambda\mu_c} & \frac{-(\sigma\mu_f - \Lambda - \mu_f)}{(\sigma\mu_f - \Lambda - \mu_f)(\sigma\mu_f - \Lambda - \mu_f - \mu_c) - \Lambda\mu_c} \end{bmatrix}.$$

Besides the QBD MC considered in a two-tier service model, there are two examples in the appendix that show construction of **K** and its eigenvalue.

3. An Eigenvalue Approach

In this section, we develop the K-matrix algorithm to solve the stationary probability $\boldsymbol{\pi}$. Before doing that, we first show derivation of the eigenvalue of **K**. Then we prove the existence and uniqueness of this eigenvalue.

3.1. The structure property of K

By (20), we continue to observe its element structure by visualization from a small one. Denote by $\mathbf{K}_m$ a square matrix with m rows and m columns where $1 \leq m \leq (M+1)$. It is easy to check that $\mathbf{K}_3$ and $\mathbf{K}_4$ are expressed respectively, by

$$\mathbf{K}_3 = \begin{bmatrix} \frac{\Lambda+\mu_f}{\mu_f} & \frac{-\Lambda}{\mu_f} & -1 \\ \frac{-\mu_c}{\mu_f} & \frac{\Lambda+\mu_f+\mu_c}{\mu_f} & \frac{-\Lambda-\mu_f}{\mu_f} \\ 0 & \frac{-\mu_c}{\mu_f} & \frac{\Lambda+\mu_c}{\mu_f} \end{bmatrix}$$

and

$$\mathbf{K}_4 = \begin{bmatrix} \frac{\Lambda+\mu_f}{\mu_f} & \frac{-\Lambda}{\mu_f} & 0 & -1 \\ \frac{-\mu_c}{\mu_f} & \frac{\Lambda+\mu_f+\mu_c}{\mu_f} & \frac{-\Lambda}{\mu_f} & -1 \\ 0 & \frac{-\mu_c}{\mu_f} & \frac{\Lambda+\mu_c+\mu_f}{\mu_f} & \frac{-\Lambda-\mu_f}{\mu_f} \\ 0 & 0 & \frac{-\mu_c}{\mu_f} & \frac{\Lambda+\mu_c}{\mu_f} \end{bmatrix}.$$

In order to find an eigenvalue of $\mathbf{K}$ which is located between 0 and 1, we need to consider the determinant of $\mathbf{K}$. Let $\ell(x)$ be the characteristic polynomial defined as $\ell(x) \triangleq \langle \mathbf{K} - x\mathbf{I} \rangle$ where $\langle \cdot \rangle$ denotes determinant of a matrix and $\mathbf{I}$ is an identity matrix with a proper size. We need to show $\ell(0) \times \ell(1) < 0$, i.e., $\langle \mathbf{K} \rangle \times \langle \mathbf{K} - \mathbf{I} \rangle < 0$ as well as there is one $x \in (0,1)$ such that $\ell(x) = 0$.

To simplify its notation, let a, b and c be defined as

$$a \triangleq \frac{\Lambda + \mu_f + \mu_c}{\mu_f}, \; b \triangleq \frac{-\Lambda}{\mu_f}, \; c \triangleq \frac{-\mu_c}{\mu_f}.$$

Let

$$\Phi_m \triangleq \begin{bmatrix} a & b & 0 & & & -1 \\ c & a & b & 0 & & -1 \\ 0 & c & a & b & 0 & \vdots \\ & & \ddots & \ddots & \ddots & -1 \\ & & & c & a & b-1 \\ & & & & c & a-1 \end{bmatrix}_{m \times m}$$

Thus, $\mathbf{K}_m$ is written in terms of a, b, and c, for example

$$\mathbf{K}_m = \begin{bmatrix} a+c & b & 0 & & & -1 \\ c & a & b & 0 & & -1 \\ 0 & c & a & b & 0 & \vdots \\ & & \ddots & \ddots & \ddots & -1 \\ & & & c & a & b-1 \\ & & & & c & a-1 \end{bmatrix}_{m \times m}.$$

It is easy to check $a+b+c=1, a>0, b<0, c<0$ and verify that

$$\begin{aligned}\langle\Phi_2\rangle &= a^2-a-cb+c = a(a-1)-c(b-1) = a(-b-c)-c(-a-c)\\ &= c^2-ab = \tfrac{\mu_c^2+(\Lambda+\mu_c+\mu_f)\Lambda}{\mu_f^2} > 0.\end{aligned}$$

Let $\langle\Phi_1\rangle = (a-x-1)$. In general, consider a characteristic polynomial of $\mathbf{\Phi}_m$. We observe the following property of $\mathbf{\Phi}_m$. For $m\geq 3$, we have the following lemmas.

Lemma 3.1. $\langle\Phi_m - x\mathbf{I}\rangle = (a-x)\langle\Phi_{m-1}-x\mathbf{I}\rangle - bc\langle\Phi_{m-2}-x\mathbf{I}\rangle + (-1)^m(-c)^{m-1}, 3\leq m\leq M+1.$

Furthermore, it is easy to prove it by induction through characteristic polynomials of K_m and Φ_m.

Lemma 3.2. $\langle K_m - x\mathbf{I}\rangle = \langle\Phi_m - x\mathbf{I}\rangle + c\langle\Phi_{m-1}-x\mathbf{I}\rangle, m\geq 3.$

It is natural to prove it by induction on $m=4,5,\cdots,M+1$. Then we have following lemmas.

Lemma 3.3. *We have* $\langle K_m - x\mathbf{I}\rangle = (1-x)\langle\Phi_{m-1}-x\mathbf{I}\rangle - b\langle K_{m-1}-x\mathbf{I}\rangle + (-1)^m(\frac{\mu_c}{\mu_f})^{m-1}$, $m=4,5,\cdots,M+1.$

Proof. For a fixed $m>3$, we have

$$\begin{aligned}\langle K_m - x\mathbf{I}\rangle &= \langle\Phi_m - x\mathbf{I}\rangle + c\langle\Phi_{m-1}-x\mathbf{I}\rangle\\ &= (a-x)\langle\Phi_{m-1}-x\mathbf{I}\rangle - bc\langle\Phi_{m-2}-x\mathbf{I}\rangle + (-1)^m(\frac{\mu_c}{\mu_f})^{m-1} + c\langle\Phi_{m-1}-x\mathbf{I}\rangle\\ &= (1-b-x)\langle\Phi_{m-1}-x\mathbf{I}\rangle - bc\langle\Phi_{m-2}-x\mathbf{I}\rangle + (-1)^m(\frac{\mu_c}{\mu_f})^{m-1}\\ &= (1-x)\langle\Phi_{m-1}-x\mathbf{I}\rangle - b(\langle\Phi_{m-1}-x\mathbf{I}\rangle + c\langle\Phi_{m-2}-x\mathbf{I}\rangle) + (-1)^m(\frac{\mu_c}{\mu_f})^{m-1}\\ &= (1-x)\langle\Phi_{m-1}-x\mathbf{I}\rangle - b\langle K_{m-1}-x\mathbf{I}\rangle + (-1)^m(\frac{\mu_c}{\mu_f})^{m-1}.\end{aligned}$$

Now, we verify

$$\langle K_3\rangle = (a+c)\langle\Phi_2\rangle - c\langle b(a-1)+c\rangle = \frac{1}{\mu_f^3}(\Lambda^3+2\Lambda^2\mu_f+\Lambda\mu_f\mu_c+\Lambda\mu_f^2) > 0. \quad (23)$$

By Lemma 3.1, when $x=0$ we have $\langle\Phi_3\rangle = \langle K_3\rangle - c\langle\Phi_2\rangle > 0$. It is easy to check that

$$\langle\Phi_3\rangle > \frac{\mu_c}{\mu_f}\langle\Phi_2\rangle > (\frac{\mu_c}{\mu_f})^3 > 0. \quad (24)$$

Lemma 3.4. $\langle\Phi_m\rangle > \frac{\mu_c}{\mu_f}\langle\Phi_{m-1}\rangle > (\frac{\mu_c}{\mu_f})^m$, *and* $\langle K_m\rangle > 0,\ 3\leq m\leq M+1.$

Proof. By induction, from $m = 4$, we have

$$\begin{aligned}\langle K_4\rangle &= \langle\Phi_4\rangle + c\langle\Phi_3\rangle \\ &= \langle\Phi_3\rangle - b\langle K_3\rangle - (\frac{\mu_c}{\mu_f})^3\end{aligned}$$

By (23) and (24), we have $\langle K_4\rangle > 0$. By Lemma (3.1), we have $\langle\Phi_4\rangle > 0$ and

$$\langle\Phi_4\rangle > \frac{\mu_c}{\mu_f}\langle\Phi_3\rangle > (\frac{\mu_c}{\mu_f})^4. \tag{25}$$

Because of $\langle K_4\rangle > 0$ and (25), it gives $\langle K_5\rangle > 0$. By recursively using Lemmas 3.3 and 3.4, it completes the proof.

Thus, it concludes that $\ell(0) = \langle\mathbf{K}\rangle > 0$. Next we need to prove that $\ell(1) = \langle\mathbf{K}-\mathbf{I}\rangle < 0$.

Lemma 3.5. *Under the stability condition, we have* $\ell(1) = \langle\mathbf{K} - \mathbf{I}\rangle < 0$.

Proof. It can be derived that $\langle\mathbf{K} - \mathbf{I}\rangle =$

$$\frac{-\mu_f(\Lambda^M + \mu_c\Lambda^{M-1} + \mu_c^2\Lambda^{M-2} + \cdots + \mu_c^M) - \Lambda^{M+1}}{\mu_f^{M+1}}$$

$$= \mu_c^M \times \frac{-\mu_f((\frac{\Lambda}{\mu_c})^M + (\frac{\Lambda}{\mu_c})^{M-1} + \cdots + 1) - (\frac{\Lambda}{\mu_c})^M\Lambda}{\mu_f^{M+1}}$$

$$= \mu_c^M[(\frac{\Lambda}{\mu_c})^M + (\frac{\Lambda}{\mu_c})^{M-1} + (\frac{\Lambda}{\mu_c})^{M-2} + \cdots + 1]$$

$$\cdot\frac{-[\mu_f - \frac{(\frac{\Lambda}{\mu_c})^M\Lambda}{(\frac{\Lambda}{\mu_c})^M+(\frac{\Lambda}{\mu_c})^{M-1}+(\frac{\Lambda}{\mu_c})^{M-2}+\cdots+1)}]}{\mu_f^{M+1}}. \tag{26}$$

By the stability condition

$$\mu_f > \frac{(\frac{\Lambda}{\mu_c})^M\Lambda}{[(\frac{\Lambda}{\mu_c})^M + (\frac{\Lambda}{\mu_c})^{M-1} + (\frac{\Lambda}{\mu_c})^{M-2} + \cdots + 1]},$$

we know that the second term in (26) is negative but the first term is positive. Hence, their product makes $\langle\mathbf{K} - \mathbf{I}\rangle < 0$.

Consequently, we have $\ell(0)\ell(1) < 0$ and shown that at least there exists an eigenvalue of **K** in (0,1). In the next section, we are going to prove the uniqueness of such an eigenvalue.

3.2. Location of the eigenvalue

In sequel, we will only illustrate by construction of a Sturm [19] sequence the uniqueness of such an eigenvalue without giving a complete proof since the uniqueness is not necessary when the existence appears critically in our approach. But the illustration shows the usefulness of this approach. First, from Grassmann [6], we know there are distinct $M+1$ positive eigenvalues of **K** of the birth-death Markov chain model as μ_f and μ_c are strictly positive. Then we shall construct a Sturm sequence to show that a unique eigenvalue between 0 and 1 (Theorem 1 in Grassmann [6]). According to Lemma 3.3 we construct a series of polynomials of x, $\{G_m(x), m = 0, 1, 2, \cdots, M+1\}$, as follows

$$\begin{aligned} G_0(x) &= 1 \\ G_1(x) &= a - x - 1 \\ G_2(x) &= (a-x)(a-x-1) - bc + c \\ G_3(x) &= (a-x)G_2(x) - bc\, G_1(x) + (-1)c^2, \\ G_m(x) &= (a-x)G_{m-1}(x) - bc\, G_{m-2}(x) - (-c)^{m-1} \end{aligned}$$

for $m = 4, 5, \cdots, M$. The last term is

$$G_{M+1}(x) = (a - x + c)G_M(x) - bc\, G_{M-1}(x) - (-c)^M.$$

Apparently, $G_{M+1}(x) = \ell(x)$ which is the characteristic polynomial of **K**. Following the Sturm sequence, we count the number of sign changes of the sequence $\{G_m(x), m = 0, 1, 2, \cdots, M+1\}$. For a real number r, define

$$S(r) = \{G_0(r), G_1(r), \cdots, G_{M+1}(r)\}$$

and $s(x)$ the sign changes in $S(x)$. Clearly, the number of sign changes $s(x)$ is 0 if $G_m(x) > 0$ for all m and $s(x) = M+1$ if the sign changes every time. The value of $s(x)$ cannot change unless there exists an m such that $G_m(x)$ goes through zero meaning $G_{m-1}(x)G_m(x) < 0$.

One of the Sturm properties is that the number of sign changes $s(r)$ in $S(r)$ equals the number of eigenvalues of **K** less than r. Thus, if $G_m(x)$ forms a Sturm sequence, the number of roots of $\ell(x)$ in (0, 1) is $s(1) - s(0)$. We already know $s(0) = 0$, since it has been proved that $G_{M+1}(0) > 0$ and $G_m(0) > 0$ for all $m = 1, 2, 3, \cdots, M$. In the following section, we are going to prove $s(1) = 1$.

Since $G_{M+1}(x) = \ell(x)$, we shall first write $G_m(x)$ in terms of x. In order to find a general form of $G_m(x)$, we consider an inhomogeneous second order difference equation, for $m \geq 3$,

$$G_m(x) - (a-x)G_{m-1}(x) + bcG_{m-2}(x) = -(-c)^{m-1},$$

where $G_0(x) = 1$ and $G_1(x) = a - x - 1$. First, we solve the homogeneous second order difference equation, namely, one of the form given above where the right-hand side is zero. In a specific case, when $x = 1$ one solves the following equation

$$y^2 + (b+c)y + bc = 0.$$

Thus, we have

$$y = \frac{-(b+c) \pm |b-c|}{2} = \begin{cases} \quad -c & \text{if } b = c \\ -c, \text{ or } -b & \text{if } b \neq c. \end{cases}$$

The particular solution of $G_m(1)$ when $b \neq c$ is

$$G_m(1) = \alpha(-c)^m + \beta(-b)^m + \frac{m(-c)^m}{c-b},$$

where $\alpha = \frac{c^2-bc-b}{(c-b)^2}$ and $\beta = \frac{b(1-c+b)}{(c-b)^2}$, $c \neq b$; or, the alternative solution with $b = c$ gives

$$G_m(1) = (\alpha + \beta m)(-c)^m + \frac{m^2(-c)^m}{2c},$$

where $\alpha = 1$ and $\beta = \frac{2c+1}{2c}$.

Lemma 3.6. *There exists* $m_0 > 0$ *such that* $G_m(1) < 0$ *for all* $m > m_0$ *if* $G_{m_0}(1) < 0$.

Proof. First in case of $b = c$, we consider $G_m(1) = (-c)^m[1 + \beta m + \frac{m}{2c}^2]$. We have $G_m(1) = 0$ if and only if $[1 + \beta m + \frac{m}{2c}^2] = 0$. Moreover, $G_m(1) < 0$ if $m > -2c$ since $(-c)^m$ is always greater than 0. Thus we have $G_m(1) < 0$, for all $m > m_0$ when $G_{m_0}(1) < 0$, $m_0 > 0$.

Second, it is clear that for $b \neq c$

$$\begin{aligned} G_{m_0+1}(1) &= \alpha(-c)^{m_0+1} + \beta(-b)^{m_0+1} + \frac{m_0+1}{c-b}(-c)^{m_0+1} \\ &= [\alpha + \beta(\frac{b}{c})^{m_0}(\frac{b}{c}) + \frac{m_0+1}{c-b}](-c)^{m_0+1}. \end{aligned}$$

To prove $G_{m_0+1}(1) < 0$, one shall claim that

$$[\alpha + \beta(\frac{b}{c})^{m_0}(\frac{b}{c}) + \frac{m_0+1}{c-b}] < 0.$$

From $G_{m_0}(1) < 0$, by induction, suppose $k > m_0$ such that

$$\alpha(-c)^k + \beta(-b)^k + \frac{k}{c-b}(-c)^k < 0.$$

Next, we need to prove that

$$[\alpha(-c)^{k+1} + \beta(-b)^{k+1} + \frac{k+1}{c-b}(-c)^{k+1}] < 0.$$

Note that $\beta < 0$ because it can be derived that $\Lambda < \mu_c + \mu_f$ implying $(c-b) < 1$.

(Case i.) If $c - b < 0$, then we have $0 < \frac{b}{c} < 1$. It can be derived that

$$[\alpha + \beta(\frac{b}{c})^k(\frac{b}{c}) + \frac{k+1}{c-b}] < 0$$

since

$$[\alpha + \beta(\frac{b}{c})^k + \frac{k}{c-b}] < 0$$

and $(\frac{b}{c}) > 0$. Thus, we have

$$[\alpha + \beta(\frac{b}{c})^k(\frac{b}{c}) + \frac{k+1}{c-b}](-c)^{k+1} < 0.$$

(Case ii.) If $c - b > 0$, then it is clear that $\frac{b}{c} > 1$. Thus, there exists a sufficiently large $m_0 \gg 0$ such that

$$[\alpha + \beta(\frac{b}{c})^{m_0}(\frac{b}{c}) + \frac{m_0+1}{c-b}] < 0.$$

Proposition 3.7. *The number of sign changes $s(1)$ of eigenvalues of* $\mathbf{K}$ *is 1 and* $s(1) - s(0) = 1$.

Proof. Suppose $m_0 - 1$ is the smallest integer such that $G_{m_0-1}(1) > 0$, implying that $G_{m_0}(1) < 0$. By the construction of sequence of $G_m(x)$ and the last term

$$\begin{aligned} G_{M+1}(1) &= (a + c - 1)G_M(1) - bcG_{M-1}(1) - (-c)^M, \\ &= (-b)(G_M(1) + cG_{M-1}(1)) - (-c)^M. \end{aligned}$$

and by Lemma 16 that $G_m(1) < 0$ for all $m > m_0$, we may prove it by induction that $G_{M+1}(1)$ is negative as M is sufficiently large since c and b are negative. Clearly, the value of $s(1)$ can not change sign until m_0, i.e., $G_{m_0-1}(1)G_{m_0}(1) < 0$. Consequently, we know that $s(1) = 1$ and $s(1) - s(0) = 1$, By the Sturm theorem, there is only one eigenvalue in $(0, 1)$.

Although the proposition has been proved completely, it is sufficient to provide the stationary probability solution through the existence of a real eigenvalue between 0 and 1. This is because the system is stable, and one can always use the normalization condition to justify the state balance equations with proper parameters at the boundary equations. Consider σ is a function of the buffer size of the cost queue, M and denote it by $\sigma(M)$. We have the following proposition to further reduce the computational efforts in calculating the eigenvalue.

Proposition 3.8. *Under the stability condition, we have*

$$\sigma(M) = \frac{\Lambda}{\mu_f + \mu_c} + o(M),$$

which implies

$$\lim_{M\to\infty} \sigma(M) = \frac{\Lambda}{\mu_f + \mu_c}.$$

When M becomes very large, it will behave like a two-independent $M/M/1$ queues with utilization λ_f/μ_f and λ_c/μ_c respectively. In the long-run, it gives a chance $\mu_f/(\mu_f+\mu_c)$ or $\mu_c/(\mu_f+\mu_c)$ for individual queues in average at the total service rates. Therefore the eigenvalue of a matrix associated with the QBD process for the system is the traffic intensity, which is

$$\frac{\lambda_f}{\mu_f}\cdot\frac{\mu_f}{\mu_f+\mu_c}+\frac{\lambda_c}{\mu_c}\cdot\frac{\mu_c}{\mu_f+\mu_c}=\frac{\Lambda}{\mu_f+\mu_c}.$$

It was easily proved that the traffic intensity is the eigenvalue of an $M/M/1$ QBD matrix. Note that, in general, $\sigma(M)$ closes to the limit when M is more than 10 in our numerical tests.

3.3. An efficient algorithm to solve $\boldsymbol{\pi}\mathbf{Q}=\mathbf{0}$

We define the $(M+1)\times(M+1)$ matrix $\mathbf{T}_i$ as follows

$$\begin{aligned}\mathbf{T}_0 &= \mathbf{I},\\ \mathbf{T}_1 &= -\mathbf{C}_{0,0}\mathbf{D}^{-1},\\ \mathbf{T}_i &= -(\mathbf{T}_{i-2}\mathbf{A}_{i-2,i-1}+\mathbf{T}_{i-1}\mathbf{C}_{i-1,i-1})\mathbf{D}^{-1}, \quad 2\le i\le n_0.\end{aligned}$$

Proposition 3.9. *The stationary probability* $\boldsymbol{\pi}_0$ *satisfies the following two sets of equations:*

(i.)
$$\boldsymbol{\pi}_0(\mathbf{T}_{n_0-1}\mathbf{A}_{n_0-1,n_0}+\mathbf{T}_{n_0}(\mathbf{C}+\mathbf{RD}))=\mathbf{0}.$$

(ii.)
$$\boldsymbol{\pi}_0\left[\sum_{i=0}^{n_0-1}\mathbf{T}_i\mathbf{1}+\mathbf{T}_{n_0}\begin{bmatrix}1\\ \vdots\\ 1\\ \tilde{h}\end{bmatrix}\right]=1,$$

where
$$\tilde{h}=1+\frac{-\Lambda}{(1-\sigma)}\sum_{i=1}^{M}r_i,\quad r_i \textit{ is the element in } \mathbf{R}$$

and $\boldsymbol{\pi}_i=\boldsymbol{\pi}_0\mathbf{T}_i$*, for* $0\le i\le n_0$.

Proof. Starting from equation (5), and $\mathbf{D}=\mu_f\mathbf{I}$ being invertible, for $0\le n\le n_0$, we have $\mathbf{T}_1=-\mathbf{C}_{00}\mathbf{D}^{-1}$. It is easy to check that $\boldsymbol{\pi}_1=\boldsymbol{\pi}_0\mathbf{T}_1$. By induction, we can also verify that $\boldsymbol{\pi}_i=\boldsymbol{\pi}_0\mathbf{T}_i$, for $0\le i\le n_0$. Thus, repeating it in equation (6) until $n=n_0-1$ and together with equation (7), it gives

$$\boldsymbol{\pi}_0(\mathbf{T}_{n_0-1}\mathbf{A}_{n_0-1,n_0}+\mathbf{T}_{n_0}(\mathbf{C}_{n_0,n_0}+\mathbf{RD}))=\mathbf{0}.$$

Since
$$(\mathbf{I}-\mathbf{R})^{-1}=\mathbf{I}+\mathbf{R}+\mathbf{R}^2+\cdots=\mathbf{I}+\frac{\mathbf{R}}{1-\sigma}$$

equation (8) can be rewritten as

$$\boldsymbol{\pi}_0 \left[\sum_{i=0}^{n_0-1} \mathbf{T}_i \mathbf{1} + \mathbf{T}_{n_0} (\mathbf{I} - \mathbf{R})^{-1} \mathbf{1} \right] = 1$$

which can be further simplified with $(\mathbf{I} - \mathbf{R})^{-1}\mathbf{1}$.

Then we can solve by $\boldsymbol{\pi}_0$ the boundary equations of (i) and (ii) with $(M+1)$ unknowns and $(M+1)$ independent equations. The computational complexity is independent of n_0 that reduces that computing burden greatly, compared with solving a system of linear equations of $n_0 \times (M+1)$ unknowns and $n_0 \times (M+1)$ equations.

Proposition 3.10. *The solution vector* $\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \ldots)$ *of* $\boldsymbol{\pi}\mathbf{Q} = \mathbf{0}$ *can be obtained from*

$$\begin{aligned} \boldsymbol{\pi}_i &= \quad \boldsymbol{\pi}_0 \mathbf{T}_i, \quad \forall\ 0 \le i \le n_0, \\ \boldsymbol{\pi}_{n_0+k} &= \quad \boldsymbol{\pi}_{n_0} \sigma^{k-1} \mathbf{R}, \quad \forall \quad k \ge 1. \end{aligned}$$

where σ *is an eigenvalue of* $\mathbf{K}$ *and* $0 < \sigma < 1$.

An Efficient Algorithm

Step 1. Set $\mathbf{K} = -[\mathbf{C} + \boldsymbol{\mathcal{P}}]\mu_f^{-1}$.

Step 2. Find an eigenvalue σ of $\mathbf{K}$ which is less than one and greater than zero.

Step 3. Define $\mathbf{R} = -\Lambda \begin{bmatrix} & \mathbf{0} & \\ r_1 & \cdots & r_M \end{bmatrix}$ by (27).

Step 4. Construct matrices $\mathbf{T}_0 = \mathbf{I}$, $\mathbf{T}_1 = -\mathbf{C}_{0,0}\mathbf{D}^{-1}$ and
$\mathbf{T}_i = -(\mathbf{T}_{i-2}\mathbf{A}_{i-2,i-1} + \mathbf{T}_{i-1}\mathbf{C}_{i-1,i-1})\mathbf{D}^{-1}$ for $2 \le i \le n_0$.

Step 5. Determine $\boldsymbol{\pi}_0$ by solving (i) and (ii) and let $\boldsymbol{\pi}_i = \boldsymbol{\pi}_0 \mathbf{T}_i$, for $0 < i \le n_0$.

Step 6. From $\boldsymbol{\pi}_{n_0+k} = \boldsymbol{\pi}_{n_0}\sigma^{k-1}\mathbf{R}$, for $k \ge 1$, we obtain $\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots)$.

4. Numerical Illustrations

In this section, we compare the two computing approaches, that is, Matrix-Geometric method in Latouche and Ramaswami [16] versus K-matrix based algorithm proposed in this paper. We use the computing language MATLAB to implement the algorithms. The numerical analysis is performed on the PC platform with Intel(R) Core(TM) i7-3770 CPU @ 3.40 GHz and 32 GB RAM. The parameters of the two-tier service queueing model are $\Lambda = 1$, $\mu_f = 0.6$ and $\mu_c = 0.6$ in the following experiments of computing the stationary distribution of queue lengths. With the stationary distribution $\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \boldsymbol{\pi}_2, \ldots)$, we can determine the expected number of customers in queue 1, L_1, and the expected number of customers in queue 2, L_2.

Although we conducted extensive numerical experiments, for the conciseness of this note, we only presented a sample of the results. Table 1 summarizes some results obtained by Matrix-Geometric method and K-matrix method as the finite buffer size M varies from 5 to 95. Figure 3 shows the expected numbers of customers in queue 1 and queue 2, L_1 and L_2, individually. It can be observed that, while the buffer size M increases, the average system sizes (queue lengths) obtained by Matrix-Geometric method and K-matrix method will approach to the same value. Figure 4 compares the CPU time taken by Matrix-Geometric method and K-matrix method while solving two-tier service queueing model under the same parameters $\Lambda = 1$, $\mu_f = 0.6$ and $\mu_c = 0.6$. We find that the K-matrix based algorithm can save huge CPU times compared with the Matrix-Geometric method. As the buffer size M increases, the K-matrix method is a much more efficient algorithm than the traditional Matrix-Geometric algorithms for solving this class of large scale service systems.

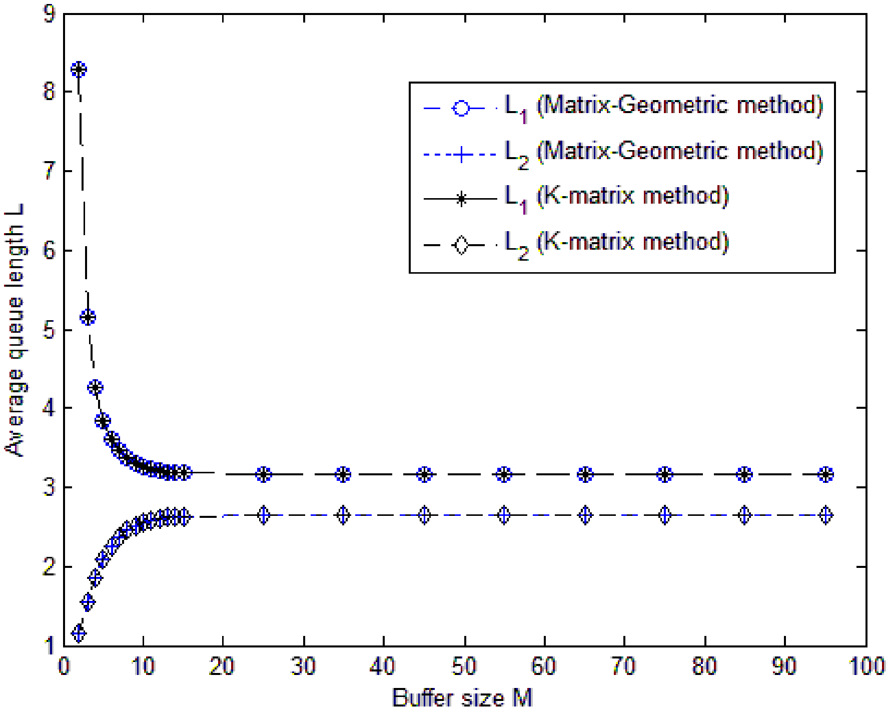

Figure 3. Average system size L_1 (L_2) versus the buffer size M obtained by Matrix-Geometric method and **K**-matrix method with parameters $\Lambda = 1$ and $\mu_f = \mu_c = 0.6$.

5. Conclusions

In this note, we develop a new K-matrix method which is demonstrated to be more efficient than Geometric-Matrix method for solving this two-tier service queueing model (QBD process). The proposed algorithm depends on a right eigenvector and an eigenvalue

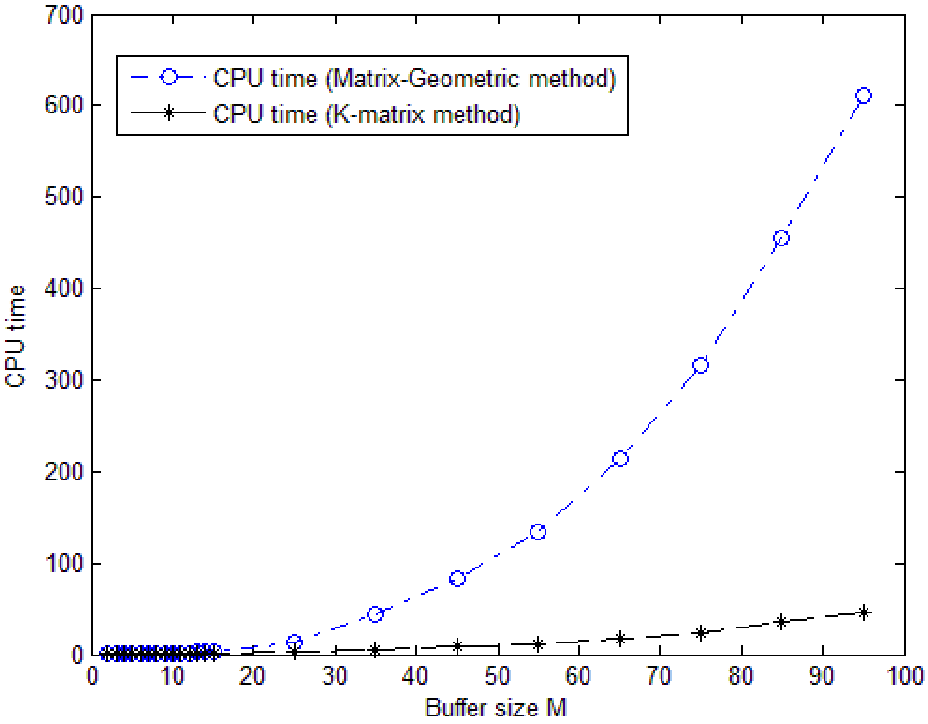

Figure 4. The CPU time versus the buffer size M obtained by Matrix-Geometric method and **K**-matrix method with parameters $\Lambda = 1$ and $\mu_f = \mu_c = 0.6$.

Table 1. Numerical results obtained by Matrix-Geometric method and **K**-matrix method with parameters $\Lambda = 1$ and $\mu_f = \mu_c = 0.6$.

	Matrix-Geometric Method			**K**-matrix Method		
M	L_1	L_2	CPU Time	L_1	L_2	CPU Time
5	3.6126	2.0068	0.3588	3.9458	2.1157	0.1560
15	3.1416	2.6050	0.7956	3.1848	2.6386	0.7020
25	3.1606	2.6475	2.0904	3.1634	2.6500	0.4212
35	3.1626	2.6501	12.6205	3.1627	2.6502	0.7644
45	3.1627	2.6502	24.8198	3.1627	2.6502	2.0436
55	3.1627	2.6502	42.8535	3.1627	2.6502	3.5256
65	3.1627	2.6502	86.7054	3.1627	2.6502	6.1152
75	3.1627	2.6502	159.7138	3.1627	2.6502	12.4333
85	3.1627	2.6502	305.6372	3.1627	2.6502	23.9306
95	3.1627	2.6502	542.7587	3.1627	2.6502	43.1499

which is simply $\Lambda/(\mu_f + \mu_c)$. It is also fairly easy to construct the K-matrix, which only depends on the right eigenvector. This right eigenvector can be determined in the exact form.

We show that when the buffer size of the toll system becomes large, the K-matrix method and Matrix-Geometric method give the same results. However, the computational complexity of K-matrix method is much lower because it only needs to solve the vector $\boldsymbol{\pi}_0$ with $M+1$ variables and the remaining probabilities are attained by substitution. There are many cases where the matrix **A** has only one non-zero row of which examples are found in Ramaswami and Latouch [18] and Tijms and van Vuuren [20]. In particular, two illustrative examples are given in Appendix F, highlighting some important computational features of the K-matrix. The computational effort of this approach suggested here is significantly reduced while the numerical stability associated with the computational procedure is controlled under a preset precision level. Since the matrices **D**, **C**, **A** and $\mathbf{C}_0$ that arise in applications are usually sparse, the results developed in this paper may be used before considering the quadratically convergent algorithms of computing the rate matrix **R**.

Since two-tier service system is a popular setting for many service systems with both public and private service providers, our proposed algorithm provides a powerful tool for practitioners to evaluate the customer service performance.

References

[1] Bright, L., & Taylor, P. G. (1995). Calculating the equilibrium distribution in level dependent quasi-birth-and-death processes. *Stochastic Models*, 11(3), 497–525.

[2] Chen, H., Qian, Q., & Zhang, A. (2012). Would allowing privately funded health care reduce the public waiting time? Empirical evidence from Canadian joint replacement surgery data. Working paper, Sauder School of Business, University of British Columbia.

[3] Dayar, T., & Quessette, F. (2002). Quasi-birth-and-death processes with level geometric distribution. *SIAM journal on matrix analysis and applications*, 24(1), 281–291.

[4] Drekic, S., & Grassmann, W. K. (2002). An eigenvalue approach to analyzing a finite source priority queueing model. *Annals of Operations Research*, 112(1), 139–152.

[5] El-Mikkawy, M., & Karawia, A. (2006). Inversion of general tridiagonal matrices. *Applied Mathematics Letters*, 19(8), 712–720.

[6] Grassmann, W. K. (2002). Real eigenvalues of certain tridiagonal matrix polynomials, with queueing applications. *Linear Algebra and its Applications*, 342(1-3), 93–106.

[7] Grassmann, W. K. (2003). The use of eigenvalues for finding equilibrium probabilities of certain Markovian two-dimensional queueing problems. *INFORMS Journal on Computing*, 15(4), 412–421.

[8] Grassmann, W. K., & Drekic, S. (2000). An analytical solution for a tandem queue with blocking. *Queueing Systems*, 36(1), 221–235.

[9] Grassmann, W. K., & Tavakoli, J. (2010). Comparing some algorithms for solving QBD processes exhibiting special structures. *INFOR: Information Systems and Operational Research*, 48(3), 133–141.

[10] Guo, P., Lindsey, R., & Zhang, Z. G. (2014). On the Downs–Thomson paradox in a self-financing two-tier queuing system. *Manufacturing & Service Operations Management*, 16(2), 315–322.

[11] Hassin, R. (2016). *Rational queueing*. CRC Press.

[12] Horn, R. A., & Johnson, C. R. (2012). *Matrix Analysis*. Cambridge University Press.

[13] Hua, Z., Chen, W., & Zhang, Z. G. (2016). Competition and coordination of two-tier service systems. working paper. College of Business and Economics, Western Washington University.

[14] Konheim, A. G., & Reiser, M. (1978). Finite capacity queuing systems with applications in computer modeling. *SIAM Journal on Computing*, 7(2), 210–229.

[15] Latouche, G., & Neuts, M. F. (1980). Efficient algorithmic solutions to exponential tandem queues with blocking. *SIAM Journal on Algebraic Discrete Methods*, 1(1), 93–106.

[16] Latouche, G., & Ramaswami, V. (1999). *Introduction to Matrix Analytic Methods in Stochastic Modelling*. ASA & SIAM, Philadelphia, USA.

[17] Neuts, M. F. (1981) *Matrix-Geometric Solutions in Stochastic Models*. The John Hopkins University Press.

[18] Ramswami, V., & Latouche, G. (1986). A general class of Markov processes with explicit matrix-geometric solutions. *Operations-Research-Spektrum*, 8(4), 209–218.

[19] Sturm, J. C. F. (1857). Mémoire sur la résolution des équations numériques. *Mém Savans Etrang*, 271–318.

[20] Tijms, H. C., & Van Vuuren, D. J. (2002). Markov Processes on a semi-infinite strip and the geometric tail algorithm. *Annals of Operations Research*, 113(1), 133-140.

Appendix

A. Proof of Proposition 2.1

Proof. We use a case of $M = 3$ to prove this proposition. Letting $\overline{\mathbf{D}} = \mathbf{D} + \mathbf{C} + \mathbf{A}$. $\overline{\mathbf{D}}$ is irreducible. Then, we have

$$\overline{\mathbf{D}} = \begin{bmatrix} -\Lambda & \Lambda & & \\ \mu_c & -(\Lambda+\mu_c) & \Lambda & \\ & \mu_c & -(\Lambda+\mu_c) & \Lambda \\ & & \mu_c & -\mu_c \end{bmatrix}.$$

Denote the stationary vector for $\overline{\mathbf{D}}$ by $\mathbf{p} = (p_0, p_1, p_2, p_3)$. Solving $\mathbf{p}\overline{\mathbf{D}} = \mathbf{0}$, we obtain $p_0 = 1/\sum_{i=0}^{3}(\Lambda/\mu_c)^i, p_1 = (\Lambda/\mu_c)/\sum_{i=0}^{3}(\Lambda/\mu_c)^i, p_2 = (\Lambda/\mu_c)^2/\sum_{i=0}^{3}(\Lambda/\mu_c)^i$, and $p_3 = (\Lambda/\mu_c)^3/\sum_{i=0}^{3}(\Lambda/\mu_c)^i$. Based on the drift stability condition of $\mathbf{pA1} < \mathbf{pD1}$, we have

$$\mu_f > p_3\Lambda = \frac{(\Lambda/\mu_c)^3\Lambda}{\sum_{i=0}^{3}(\Lambda/\mu_c)^i} = \frac{\left(1-\frac{\Lambda}{\mu_c}\right)\left(\frac{\Lambda}{\mu_c}\right)^3\Lambda}{1-\left(\frac{\Lambda}{\mu_c}\right)^4}.$$

For a general case with buffer size M, we get $\mu_f > p_M\Lambda = \left(1 - \frac{\Lambda}{\mu_c}\right)\left(\frac{\Lambda}{\mu_c}\right)^M \Lambda \left(1 - \left(\frac{\Lambda}{\mu_c}\right)^{M+1}\right)^{-1}$ which is the condition in Proposition 1.

B. Proof of Corollary 2.2

Proof. If $\Lambda/\mu_c < 1$, it is easy to see that the left hand side (l.h.s.) of (12) approaches to zero as $M \longrightarrow \infty$ or we have $\mu_f > 0$. For $\Lambda/\mu_c = 1$, to evaluate the l.h.s, we use L'Hôpital's rule. By taking the derivative of both numerator and denominator with respect to Λ/μ_c, we have

$$\lim_{\Lambda/\mu_c \longrightarrow 1} \frac{\left(1 - \frac{\Lambda}{\mu_c}\right)\left(\frac{\Lambda}{\mu_c}\right)^M \Lambda}{1 - \left(\frac{\Lambda}{\mu_c}\right)^{M+1}} = \lim_{\Lambda/\mu_c \longrightarrow 1} \frac{M\left(\frac{\Lambda}{\mu_c}\right)^{M-1} - (M+1)\left(\frac{\Lambda}{\mu_c}\right)^M}{-(M+1)\left(\frac{\Lambda}{\mu_c}\right)^M}\Lambda = \frac{\Lambda}{M+1}.$$

Thus as $M \longrightarrow \infty$, we again have $\mu_f > 0$. If $\Lambda/\mu_c > 1$, we again use L'Hôpital's rule to evaluate the l.h.s. of (12). Taking the derivative of both numerator and denominator with respect to M, we have

$$\lim_{M \longrightarrow \infty} \frac{\left(1 - \frac{\Lambda}{\mu_c}\right)\left(\frac{\Lambda}{\mu_c}\right)^M \Lambda}{1 - \left(\frac{\Lambda}{\mu_c}\right)^{M+1}} = \lim_{M \longrightarrow \infty} \frac{\left(\frac{\Lambda}{\mu_c} - 1\right)\Lambda\left(\frac{\Lambda}{\mu_c}\right)^M \ln\left(\frac{\Lambda}{\mu_c}\right)}{\left(\frac{\Lambda}{\mu_c}\right)^{M+1} \ln\left(\frac{\Lambda}{\mu_c}\right)} = \frac{\left(\frac{\Lambda}{\mu_c} - 1\right)\Lambda}{\frac{\Lambda}{\mu_c}},$$

which leads (12) to $\mu_f > \left(\frac{\Lambda}{\mu_c} - 1\right)\Lambda/\left(\frac{\Lambda}{\mu_c}\right)$. Simplifying it yields $\mu_f + \mu_c > \Lambda$.

C. Proof of Lemma 2.5

Proof. Note that $\mathbf{K} = -[\mathbf{C} + \boldsymbol{\mathcal{P}}]\mathbf{D}^{-1}$

$$= \begin{bmatrix} \frac{\mu_f+\Lambda}{\mu_f} & \frac{-\Lambda}{\mu_f} & & & -1 \\ \frac{-\mu_c}{\mu_f} & \frac{\mu_f+\mu_c+\Lambda}{\mu_f} & \ddots & & \vdots \\ & \ddots & \ddots & \ddots & -1 \\ & & \ddots & \frac{\mu_f+\mu_c+\Lambda}{\mu_f} & \frac{-\mu_f-\Lambda}{\mu_f} \\ & & & \frac{-\mu_c}{\mu_f} & \frac{\mu_c+\Lambda}{\mu_f} \end{bmatrix}.$$

Since the diagonal entry of **K** is positive, we can choose a upper triangular matrix **U** and lower triangular matrix **L** such that $\mathbf{K} = \mathbf{L} + \mathbf{U}$ and the diagonal entry of **L** and **U** are positive. Clearly, **L** and **U** are positive definite because eigenvalues of **L** and **U** are diagonal entries. Since the addition of two positive define matrix is again positive define, it concludes that **K** is also positive definite. It suffices to show that **K** is positive definite and its eigenvalues are positive.

D. Proof of Lemma 2.10

Proof. We solve the following equation for x,

$$\begin{aligned} < \mathbf{K}_2 - x\mathbf{I} > \quad &= \begin{bmatrix} \frac{\Lambda+\mu_f}{\mu_f} - x & \frac{-(\Lambda+\mu_f}{\mu_f} \\ \frac{-\mu_c}{\mu_f} & \frac{\Lambda+\mu_c}{\mu_f} - x \end{bmatrix}. \\ &= x^2 - (\tfrac{2\Lambda}{\mu_f} + \tfrac{\mu_c}{\mu_f} + 1)x + \tfrac{\Lambda}{\mu_f}(1 + \tfrac{\Lambda}{\mu_f}) \\ &= 0 \end{aligned}$$

By a quadratic formula, x is given by

$$x = \frac{(\frac{2\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f} + 1) \pm ((\frac{\mu_c}{\mu_f})^2 + \frac{4\Lambda\mu_c}{\mu_f^2} + \frac{2\mu_c}{\mu_f} + 1)^{\frac{1}{2}}}{2}$$

It is easy to check that

$$\begin{aligned} &\frac{1}{2}\{(\frac{2\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f} + 1) + [(\frac{\mu_c}{\mu_f})^2 + \frac{4\Lambda\mu_c}{\mu_f^2} + \frac{2\mu_c}{\mu_f} + 1]^{\frac{1}{2}}\} \\ &> \frac{1}{2}\{(\frac{2\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f} + 1) + [(1 + \frac{\mu_c}{\mu_f})^2]^{\frac{1}{2}}\} \\ &= \frac{1}{2}\{\frac{2\Lambda}{\mu_f} + 2 + \frac{2\mu_c}{\mu_f}\} > 1. \end{aligned}$$

We consider the difference of two terms in the numerator of x. First,

$$\begin{aligned} &\left((\frac{2\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f} + 1)^2\right)^{\frac{1}{2}} = \left((\frac{\mu_c}{\mu_f} + 1)^2 + \frac{4\Lambda}{\mu_f}\frac{\mu_c}{\mu_f} + \frac{4\Lambda}{\mu_f} + \frac{4\Lambda^2}{\mu_f^2}\right)^{\frac{1}{2}} \\ &> \left((\frac{\mu_c}{\mu_f} + 1)^2 + \frac{4\Lambda}{\mu_f}\frac{\mu_c}{\mu_f}\right)^{\frac{1}{2}} \end{aligned}$$

Thus, we have their difference is greater than 0. Second, recall (12) $\frac{\mu_c}{\Lambda}(1 + \frac{\Lambda}{\mu_c}) > \frac{\Lambda}{\mu_f}$, i.e., $(\frac{\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f}) > \frac{\Lambda^2}{\mu_f^2}$. Now, check,

$$\begin{aligned} &(2\frac{\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f} - 1)^2 = \frac{4\Lambda^2}{\mu_f^2} + 1 + (\frac{\mu_c}{\mu_f})^2 - \frac{2\mu_c}{\mu_f} + \frac{4\Lambda\mu_c}{\mu_f^2} - \frac{4\Lambda}{\mu_f} \\ &< 4(\frac{\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f}) + 1 + (\frac{\mu_c}{\mu_f})^2 - \frac{2\mu_c}{\mu_f} + \frac{4\Lambda\mu_c}{\mu_f^2} - \frac{4\Lambda}{\mu_f} \\ &= (\frac{\mu_c}{\mu_f})^2 + 1 + \frac{2\mu_c}{\mu_f} + \frac{4\Lambda\mu_c}{\mu_f^2}. \end{aligned}$$

Therefore, we have

$$2\frac{\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f} < \{(\frac{\mu_c}{\mu_f})^2 + 1 + \frac{2\mu_c}{\mu_f} + \frac{4\Lambda\mu_c}{\mu_f^2}\}^{\frac{1}{2}} + 1.$$

Equivalently, it implies

$$(2\frac{\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f} + 1) - \{(\frac{\mu_c}{\mu_f})^2 + 1 + \frac{2\mu_c}{\mu_f} + \frac{4\Lambda\mu_c}{\mu_f^2}\}^{\frac{1}{2}} < 2$$

Thus, we conclude

$$\frac{1}{2}(2\frac{\Lambda}{\mu_f} + 1 + \frac{\mu_c}{\mu_f}) - \frac{1}{2}[(1 + \frac{\mu_c}{\mu_f})^2 + \frac{4\Lambda\mu_c}{\mu_f^2}]^{\frac{1}{2}} < 1.$$

It yields

$$\sigma = \frac{(\frac{2\Lambda}{\mu_f} + \frac{\mu_c}{\mu_f} + 1) - ((\frac{\mu_c}{\mu_f})^2 + \frac{4\Lambda\mu_c}{\mu_f^2} + \frac{2\mu_c}{\mu_f} + 1)^{\frac{1}{2}}}{2}$$

$$0 < \sigma < 1..$$

E. Elements in R

With modification of the formula, r_i can be calculated by solving the difference equations in El-Mikkawy and Karawia (2006) and the final forms are given below

$$\begin{cases} r_i = (-\mu_c)^{(M+1-i)} \frac{t_i}{\beta_{i+1}} & i = 1, 2, 3, \cdots, M \\ r_{M+1} = (d - \frac{\mu_c \Lambda a_{M-1}}{a_M})^{-1} \end{cases} \tag{27}$$

where

$$\begin{cases} t_1 = (d_1 - \frac{\mu_c\Lambda\beta_3}{\beta_2})^{-1} \\ t_i = (d - \frac{\mu_c\Lambda a_{i-2}}{a_{i-1}} - \frac{\mu_c\Lambda\beta_{i+2}}{\beta_{i+1}})^{-1} & i = 2, 3, \cdots, M \end{cases}$$

$$\text{with} \qquad d = -(\Lambda + \mu_f + \mu_c) + \sigma\mu_f$$

$$d_1 = -(\Lambda + \mu_f) + \sigma\mu_f$$

and

$$\begin{cases} a_i = (1-k) \times (u_1)^i + k \times (u_2)^i & i = 2, 3, \cdots, M \\ a_0 = 1, \quad a_1 = d_1 \\ k = \frac{d_1 - u_1}{u_2 - u_1} \end{cases}$$

$$\begin{cases} \beta_i = (1-g) \times (u_1)^{M+2-i} + g \times (u_2)^{M+2-i} & i = 2, 3, \cdots, M \\ \beta_{M+2} = 1, \quad \beta_{M+1} = d \\ \beta_1 = d_1\beta_2 - \mu_c\Lambda\beta_3 \end{cases}$$

$$\text{with} \qquad g = \frac{d - u_1}{u_2 - u_1}$$

$$u_1 = \frac{d + \sqrt{d^2 - 4\mu_c\Lambda}}{2\mu_c\Lambda}$$

$$u_2 = \frac{d - \sqrt{d^2 - 4\mu_c\Lambda}}{2\mu_c\Lambda}$$

From the introduction in Neuts (1981), **R** in (22) records the rate of sojourn in the states of the toll queue and $(n+1)$ customers in the free queue per unit of the sojourn time when there are n customers in the free queue as $n \geq n_0$.

F. Illustrative Examples for Constructing K

Besides the two-tier service case with QBD process, we borrow two other examples from Dayar and Quessette (2002) to illustrate the construction of **K**.

Example 1: Consider the continuous-time equivalent of the discrete-time QBD process discussed in [pp. 668-669 in Latouche and Ramaswami (1999)]. The model has 2 phases at each level (i.e., m = 2). Assuming that $0 < p < 1/2$, the process moves from state $(\ell, 1)$, $\ell \geq 1$ to $(\ell, 2)$ with rate p, and to $(\ell - 1, 1)$ with rate $(1 - p)$. The process moves from state $(\ell, 2)$, $\ell \geq 0$, to $(\ell, 1)$ with rate $2p$, and to $(\ell + 1, 2)$ with rate $(1 - 2p)$. Finally the process moves from state $(0, 1)$ to $(0, 2)$ with rate 1. All diagonal elements of Q are -1. Hence, we have

$$\mathbf{A}_0 = \mathbf{A} = \begin{pmatrix} 0 & 0 \\ 0 & 1-2p \end{pmatrix}, \mathbf{C} = \begin{pmatrix} -1 & p \\ 2p & -1 \end{pmatrix}, \mathbf{D}_1 = \mathbf{D} = \begin{pmatrix} 1-p & 0 \\ 0 & 0 \end{pmatrix}, \quad \mathbf{C}_0 = \begin{pmatrix} -1 & 1 \\ 2p & -1 \end{pmatrix},$$

Define

$$\boldsymbol{\mathcal{P}} = \begin{pmatrix} 0 & 0 \\ 1-2p & 0 \end{pmatrix} \text{ and } \mathbf{d}_n = \boldsymbol{\pi}_n - \boldsymbol{\pi}_{n-1}$$

Since

$$\begin{aligned} \boldsymbol{\pi}_n\mathbf{A} + \boldsymbol{\pi}_{n+1}\mathbf{C} + \boldsymbol{\pi}_{n+2}\mathbf{D} = \mathbf{0}, \qquad & n \geq 0, \\ \mathbf{d}_n\mathbf{A1} + \mathbf{d}_{n+1}\mathbf{C1} + \mathbf{d}_{n+2}\mathbf{D1} = 0, \qquad & n \geq 1 \\ \boldsymbol{\pi}_n\boldsymbol{\mathcal{P}} + \boldsymbol{\pi}_{n+1}(-\boldsymbol{\mathcal{P}} - \mathbf{D}) + \boldsymbol{\pi}_{n+2}\mathbf{D} = \mathbf{0} \end{aligned}$$

$$\mathbf{d}_n\boldsymbol{\mathcal{P}} = \mathbf{d}_{n+1}\mathbf{D}$$

$$\mathbf{d}_n(\mathbf{A} - \boldsymbol{\mathcal{P}}) + \mathbf{d}_{n+1}(\mathbf{C} + \boldsymbol{\mathcal{P}} + \mathbf{D}) = \mathbf{0}$$

$$\mathbf{d}_n\mathbf{A} = -\mathbf{d}_{n+1}(\boldsymbol{\mathcal{P}} + \mathbf{C})$$

$$\mathbf{K} = -\mathbf{A}(\mathbf{C} + \boldsymbol{\mathcal{P}})^{-1} = \frac{1-2p}{1-p}\begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$$

we have an eigenvalue σ, where $0 < \sigma = \frac{1-2p}{1-p} < 1$. In this case, it gives

$$\mathbf{R} = -\mathbf{A}(\mathbf{C} + \alpha\mathbf{D})^{-1} = \begin{pmatrix} 0 & 0 \\ 0 & 1-2p \end{pmatrix} \frac{1}{2p(1-p)} \begin{pmatrix} 1 & p \\ 2p & 2p \end{pmatrix} = \frac{1-2p}{1-p}\begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} = \mathbf{K}.$$

Example 2: Consider the Em/M/1 FCFS queue which has an exponential service distribution with rate μ and an m-phase Erlang arrival process with rate $m\lambda$ in each phase [pp. 206-208 in

Latouche and Ramaswami (1999)]. The expected interarrival time and the expected service time of this queue are respectively $1/\lambda$ and $1/\mu$. We assume $\lambda < \mu$. The queue corresponds to a QBD process with level representing the queue length (including any in service) and phase representing the state of the Erlang arrival process. Letting $d = m\lambda + \mu$ we have the $(m \times m)$ matrices $\mathbf{A}_0 = \mathbf{A} = m\lambda \mathbf{e}_m \mathbf{e}_1^T$, $\mathbf{D}_1 = \mathbf{D} = \mu\mathbf{I}$.

$$\mathbf{C} = \begin{pmatrix} -d & m\lambda & & \\ & \cdots & \cdots & \\ & & -d & m\lambda \\ & & & -d \end{pmatrix}, \quad \mathbf{C}_0 = \begin{pmatrix} -m\lambda & m\lambda & & \\ & \cdots & \cdots & \\ & & -m\lambda & m\lambda \\ & & & -m\lambda \end{pmatrix},$$

Define

$$\boldsymbol{\mathcal{P}} = \begin{pmatrix} \mu & 0 & \cdots & 0 \\ \mu & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \mu & 0 & \cdots & 0 \end{pmatrix}$$

$$\boldsymbol{\pi}_n \mathbf{A1} + \boldsymbol{\pi}_{n+1}\mathbf{C1} + \boldsymbol{\pi}_{n+2}\mathbf{D1} = \mathbf{0}, \qquad n \geq L, \tag{28}$$

$$\boldsymbol{\pi}_n \mathbf{A} + \boldsymbol{\pi}_{n+1}(-\boldsymbol{\mathcal{P}} - \mathbf{A}) + \boldsymbol{\pi}_{n+2}\boldsymbol{\mathcal{P}} = \mathbf{0} \tag{29}$$

It yields

$$\mathbf{d}_{n+2}\boldsymbol{\mathcal{P}} = \mathbf{d}_{n+1}\mathbf{A}.$$

Without right multiplication of **1** in (28), taking the difference of (28) and (29) multiplied by $\mathbf{d}_n$, we have

$$\mathbf{d}_{n+2}(\mathbf{D} - \boldsymbol{\mathcal{P}}) + \mathbf{d}_{n+1}(\mathbf{C} + \boldsymbol{\mathcal{P}} + \mathbf{A}) = \mathbf{0},$$

which may be simplified to

$$\mathbf{d}_{n+2}\mathbf{D} = -\mathbf{d}_{n+1}(\boldsymbol{\mathcal{P}} + \mathbf{C}).$$

Therefore,

$$\mathbf{K} = -(\mathbf{C} + \boldsymbol{\mathcal{P}})\mathbf{D}^{-1}$$

Let $\rho = \frac{m\lambda}{\mu}$. For the stability condition, we have $\rho < m$. We write the characteristic polynomial as

$$\begin{aligned} \ell(x) &= \langle \mathbf{K} - x\mathbf{I} \rangle \\ &= -\sum_{i=0}^{m-1} C_i^m (1-x)^{m-i-1} x\rho^i + \rho^m \end{aligned}$$

$$\text{where} \quad C_i^m = \frac{m!}{i!(m-i)!}.$$

For example, when $m = 2$ and $\rho = 2\lambda/\mu$, $\ell(x) = -(1-x)x - 2x\rho + \rho^2$. The eigenvalue of **K**, is less than 1, i.e.,

$$0 < \sigma = \rho + \frac{1}{2} - (\rho + \frac{1}{4})^{1/2} < 1.$$

For $m > 2$, we can easily check there is one eigenvalue σ between 0 and 1 by using Descartes' root test with taking $\ell(0) \times \ell(1)$ which is $\rho^m \times \rho^{m-1}(\rho - m) < 0$. Numerically, **R** can be obtained with σ given from **K**.