Contents lists available at ScienceDirect

Performance Evaluation

journal homepage: www.elsevier.com/locate/peva

System-theoretical algorithmic solution to waiting times in semi-Markov queues

N. Akar[a], K. Sohraby[b,*]

[a] Electrical and Electronics Engineering Department, Bilkent University, Bilkent 06800, Ankara, Turkey

[b] Computer Science and Electrical Engineering, University of Missouri-Kansas City, Kansas City, MO 64110, USA

ARTICLE INFO

Article history:
Received 6 May 2008
Received in revised form 9 February 2009
Accepted 1 May 2009
Available online 7 May 2009

Keywords:
Semi-Markov queues
Lindley equation
Correlated arrivals and services
Schur decomposition
Matrix-analytical approach

ABSTRACT

Markov renewal processes with matrix-exponential semi-Markov kernels provide a generic tool for modeling auto-correlated interarrival and service times in queueing systems. In this paper, we study the steady-state actual waiting time distribution in an infinite capacity single-server semi-Markov queue with the auto-correlation in interarrival and service times modeled by Markov renewal processes with matrix-exponential kernels. Our approach is based on the equivalence between the waiting time distribution of this semi-Markov queue and the output of a linear feedback interconnection system. The unknown parameters of the latter system need to be determined through the solution of a SDC (Spectral-Divide-and-Conquer) problem for which we propose to use the ordered Schur decomposition. This approach leads us to a completely matrix-analytical algorithm to calculate the steady-state waiting time which has a matrix-exponential distribution. Besides its unifying structure, the proposed algorithm is easy to implement and is computationally efficient and stable. We validate the effectiveness and the generality of the proposed approach through numerical examples.

1. Introduction

In this paper, we study the steady-state waiting time in a single-server queue in which there is auto-correlation both in interarrival and service times. We model such auto-correlations by using a Markov renewal process or sequence (denoted by MRP) with finite state-space for both interarrivals and services with the associated semi-Markov kernels in matrix-exponential form. In particular, we study the Lindley equation in continuous-time:

$$W_{k+1} = (W_k + B_k - A_k)^+ = \max(0, W_k + B_k - A_k), \quad k \geq 0, \tag{1}$$

where A_k (designating interarrival times) and B_k (designating service times) are Markov renewal processes and the distribution of $W = \lim_{k\to\infty} W_k$ needs to be obtained algorithmically (when it exists), which is the scope of the current paper. Above, for $k \geq 0$, A_k and B_k are independent of each other and also of W_k. The case of cross-correlation between A_k and B_k leads to dependent queues which are outside the scope of the current paper; see [3] for a review of the early literature and [4,5] for more recent studies on this topic. The random variable W designates the steady-state waiting time for the queueing system described in (1) which is known to be the semi-Markov queue or the SM/SM/1 queue in short [6]. We note that the Lindley equation (1) perfectly describes the queue waiting time in a queueing system with renewal-type services using the first come first serve (FCFS) service discipline. However, the situation is different when the service times

[*] Corresponding author. Tel.: +1 816 235 1232.
E-mail addresses: akar@ee.bilkent.edu.tr (N. Akar), sohrabyk@umkc.edu (K. Sohraby).

0166-5316/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.peva.2009.05.001

are of the more general Markov renewal type in which case the Lindley recurrence may come short of accommodating different models based on how the service process behaves when the queue is empty [7]. In the current paper, we focus on queues that are described solely by the Lindley equation (1) as in [6]. For the case of non-renewal service times, more general models, for instance the ones discussed in [7], with more general descriptions than that of the recurrence (1), are left for future research.

Queueing systems with arrivals governed by a semi-Markov process have been studied extensively since an SM/M/1 queue was first analyzed in [8]. On the other hand, [9] uses a matrix factorization method for the more general SM/G/1 queue. The Reference [10] studies the SM/PH/1 queue and introduces a non-linear matrix equation for calculating the waiting time distribution. However, the proposed iterative techniques for solving the matrix equation are relatively slow due to their linear convergence rates. The more general SM/SM/1 queue is studied in [6] by solving Wiener-Hopf equations, and waiting times are found without having to find queue occupancy probabilities first. However, the method is based on transform domain calculations and polynomial root finding; therefore it may not scale to large-scale problems. The work in [4] is similar to ours in the sense of solving the semi-Markov queue described by the Lindley equation, and [4] also allows dependence between interarrival and service times, but their models are relatively limited compared with ours and their methods again rely on transform domain calculations. Recently, a spectral decomposition approach is proposed in [11] for the SM/SM/1 queue using the calculation of eigenvalues and eigenvectors of a so-called coupling matrix and potential use of complex arithmetic. However, calculating eigenvectors is known to be error-prone, especially for closely located eigenvalues or eigenvalues with multiplicity.

The alternative approach is the matrix-analytical approach pioneered by Neuts which does not rely on calculating polynomial roots or eigenvectors. In the matrix-analytical approach, the queue occupancy is observed at certain embedded epochs and a structured Markov chain (of M/G/1 or G/M/1 type) is constructed for the queue length, for instance for the (B)MAP/G/1 queue [12,13]. The key to the matrix-analytical approach is the solution to a nonlinear matrix equation which can be solved by quadratical convergence rate algorithms like the logarithmic reduction algorithm for [14] and the iterative scheme of [15] for QBD (Quasi Birth and Death) type Markov chains, and the cyclic reduction algorithm of [16], the invariant subspace approach of [17], and the technique proposed in [18] for M/G/1 type Markov chains. Once a solution for this matrix equation is obtained, one can then find the queue length probabilities recursively [19]. Given the steady-state queue length probabilities, the waiting time distribution and its moments can be obtained, although not in a very compact form [2]. We refer the reader to [20] and [13] for an extensive treatment of the matrix-analytical approach. Studies related to auto-correlated service times are less common and we now list a few from the existing literature. A MAP/MSP/1 queue is studied in [7] using matrix-analytical techniques in terms of the queue length, but waiting time results are not given. The Reference [21] studies a queue whose service speed changes according to an external environment governed by a Markov process.

The goal of the current paper is to provide a computationally efficient and stable algorithmic method to compute the actual waiting time distribution for the semi-Markov queue described by the Lindley recurrence (1). This problem is the same as the one stated in [6]. The starting point of the current paper is the assumption of the semi-Markov kernels of interarrival times and service times having matrix-exponential representations. A semi-Markov kernel (or kernel matrix) denoted by $F(t)$ is said to have a matrix-exponential (ME) representation if the kernel $F(t)$ satisfies:

$$F(t) = V\mathrm{e}^{tT}U + F, \quad t \geq 0, \tag{2}$$

where $F(t)$ is square of size n and T is of size m. In this representation, n is the number of states and m is the number of modes of the underlying MRP. Throughout this paper, we call such processes MRP-ME. An MRP-ME is a generalization of some well-known processes like phase-type (PH-type) [22] and ME-type renewal processes [23], Markovian arrival process (MAP) [24], rational arrival process (RAP) [25], and some batch arrival models, for instance the batch Poisson model. Assuming that n_A (n_B) and m_A (m_B) denote the number of states and the number of modes of the underlying MRP-ME for interarrival times (service times), our main result is that steady-state waiting time for the corresponding semi-Markov queue has a matrix-exponential distribution with $n_A m_B$ modes, and finding the coefficients of this distribution amounts to solving a particular spectral-divide-and-conquer (SDC) problem applied to a matrix of size $n_A m_B + n_B m_A$. Similar results have also been obtained in the discrete-time setting, but for renewal arrivals and services only in [26]. Given a matrix A, the SDC problem of interest in this paper is finding an orthogonal matrix Q such that

$$Q^{\mathrm{T}}AQ = \begin{bmatrix} A_{++} & A_{+-} \\ 0 & A_{--} \end{bmatrix}, \tag{3}$$

where the eigenvalues of A_{++} are exactly the same as the eigenvalues of A with positive or zero real parts. The advantages of the proposed approach are:

- The approach benefits from being purely matrix-analytical, as explained in [22,13], and the generality of the MRP-ME model allows one to use a single unifying algorithm for different well-known queueing models. While doing so, we do not need to construct the embedded Markov chain as in the matrix-analytical approach of Neuts and we directly obtain the waiting time distribution and its moments using expressions that appear to be much simpler than the ones available in the literature. However, we note that we do not give expressions for the queue length in our approach.

- As the numerical engine, we propose to use the ordered Schur decomposition which is known to be the standard serial algorithm for solving the SDC problem in the numerical linear algebra literature due to its well-established numerical stability and computational efficiency [27]. The stability of the Schur decomposition approach stems from its ability to avoid the calculation of all eigenvectors. We calculate only one single eigenvector for which we have a closed form expression as opposed to numerically calculating all eigenvectors, as in [4]. All other steps of the proposed algorithm are based on standard vector and matrix operations and are very easy to implement using a linear algebra package like MATLAB.
- In the definition of the MRP-ME, we allow point masses at the origin, which allows us to model batch point processes as well, with the same structure. This flexibility has the potential to reduce the need for separate algorithms for queueing systems with batch arrivals or services.
- In case the interarrival and service MRP-MEs are renewal processes, i.e., $n_A = n_B = 1$, then the SDC problem is to apply to a matrix of additive size $m_A + m_B$ as opposed to multiplicative size, which is a significant advantage for this special case.
- In the special case of $n_B = m_A = 1$, we propose a Householder transformation for the SDC problem which is computationally more efficient than ordered Schur decomposition.

The remainder of the paper is organized as follows. Section 2 provides preliminaries and notation used in the paper. Section 3 describes the ME distribution and Markov renewal processes with matrix-exponential kernels and how they relate to well-known stochastic models used in queueing literature. We introduce a state-space algebra for such distributions in Section 4. We provide our results on the SM/SM/1 queue in Section 5. In Section 6, we present our matrix-analytical algorithm for the SM/SM/1 queue. Section 7 addresses a modified Lindley recurrence within the same framework. Section 8 provides numerical examples to validate the effectiveness of the proposed approach. We conclude in the final section.

2. Preliminaries and notation

We use uppercase (lowercase) letters to denote matrices (vectors or scalars). We use $g(\cdot)$ to denote a density; g_X denotes the density of the random variable X. We use $F(\cdot)$ to denote a semi-Markov kernel (matrix) and $G(\cdot)$ its kernel density matrix. I and e denote the identity matrix and a column matrix of ones of appropriates sizes, respectively. Let $A = \{A_{ij}\}$ be an $n \times m$ matrix. Then the vectorized form of A is denoted by vec(A):

$$\text{vec}(A) = (A_{11}, A_{12}, \ldots, A_{1m}, A_{21}, A_{22}, \ldots, A_{nm}) \,. \tag{4}$$

The adjoint of a matrix is defined through $A^{-1} = \text{adj}(A)/\det(A)$. Given a $p \times q$ matrix B, the Kronecker product of the matrices A and B is denoted by $A \otimes B$ and the size of $A \otimes B$ is $np \times mq$. We also have $(A \otimes B)(C \otimes D) = AC \otimes BD$. A^{T} denotes A transposed and the matrix A is orthogonal if $A^{\mathrm{T}}A = I$. For a given real, non-symmetric, and square matrix A and a region $\mathcal{D}$ of the complex plane, we will need to find an orthogonal matrix Q such that

$$Q^{\mathrm{T}}AQ = \begin{bmatrix} A_{\mathcal{D}\mathcal{D}} & A_{\mathcal{D}\bar{\mathcal{D}}} \\ 0 & A_{\bar{\mathcal{D}}\bar{\mathcal{D}}} \end{bmatrix}, \tag{5}$$

where the eigenvalues of $A_{\mathcal{D}\mathcal{D}}$ are exactly the same as the eigenvalues of A in $\mathcal{D}$. This problem is called the ordinary SDC problem [28]. A real square matrix A of size n can be transformed via an orthogonal transformation U into the so-called real Schur form by writing $U^{\mathrm{T}}AU = R$ where R is quasi-upper triangular, which means that the matrix R has either 1-by-1 or 2-by-2 diagonal blocks on the diagonal corresponding to the real and complex eigenvalues, respectively, of the matrix A [29]. By reordering the blocks by orthogonal transformations, the eigenvalues are made to appear in any order and one can obtain a matrix Q such that the identity (3) is satisfied and the matrices $A_{\mathcal{D}\mathcal{D}}$ and $A_{\bar{\mathcal{D}}\bar{\mathcal{D}}}$ are quasi-upper triangular and the eigenvalues of $A_{\mathcal{D}\mathcal{D}}$ are the same as those of A in $\mathcal{D}$ [30,31]. This form is called the ordered Schur form and the operation to obtain this form is called the ordered Schur decomposition. We note that obtaining the real Schur form is known to be backward stable and has a complexity of $O(n^3)$ [30]. On the other hand, for the computation of the Jordan form that requires calculation of all eigenvectors, there are no results of guaranteed backward stability and the complexity of this decomposition is much higher than that of the Schur decomposition [32,33]. For this reason, the Reference [32] recommends not to use the Jordan decomposition whenever it is possible and use instead the more reliable Schur form, which we do in the current paper. We note that this view is also shared by [27] in which the standard serial algorithm for the SDC problem is proposed to be the ordered Schur decomposition due to its well-established numerical stability. The ordered Schur form implementations are available in various platforms in LAPACK [34], OCTAVE [35], MATLAB 7.0 [36], and as a public add-on to MATLAB [31]. Although there are other proposed algorithms for the decomposition given by (5) (see [28] for an elaborate discussion of various SDC-solvers), we will focus on the ordered Schur decomposition approach for obtaining numerical results for this paper.

Of particular interest to the current paper is when $\mathcal{D}$ is taken as the closed right-half plane $\mathbb{C}^+ = \{c \in \mathbb{C} : \ Re(c) \geq 0\}$. We also define the complementary set $\mathbb{C}^- = \{c \in \mathbb{C} : \ Re(c) < 0\}$. In the problems studied in this paper, the matrix of interest, say A, turns out to have a single eigenvalue at the origin, and we now attempt to show how to calculate the desired ordered Schur decomposition of A in an example scenario where

$$A = \begin{bmatrix} -1 & -1 & -1/3 & -5/3 \\ -1/3 & -1/3 & 0 & -1/3 \\ 1/7 & 3/7 & 2/7 & 1/7 \\ 0 & 0 & -1/3 & -2/3 \end{bmatrix}.$$

Note that A has a single eigenvalue at the origin with a left null vector $x_L = (-1, 3, 0, 1)$ and a right null vector $x_R = (-3, 2, -2, 1)^{\mathrm{T}}$ such that $x_L A = 0, A x_R = 0$. MATLAB uses finite precision arithmetic to find the eigenvalues of A as $-1.0107 + j1.0967 \times 10^{-1}$, $-1.0107 - j1.0967 \times 10^{-1}$, -1.3341×10^{-16}, and 3.0715×10^{-1}. Although the actual eigenvalue at the origin should be included in $\mathbb{C}^+$, there is the risk of counting it in $\mathbb{C}^-$ since the real part of the corresponding numerical eigenvalue is negative. The solution we propose is to use the MATLAB function `schur.m` that calculates the Schur decomposition of a matrix without any specific ordering of eigenvalues [36]. In particular, we use the following MATLAB command

```
[Q1, T1] = schur(A + xR * xL/(xL * xR))
```

so as to produce $Q_1^{\mathrm{T}}(A + \frac{x_R x_L}{x_L x_R})Q_1 =$

$$T_1 = \begin{bmatrix} -1.0107 & -1.5310 & -1.2190 & 1.2455 \\ 7.8565 \times 10^{-3} & -1.0107 & -3.7428 \times 10^{-1} & 5.2642 \times 10^{-1} \\ 0 & 0 & 1.0000 & -5.2959 \times 10^{-1} \\ 0 & 0 & 0 & 3.0715 \times 10^{-1} \end{bmatrix}.$$

Note that with the rank-1 update on the matrix A, the single eigenvalue at the origin is moved to $\lambda = 1$ which then has no risk of being counted in $\mathbb{C}^-$. We then use the MATLAB function `ordschur.m` for ordering purposes with respect to a specific region, whether it be the right-half plane (rhp) or the left-half plane (lhp) (see [36]). In particular, we use the MATLAB command

```
[Q, T] = ordschur(Q1, T1 ,′ rhp′)
```

that generates $Q^{\mathrm{T}}AQ =$

$$\begin{bmatrix} -1.1102 \times 10^{-16} & -1.7964 \times 10^{-1} & 4.2371 \times 10^{-1} & -8.8190 \times 10^{-1} \\ -8.3267 \times 10^{-17} & 3.0715 \times 10^{-1} & 3.9948 \times 10^{-1} & -8.7693 \times 10^{-1} \\ 2.7756 \times 10^{-16} & 2.2204 \times 10^{-16} & -1.0107 & 1.4242 \\ -2.7756 \times 10^{-16} & -1.6653 \times 10^{-16} & -8.4459 \times 10^{-3} & -1.0107 \end{bmatrix}.$$

Note that this particular form of the MATLAB function `ordschur.m` was able to move the eigenvalues with positive real parts and the one at the origin to the north-west corner of the final Schur form, which is the decomposition (5) sought in this paper. Also note how the complex eigenvalue cases can be handled relatively easily with real arithmetic within the Schur decomposition framework.

One important special case arises when the matrix A has no eigenvalues in the open right-half plane and only the single eigenvalue at zero should be placed in the north-west corner of the reduced form. We do not need to find the Schur form of the matrix A in this case for solving the SDC problem for A with respect to the complex region $\mathcal{D} = \mathbb{C}^+$. For this purpose, let x_R be a right null vector of A. Let e_1 be a column vector of zeros except for a one as its first entry. Then we define

$$u = x_R - \|x_R\|_2 e_1, \qquad Q = I - \frac{2uu^{\mathrm{T}}}{u^{\mathrm{T}}u}. \tag{6}$$

Note that Q is a Householder matrix and is orthogonal, symmetric, and moreover $Q^{\mathrm{T}}AQ = QAQ$ being in the form of (5) with a scalar zero substituting for $A_{\mathcal{D}\mathcal{D}}$ in (5) which is the desired reduced form; see [29] for details.

Let $x(t)$ be a vector function of the indeterminate variable $t \in (-\infty, +\infty)$. The one-sided Laplace transform of $x(t)$ is given by $x^*(s) = \int_{0-}^{+\infty} \mathrm{e}^{-ts} x(t) \mathrm{d}t$. Note that we use the $*$ notation for Laplace transforms throughout this paper. The Dirac delta function $\delta(t)$ is a commonly used tool in engineering and it satisfies $\int_{-\epsilon}^{+\epsilon} \delta(t) = 1, \forall \epsilon > 0$. We can also use $\int_{0-}^{0+} \delta(t) = 1$ to refer to the same identity. We note that existence of Dirac delta functions in probability density functions is indicative of a probability mass at the origin. The degree of a polynomial $n^*(s)$ in the indeterminate s is denoted by $\deg(n^*)$. A transform is said to be rational if $x^*(s) = \frac{n^*(s)}{d^*(s)}$, for some polynomials $n^*(s)$ and $d^*(s)$. The rational transform $x^*(s)$ is strictly proper if $\deg(n^*) < \deg(d^*)$ and is proper if $\deg(n^*) = \deg(d^*)$. The latter case implies a constant term in the transform and is indicative of a Dirac delta function in the original domain. The poles of the rational function $x^*(s)$ are the roots of the denominator polynomial $d^*(s)$. Any strictly proper rational function $x^*(s)$ can additively be decomposed as $x^*(s) = x^*_-(s) + x^*_+(s)$, where the poles of $x^*_-(s)$ and $x^*_+(s)$ reside in $\mathbb{C}^-$ and $\mathbb{C}^+$, respectively. Moreover, this decomposition is unique. Throughout this paper, a rational transform $x^*(s)$ is said to be stable if all its poles lie in the open right-half plane ($\mathbb{C}^-$) and is anti-stable if all its poles lie in $\mathbb{C}^+$.

A linear time-invariant dynamical system with p inputs and q outputs is represented by the following set of ordinary differential equations for $t \geq 0$ [37]:

$$\frac{\mathrm{d}}{\mathrm{d}t}x(t) = x(t)T + u(t)V, \tag{7}$$

$$y(t) = x(t)H + u(t)D, \tag{8}$$

where $u(t) = \left(u_1(t), \dots, u_p(t)\right)$ and $y(t) = \left(y_1(t), \dots, y_q(t)\right)$ denote the input and output vectors, respectively, and $x(t) = (x_1(t), \dots, x_m(t))$ is called the state vector and its components are called the state variables, or simply the states. The matrices V, T, H, and D in the Eqs. (7) and (8) are real matrices of suitable sizes. Considering zero initial state, the transfer matrix $G^*(s)$ between the input and output vectors is written as [37]:

$$y^*(s) = u^*(s)G^*(s) = u^*(s)\left(V(sI - T)^{-1}H + D\right), \tag{9}$$

where $u^*(s)$ and $y^*(s)$ are the Laplace transforms of the input and output vectors, respectively. The equations of the form (7) and (8) are said to constitute a state-space representation or realization of the given linear time-invariant system with transfer matrix $G^*(s) = \{g_{ij}(s)\}$, $1 \leq i \leq p$, $1 \leq j \leq q$ if (9) holds [37]. The number of states (i.e., m) is referred to as the order of the state-space representation. This representation is said to be minimal if one cannot satisfy the identity (9) with a smaller order. Using similarity transformations, one can obtain infinitely many representations, whereas realization theory deals with finding state-space descriptions of linear systems and the properties of these descriptions [37]. Note that the poles of $g^*_{ij}(s)$, $1 \leq i \leq p$, $1 \leq j \leq q$ are contained in the eigenvalues of T [37]. In other words, if $g^*_{ij}(s) = \frac{n^*_{ij}(s)}{d^*_{ij}(s)}$, $1 \leq i \leq p$, $1 \leq j \leq q$ for polynomials $n^*_{ij}(s)$ and $d^*_{ij}(s)$, then $d^*_{ij}(s)$ is a factor of $\det(sI - T)$ [37]. It is also clear that $d^*_{ij}(-s)$ is a factor of $\det(sI + T)$.

3. Processes with matrix-exponential structure

We first define a random variable of matrix-exponential (ME) type. A non-negative random variable X is of ME type if its distribution function is of the form

$$\begin{cases} d, & \text{if } t = 0 \\ 1 + v\mathrm{e}^{tT}u, & \text{if } t > 0 \end{cases}$$

where, for $m \geq 1$, v is a $1 \times m$ row vector, T is a $m \times m$ matrix, and u is a $m \times 1$ column vector, all with real entries. The parameter d is the probability mass at zero. Note that $d = 1 + vu$. The corresponding density function $g(t)$ is then of the form

$$g(t) = v\mathrm{e}^{tT}h + \mathrm{d}\delta(t),$$

where $h = Tu$. An ME distribution is characterized with the quadruple (v, T, h, d) although d can be derived from the others. The moments of the ME-type random variable X are easily written as

$$E[X^i] = (-1)^{i+1}i!vT^{-(i+1)}h, \quad i > 0. \tag{10}$$

An ME-type renewal process is one with ME-type inter-renewal times. Next, we extend this definition to a Markov renewal process whose characterization is given in [8]. We define, for each $k \in \mathbb{N}$, a random variable X_k taking values in a finite set $E = \{1, 2, \dots, n\}$ and a random variable T_k taking values in $\mathbb{R}_+ = [0, \infty)$ such that $0 = T_0 \leq T_1 \leq T_2 \leq \cdots$. The stochastic process $(X, T) = \{X_k, T_k; k \in \mathbb{N}\}$ is said to be a Markov renewal process (MRP) with state space E provided that

$$P\{X_{k+1} = j, T_{k+1} - T_k \leq t | X_0, \dots, X_k; T_0, \dots, T_k\} = P\{X_{k+1} = j, T_{k+1} - T_k \leq t | X_k\},$$

for all $k \in \mathbb{N}$, $1 \leq j \leq n$, and $t \in \mathbb{R}_+$. The marginal process X_k of the MRP is called the modulating chain and $i \in E$ is called a state. On the other hand, the other marginal process Δ_k, $k \in \mathbb{N}$ defined by $\Delta_k = T_{k+1} - T_k$ is called the modulated process. In this study, we focus our attention to the time-homogeneous case for which the probability

$$F_{ij}(t) = P\{X_{k+1} = j, \Delta_k \leq t | X_k = i\}$$

is independent of the customer number k. The matrix $F(t) = \{F_{ij}(t)\}$ is then called the semi-Markov kernel of the MRP. For each pair $1 \leq i, j \leq n$, the function $F_{ij}(t)$ has all the properties of a distribution function except that the quantities defined by $F_{ij} = \lim_{t\to\infty} F_{ij}(t)$ are not necessarily one, but instead satisfy

$$F_{ij} \geq 0, \quad \sum_{j=1}^{n} F_{ij} = 1.$$

We note that $F_{ij} = P\{X_{n+1} = j | X_n = i)$ is the state transition probability from state i to j and we assume $F = \{F_{ij}\}$ is irreducible. Let π be the stationary solution of this discrete-time Markov chain (DTMC) such that

$$\pi F = \pi, \qquad \pi e = 1. \tag{11}$$

We also note that the quantity

$$F_{ij}(t)/F_{ij} = P\{\Delta_k \le t | X_{k+1} = j, X_k = i\} \tag{12}$$

is the sojourn time distribution in state i when the next state is j. A rich sub-case of the MRP is when the kernel $F(t)$ takes a matrix-exponential form, i.e.,

$$F(t) = Ve^{tT}U + F, \quad t \ge 0, \tag{13}$$

and equals zero elsewhere. Here, T is square and of size m and all its eigenvalues have negative real parts. Moreover, V and U are $n \times m$ and $m \times n$, respectively. We call an MRP with a matrix-exponential kernel an MRP-ME. We also define a kernel density (matrix) $G(t), t \ge 0$ by differentiating $F(t)$ with respect to t:

$$G(t) = \frac{\mathrm{d}}{\mathrm{d}t}F(t) = Ve^{tT}TU + (F + V\ U)\delta(t), \tag{14}$$

$$= Ve^{tT}H + D\delta(t), \quad t \ge 0 \tag{15}$$

where $\delta(t)$ is the Dirac delta function and $H = TU$ and $D = F + VU$. We also define the Laplace transform $G^*(s)$ of the kernel density matrix:

$$G^*(s) = \int_{0^-}^{\infty} e^{-ts}G(t)\mathrm{d}t = V(sI - T)^{-1}H + D. \tag{16}$$

An MRP-ME is then characterized by the quadruple (V, T, H, D). In general, one uses the sojourn times of the MRP, i.e., Δ_k, $k \in \mathbb{N}$, to model interarrival or service times in queueing systems. We note that the moments of the sojourn times satisfy

$$E[\Delta_k^i] = (-1)^{i+1} i! \pi V T^{-(i+1)} He, \quad i > 0. \tag{17}$$

It is clear that a phase-type process is an ME-type renewal process, which is clearly an MRP-ME with one state but with multiple modes (or phases). On the other hand, the well-known Markovian arrival process (MAP) is characterized with two square matrices D_0 and D_1 with D_0 having negative diagonal elements and non-negative off-diagonal elements, D_1 being non-negative, and $D = D_0 + D_1$ being an irreducible infinitesimal generator [13]. It is not difficult to show that this MAP is an MRP-ME with a kernel $F(t) = (e^{D_0 t} - I)D_0^{-1}D_1$ and is therefore characterized by the quadruple $(I, D_0, D_1, 0)$ [2]. If the above model is used to describe a service process, then we refer to that as a Markovian service process (MSP). The rational arrival process (RAP) introduced in [25] can again be viewed as an MRP-ME characterized by the quadruple $(I, D_0, D_1, 0)$ similar to a MAP but the matrices D_0 and D_1 do not necessarily possess the probabilistic interpretation available for MAPs.

With an MRP-ME, it is also possible to model point processes with batch arrivals. For example, the quadruple $(p, -\lambda, \lambda, (1 - p))$ provides an MRP-ME characterization of a batch Poisson process with batch arrival rate λ and geometrically distributed batch sizes with mean $1/p$. Other batch size distributions are also possible to characterize with MRP-MEs. Consider a batch Poisson process with batch arrival rate λ and batch size S associated with a finite support probability generating function $h_S^*(z) = \sum_{i=1}^N h_i z^i$ for some integer $N > 0$. It is not difficult to show that the kernel density matrix is of the form

$$G(t) = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} e^{-\lambda t} \begin{bmatrix} \lambda h_1 & \lambda \bar{h}_2 & 0 & \cdots & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & \cdots & 0 \\ \dfrac{h_2}{\bar{h}_2} & 0 & \dfrac{\bar{h}_3}{\bar{h}_2} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \dfrac{h_{N-1}}{\bar{h}_{N-1}} & 0 & \cdots & \cdots & 0 & \dfrac{\bar{h}_N}{\bar{h}_{N-1}} \\ 1 & 0 & \cdots & \cdots & 0 & 0 \end{bmatrix} \delta(t),$$

where $\bar{h}_k = \sum_{i=k}^N h_i$, $2 \le k \le N$.

The Batch Markovian Arrival Process (BMAP) is a generalization of the MAP which allows batch arrivals [38]. One can also use MRP-MEs to characterize BMAPs. For example, a BMAP with finite batch sizes is characterized with square matrices D_i, $0 \le i \le N$ with D_0 having negative diagonal elements and non-negative off-diagonal elements, D_i, $0 \le i \le N$ being non-negative, and $D = \sum_{i=0}^N D_i$ being an irreducible infinitesimal generator [38]. A Batch Markov Modulated Poisson Process (BMMPP) is a BMAP with diagonal D_i, $i \ge 1$ [39]. We further assume that D_N is invertible. As an extension of the batch Poisson process, one can show that a BMMPP is an MRP-ME with a kernel density matrix

$$G(t) = \begin{bmatrix} I \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} e^{D_0 t} \begin{bmatrix} D_1 & \bar{D}_2 & 0 & \cdots & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & \cdots & 0 \\ D_2\bar{D}_2^{-1} & 0 & \bar{D}_3\bar{D}_2^{-1} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ D_{N-1}\bar{D}_{N-1}^{-1} & 0 & \cdots & \cdots & 0 & \bar{D}_N\bar{D}_{N-1}^{-1} \\ I & 0 & \cdots & \cdots & 0 & 0 \end{bmatrix} \delta(t),$$

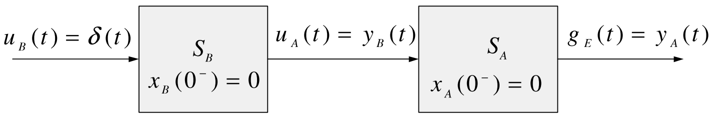

Fig. 1. Cascade interconnection diagram of the two linear systems of differential equations S_A and S_B to characterize $E = A + B$.

where $\bar{D}_k = \sum_{i=k}^{N} D_i$, $2 \le k \le N$. Note that $\bar{D}_k$, $2 \le k \le N$ is invertible due to the invertibility of D_N. More general BMAPs and their MRP-ME representations are left for future work.

4. State-space algebra for ME-type random variables

In this section, we will introduce a state-space algebra for ME-type random variables. Let the random variable A be of ME-type characterized with the quadruple (v_A, T_A, h_A, d_A). We then define a linear time-invariant system S_A for $t \ge 0$ with the following state-space representation

$$\begin{aligned} S_A : \frac{\mathrm{d}}{\mathrm{d}t} x_A(t) &= x_A(t) T_A + u_A(t) v_A, \\ y_A(t) &= x_A(t) h_A + u_A(t) d_A, \end{aligned}$$

where $x_A(t)$, $u_A(t)$, and $y_A(t)$ are the state, input, and the output of the system S_A. It is clear that the density of the random variable A (denoted by $g_A(t)$) can be viewed as the output of S_A when the input $u_A(t) = \delta(t)$ and the initial value $x_A(0^-) = 0$. Let B be another ME-type random variable characterized with the quadruple (v_B, T_B, h_B, d_B) and let A and B be independent. We can now define another linear time-invariant system S_B with the following state-space representation

$$\begin{aligned} S_B : \frac{\mathrm{d}}{\mathrm{d}t} x_B(t) &= x_B(t) T_B + u_B(t) v_B, \\ y_B(t) &= x_B(t) h_B + u_B(t) d_B, \end{aligned}$$

where $x_B(t)$, $u_B(t)$, and $y_B(t)$ being the state, input, and the output of the system S_B. We are now interested in the characterization of $E = A + B$ which should involve cascading the two systems S_A and S_B such that the output $y_B(t)$ of the system S_B is connected to the input $u_A(t)$ of the system S_A and $u_B(t) = \delta(t)$. While doing so, we employ the initial conditions $x_A(0^-) = 0$ and $x_B(0^-) = 0$. This situation is depicted in Fig. 1 in which the density of E (denoted by $g_E(t)$) should be taken as the output of the cascaded system. Rewriting the differential equations for $t \ge 0$ for the cascaded system, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} x_A(t) & x_B(t) \end{bmatrix} = \begin{bmatrix} x_A(t) & x_B(t) \end{bmatrix} \underbrace{\begin{bmatrix} T_A & 0 \\ h_B v_A & T_B \end{bmatrix}}_{T_E} + \underbrace{\begin{bmatrix} d_B v_A & v_B \end{bmatrix}}_{v_E} \delta(t),$$

$$g_E(t) = \begin{bmatrix} x_A(t) & x_B(t) \end{bmatrix} \underbrace{\begin{bmatrix} h_A \\ h_B d_A \end{bmatrix}}_{h_E} + \delta(t) \underbrace{d_A d_B}_{d_E},$$

from which we conclude that the distribution of E is of ME-type characterized with the quadruple (v_E, T_E, h_E, d_E).

Now, we are interested in the characterization of a random variable C defined through $C = (B - A)^+$. Note that the $+$ operator is key to analyzing queueing systems. We will use state-space algebra to show that the distribution of C is of ME-type and we will present a method to find the characterizing quadruple for C. For this purpose, we first need the following theorem.

Theorem 1. *Let X and Y be two independent non-negative random variables with ME-type distributions and their densities have rational Laplace transforms*

$$g_X^*(s) = d_X + \frac{n_X^*(s)}{d_X^*(s)}, \quad \deg(n_X^*) < \deg(d_X^*) \tag{18}$$

and

$$g_Y^*(s) = d_Y + \frac{n_Y^*(s)}{d_Y^*(s)}, \quad \deg(n_Y^*) < \deg(d_Y^*) \tag{19}$$

respectively. Then there exists a polynomial $u^*(s)$ of degree $\deg(d_Y^*)$ with $u^*(0) = 0$ so that the random variable defined by $Z = (X - Y)^+$ has an ME-type density with Laplace transform $g_Z^*(s)$ of the form

$$g_Z^*(s) = g_X^*(s) g_Y^*(-s) - \frac{u^*(s)}{d_Y^*(-s)}. \tag{20}$$

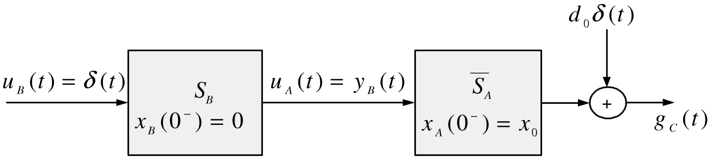

Fig. 2. Cascade interconnection diagram of the two linear systems of differential equations $\bar{S}_A$ and S_B to characterize $C = (B - A)^+$.

Conversely, if one can find a polynomial $u^*(s)$ with degree $\deg(d_Y^*)$ satisfying $u^*(0) = 0$ and the right-hand side of (20) having all its poles in the open left-half plane, then the identity (20) holds and gives the expression for the Laplace transform of the density of the random variable Z.

Proof. Consider double-sided Laplace transforms [40] and recall that right-sided densities $g(t)$, i.e., $g(t) = 0, t < 0$ possess stable Laplace transforms, i.e., their poles are in the open left-half plane. On the other hand, left-sided densities $g(t)$, i.e., $g(t) = 0, t > 0$, have anti-stable Laplace transforms, i.e., their poles are in the open right-half plane. Note that the density of the random variable $Z_1 = X - Y$ denoted by $g_{Z_1}(t)$ is double-sided and it has the two-sided Laplace transform

$$g_{Z_1}^*(s) = g_X^*(s) g_Y^*(-s) \tag{21}$$

but the strictly proper part of g_{Z_1} can be decomposed into its stable and anti-stable components in the following unique way:

$$g_{Z_1}^*(s) = d_X d_Y + \frac{u_1^*(s)}{d_X^*(s)} + \frac{u_2^*(s)}{d_Y^*(-s)}, \tag{22}$$

where $\deg(u_1^*) < \deg(d_X^*)$ and $\deg(u_2^*) < \deg(d_Y^*)$. Note that the + operator removes the left side of a double-sided density and places all the corresponding probability mass at $t = 0$. Therefore, in the transform domain

$$\begin{aligned} g_Z^*(s) &= d_X d_Y + \frac{u_1^*(s)}{d_X^*(s)} + \frac{u_2^*(0)}{d_Y^*(0)} \\ &= g_X^*(s) g_Y^*(-s) - \frac{u_2^*(s)}{d_Y^*(-s)} + \frac{u_2^*(0)}{d_Y^*(0)} \\ &= g_X^*(s) g_Y^*(-s) - \frac{u^*(s)}{d_Y^*(-s)} \end{aligned} \tag{23}$$

where

$$u^*(s) = u_2^*(s) - \frac{u_2^*(0) d_Y^*(-s)}{d_Y^*(0)}. \tag{24}$$

Evaluating the identity (23) at $s = 0$ and noting that $d_Y^*(0)$ cannot be zero, we show that $u^*(0) = 0$ and this concludes the if part of the proof. The only if part can be proved by observing the unique spectral decomposition of a rational function into its stable and anti-stable parts and tracing back the proof of the if part. □

Let us go back to our problem of characterizing the random variable $C = (B - A)^+$. For this purpose, we now define a linear time-invariant system $\bar{S}_A$ for $t \geq 0$ with the following state-space representation

$$\begin{aligned} \bar{S}_A : \frac{\mathrm{d}}{\mathrm{d}t} x_A(t) &= -x_A(t) T_A + u_A(t) v_A, \\ y_A(t) &= -x_A(t) h_A + u_A(t) d_A. \end{aligned}$$

Consider the depicted cascade connection diagram in Fig. 2 where $x_A(0^-) = x_0$ and we define $g_C(t) = y_A(t) + d_0\delta(t)$. If the initial condition x_0 and the external input parameter d_0 are chosen such that $g_C^*(s)$ becomes stable and $g_C^*(0) = 1$, then $g_C^*(s)$ gives the Laplace transform of the density of the random variable C. It is not difficult to see this by writing the transform $g_C^*(s)$ and showing that the associated expression is of the form (20).

Now, let us study the steady-state Lindley equation $W \overset{D}{=} (W + B - A)^+$ where the equality is to show that both sides have the same distribution. Also assume a solution exists to this equation. Consider the depicted feedback interconnection diagram in Fig. 3. If the initial condition x_0 and the external input parameter d_0 are chosen so that $g_W^*(s)$ is stable and $g_W^*(0) = 1$, then $g_W^*(s)$ gives the Laplace transform of the density of the random variable W. For a proof, see [41] which also provides a matrix-analytical procedure to find x_0 and d_0 and the ME characterization of the random variable W.

In all cases above, we have used state-space algebra to solve certain applied probability and queueing problems where the related random variables are ME-distributed. In the next section, we will use state-space algebra to study the waiting time of the semi-Markov queue when the interarrival and service times are MRP-ME distributed.

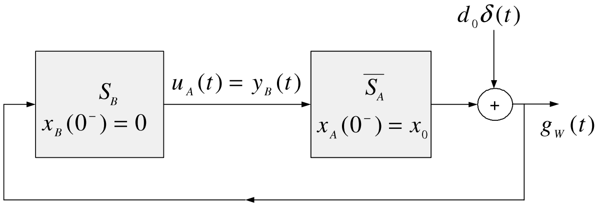

Fig. 3. Feedback interconnection diagram of the two linear systems of differential equations $\bar{S}_A$ and S_B to characterize the random variable W satisfying $W \overset{D}{=} (W + B - A)^+$.

5. The single-server semi-Markov queue

We study the semi-Markov queue governed by the Lindley recurrence given in (1), where B_k and A_k denote the service time of customer k and the interarrival times between customers k and $k+1$, respectively, and W_k denotes the kth customer's waiting time in the queue. We assume, in this study, that the individual processes A_k and B_k are both auto-correlated and we use the MRP-ME process to model auto-correlation in interarrival and service times. We use the sojourn times of an MRP-ME to model the processes A_k and B_k, i.e., (X_k^A, T_k^A) and (X_k^B, T_k^B) are the two Markov renewal processes with state spaces $E_A = \{1, 2, \dots, n_A\}$ and $E_B = \{1, 2, \dots, n_B\}$, respectively, describing the interarrival and service times. Let G_A and G_A^* denote the kernel density matrix and its Laplace transform, respectively, for the MRP-ME underlying the arrival process:

$$G_A(t) = V_A e^{tT_A} H_A + D_A\delta(t), \qquad G_A^*(s) = V_A(sI - T_A)^{-1} H_A + D_A. \tag{25}$$

We note that the MRP-ME process has n_A states and the matrix T_A is square of size m_A. Let $F_A = -V_A T_A^{-1} H_A + D_A$ which is a DTMC with $F_A e = e$. Let π_A be the steady-state vector of the modulating process X_k^A so that π_A satisfies

$$\pi_A F_A = \pi_A, \qquad \pi_A e = 1. \tag{26}$$

Similarly, let G_B and G_B^* denote the kernel density matrix and its Laplace transform, respectively, for the MRP-ME that models the service times:

$$G_B(t) = V_B e^{tT_B} H_B + D_B\delta(t), \qquad G_B^*(s) = V_B(sI - T_B)^{-1} H_B + D_B. \tag{27}$$

We assume the service MRP-ME process has n_B states and the matrix T_B is square of size m_B. Let $F_B = -V_B T_B^{-1} H_B + D_B$ which is also a DTMC with $F_B e = e$. Also let π_B be the steady-state vector of the modulating process X_k^B so that π_B satisfies

$$\pi_B F_B = \pi_B, \qquad \pi_B e = 1. \tag{28}$$

We also assume that these two state-space representations are irreducible. The queue described by the evolution equation (1) with the MRP-ME interarrival and service times described above is referred to as the semi-Markov queue. Using (17), we can write the mean interarrival time $E[A_k] = E[A] = \pi_A V_A T_A^{-2} H_A e$ and the mean service time $E[B_k] = E[B] = \pi_B V_B T_B^{-2} H_B e$. We assume, throughout this paper, that the load ρ defined $\rho = E[B]/E[A]$ is strictly less than one. Therefore, $W_k \to W$ as $k \to \infty$ in distribution, where W is called the steady-state waiting time with density $g_W(t)$. The Laplace transform of $g_W(t)$ is denoted by $g_W^*(s)$. In this paper, our goal is to calculate $g_W(t)$, $t \geq 0$.

We're now ready to study the steady-state solution of Lindley's equation (1). For this purpose, we define for $i = 1, \dots, n_A$ and $j = 1, \dots, n_B$:

$$\begin{aligned} G_{W,ij}(t) &= \frac{d}{dt} \lim_{k\to\infty} P\{W_k \leq t, X_k^A = i, X_k^B = j\}, \\ &= \frac{d}{dt} P\{W \leq t, X^A = i, X^B = j\}, \end{aligned} \tag{29}$$

and

$$\tilde{g}_W(t) = \mathrm{vec}(\{G_{W,ij}(t)\}). \tag{30}$$

In our analysis, the following Laplace transforms are crucial:

$$G_{W,ij}^*(s) = \int_{0^-}^{\infty} e^{-st} G_{W,ij}(t) dt, \qquad \tilde{g}_W^*(s) = \mathrm{vec}(\{G_{W,ij}^*(s)\}). \tag{31}$$

From Lindley's equation (1) and Theorem 1, we note the existence of polynomials $u_{klij}^*(s)$, $k, i = 1, \dots, n_A$, $l, j = 1, \dots, n_B$ with $u_{klij}^*(0) = 0$ such that the following hold:

$$G_{W,ij}^*(s) = \sum_{k=1}^{n_A} \sum_{l=1}^{n_B} G_{W,kl}^*(s) G_{A,ki}^*(-s) G_{B,lj}^*(s) - \sum_{k=1}^{n_A} \sum_{l=1}^{n_B} \frac{u_{klij}^*(s)}{d_{A,ki}^*(-s)}, \tag{32}$$

where $d^*_{A,ki}(-s)$ is the denominator of $G^*_{A,ki}(-s)$ and its degree is the same as that of $u^*_{klij}(s)$. This identity can be shown to reduce to the existence of polynomials $u^*_{ij}(s)$ with degree m_A and $u^*_{ij}(0) = 0$ such that the second term on the right-hand side of (32) can be simplified as in

$$G^*_{W,ij}(s) = \sum_{k=1}^{n_A} \sum_{l=1}^{n_B} G^*_{W,kl}(s) G^*_{A,ki}(-s) G^*_{B,lj}(s) - \frac{u^*_{ij}(s)}{\det(sI + T_A)}, \tag{33}$$

since $d^*_{A,ki}(-s)$ is a factor of $\det(sI+T_A)$ for all k, i. Conversely, if one can find polynomials $u^*_{ij}(s)$ with degree m_A and $u^*_{ij}(0) = 0$ such that (33) holds where the right-hand side of (33) is free of the closed right-half plane poles, i.e., all poles in the open left-half plane, then the identity (33) completely describes $G^*_{W,ij}(s)$ which is what we want to find. The identity (33) can also be put into the following vector form:

$$\tilde{g}^*_W(s) = \tilde{g}^*_W(s) \left(I_{n_A} \otimes G^*_B(s)\right) \left(G^*_A(-s) \otimes I_{n_B}\right) - \frac{\tilde{u}^*(s)}{\det(sI + T_A)}, \tag{34}$$

where $\tilde{u}^*(s) = \mathrm{vec}\left(\{u^*_{ij}(s)\}\right)$. Our goal is then to find $\tilde{u}^*(s)$ with $\tilde{u}^*(0) = 0$ such that identity (34) holds for a stable transform vector $\tilde{g}^*_W(s)$. However, the identity (34) does not directly lend itself to a computational procedure for finding the unknown polynomials and, even so, doing the calculations in the transform domain is cumbersome and ill-conditioned, and additionally one needs to perform transform inversion in the end to calculate $\tilde{g}_W(t)$. In order to avoid transform domain calculations for finding $\tilde{u}^*(s)$, we first introduce the following linear system of differential equations defined for $t \geq 0$ (denoted by $\bar{S}^e_A$) associated with the interarrival times in state-space form but with non-zero initial states for $t \geq 0$:

$$\bar{S}^e_A : \frac{\mathrm{d}}{\mathrm{d}t} x_A(t) = -x_A(t)\tilde{T}_A + u_A(t)\tilde{V}_A, \qquad x_A(0^-) = x_0, \tag{35}$$

$$y_A(t) = -x_A(t)\tilde{H}_A + u_A(t)\tilde{D}_A, \tag{36}$$

where

$$\tilde{V}_A = V_A \otimes I_{n_B}, \qquad \tilde{T}_A = T_A \otimes I_{n_B}, \qquad \tilde{H}_A = H_A \otimes I_{n_B}, \qquad \tilde{D}_A = D_A \otimes I_{n_B}. \tag{37}$$

In the linear system above, $x_A(t)$ is the system state with a non-zero initial value x_0 which is of size $1 \times m_A n_B$. Next, consider another linear system of differential equations (denoted by S^e_B) associated with the service times in state-space form for $t \geq 0$:

$$S^e_B : \frac{\mathrm{d}}{\mathrm{d}t} x_B(t) = x_B(t)\tilde{T}_B + u_B(t)\tilde{V}_B, \qquad x_B(0^-) = 0, \tag{38}$$

$$y_B(t) = x_B(t)\tilde{H}_B + u_B(t)\tilde{D}_B, \tag{39}$$

where

$$\tilde{V}_B = I_{n_A} \otimes V_B, \qquad \tilde{T}_B = I_{n_A} \otimes T_B, \qquad \tilde{H}_B = I_{n_A} \otimes H_B, \qquad \tilde{D}_B = I_{n_A} \otimes D_B. \tag{40}$$

In the linear system above, $x_B(t)$ is the system state with a zero initial value $x_B(0^-) = 0$. We interconnect these two systems, for reasons to be made clear later, through the following feedback configuration:

$$u_A(t) = y_B(t) =: u_F(t), \qquad u_B(t) = y_A(t) + d_0\delta(t) =: y_F(t), \tag{41}$$

where the subscript F is used to refer to the feedback configuration and d_0 is an unknown vector of size n_A. This situation corresponds to the same feedback interconnection diagram in Fig. 3, but $\bar{S}_A$ and S_B are replaced with $\bar{S}^e_A$ and S^e_B, respectively. The system parameters x_0 and d_0 are not known yet but they are to be determined. We now obtain an expression for the Laplace transform $y^*_F(s)$ of the output vector $y_F(t)$. Note from (36) and (41) that

$$y^*_F(s) = -x^*_A(s)\tilde{H}_A + u^*_F(s)\tilde{D}_A + d_0.$$

Also note from (35) and the derivative rule for Laplace transforms that

$$x^*_A(s) = \left(u^*_F(s)\tilde{V}_A + x_0\right)(sI + \tilde{T}_A)^{-1}.$$

Consequently,

$$\begin{aligned} y^*_F(s) &= -\left(u^*_F(s)\tilde{V}_A + x_0\right)(sI + \tilde{T}_A)^{-1}\tilde{H}_A + u^*_F(s)\tilde{D}_A + d_0, \\ &= u^*_F(s)\left(-\tilde{V}_A(sI + \tilde{T}_A)^{-1}\tilde{H}_A + \tilde{D}_A\right) - \left(x_0(sI + \tilde{T}_A)^{-1}\tilde{H}_A - d_0\right). \end{aligned} \tag{42}$$

From (38) and (39), we know that

$$u^*_F(s) = y^*_F(s)\left(\tilde{V}_B(sI - \tilde{T}_B)^{-1}\tilde{H}_B + \tilde{D}_B\right),$$

from which (42) simplifies to

$$y_F^*(s) = y_F^*(s)\left(I_{n_A} \otimes G_B^*(s)\right)\left(G_A^*(-s) \otimes I_{n_B}\right) - \left(x_0(sI + \tilde{T}_A)^{-1}\tilde{H}_A - d_0\right). \tag{43}$$

Note the similarity in the identities (43) and (34) with. If we obtain x_0 and d_0 such that the following two conditions are satisfied:

$$y_F^*(s) \quad \text{is stable}, \tag{44}$$

$$y_F^*(0) = \tilde{\pi} = \pi_A \otimes \pi_B, \tag{45}$$

then (43) provides an equivalent expression to (33) and (34) with the choice of $\tilde{u}^*(s) = x_0 \operatorname{adj}(sI + \tilde{T}_A)\tilde{H}_A - d_0 \det(sI + \tilde{T}_A)$. Therefore, in case one has x_0 and d_0 such that the output of the feedback system $y_F^*(s)$ satisfies the two conditions given above, then it is true that

$$y_F^*(s) = \tilde{g}_W^*(s). \tag{46}$$

The next section describes a matrix-analytical method to find $\tilde{g}_W(t)$ without using any transform domain calculations, but instead uses the transform identities (33) or (43) only for validating the proposed algorithmic method.

6. Matrix-analytical method for the semi-Markov queue

In this section, we present a matrix-analytical method for solving for the steady-state waiting time distribution through the calculation of the vector $\tilde{g}_W(t)$. For this purpose, we obtain a state-space representation for the feedback configuration F given through (41). We first write the vector input $u_F(t)$ in terms of the individual states $x_A(t)$ and $x_B(t)$ of the systems $\bar{S}_A^e$ and S_B^e, respectively:

$$\begin{aligned} u_F(t) &= x_B(t)\tilde{H}_B + u_B(t)\tilde{D}_B, \\ &= x_B(t)\tilde{H}_B + \left(-x_A(t)\tilde{H}_A + u_A(t)\tilde{D}_A + d_0\delta(t)\right)\tilde{D}_B, \\ &= \left(-x_A(t)\tilde{H}_A\tilde{D}_B + x_B(t)\tilde{H}_B + d_0\tilde{D}_B\delta(t)\right)\tilde{D}_{AB}, \end{aligned} \tag{47}$$

where

$$\tilde{D}_{AB} = (I - \tilde{D}_A\tilde{D}_B)^{-1}. \tag{48}$$

Similarly, the output vector $y_F(t)$ is written in terms of the individual states $x_A(t)$ and $x_B(t)$ as follows:

$$\begin{aligned} y_F(t) &= -x_A(t)\tilde{H}_A + u_A(t)\tilde{D}_A + d_0\delta(t), \\ &= -x_A(t)\tilde{H}_A + \left(x_B(t)\tilde{H}_B + u_B(t)\tilde{D}_B\right)\tilde{D}_A + d_0\delta(t) \\ &= \left(-x_A(t)\tilde{H}_A + x_B(t)\tilde{H}_B\tilde{D}_A + d_0\delta(t)\right)\tilde{D}_{BA}, \end{aligned} \tag{49}$$

where

$$\tilde{D}_{BA} = (I - \tilde{D}_B\tilde{D}_A)^{-1}. \tag{50}$$

By substituting (47) in the state equation (35), one can easily show that

$$\frac{\mathrm{d}}{\mathrm{d}t}x_A(t) = x_A(t)(-\tilde{T}_A - \tilde{H}_A\tilde{D}_B\tilde{D}_{AB}\tilde{V}_A) + x_B(t)\tilde{H}_B\tilde{D}_{AB}\tilde{V}_A + d_0\tilde{D}_B\tilde{D}_{AB}\tilde{V}_A\delta(t). \tag{51}$$

On the other hand, insertion of (49) in the state equation (38) leads to the following modified state equations for $x_B(t)$:

$$\frac{\mathrm{d}}{\mathrm{d}t}x_B(t) = -x_A(t)\tilde{H}_A\tilde{D}_{BA}\tilde{V}_B + x_B(t)(\tilde{T}_B + \tilde{H}_B\tilde{D}_A\tilde{D}_{BA}\tilde{V}_B) + d_0\tilde{D}_{BA}\tilde{V}_B\delta(t). \tag{52}$$

Combining the differential equations (51) and (52) we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}x_F(t) = \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x_A(t) & x_B(t) \end{bmatrix} = x_F(t)T_F + d_0V_F\delta(t), \tag{53}$$

with the initial value

$$x_F(0^-) = (x_0, 0), \tag{54}$$

and the matrices T_F and V_F are defined as

$$T_F = \begin{bmatrix} -\tilde{T}_A - \tilde{H}_A\tilde{D}_B\tilde{D}_{AB}\tilde{V}_A & -\tilde{H}_A\tilde{D}_{BA}\tilde{V}_B \\ \tilde{H}_B\tilde{D}_{AB}\tilde{V}_A & \tilde{T}_B + \tilde{H}_B\tilde{D}_A\tilde{D}_{BA}\tilde{V}_B \end{bmatrix} \tag{55}$$

and

$$V_F = \begin{bmatrix} \tilde{D}_B\tilde{D}_{AB}\tilde{V}_A & \tilde{D}_{BA}\tilde{V}_B \end{bmatrix}. \tag{56}$$

Integrating the state equation (53) from 0^- to 0^+ (or simply 0), we obtain the following state equation

$$\frac{\mathrm{d}}{\mathrm{d}t}x_F(t) = x_F(t)T_F, \qquad x_F(0) = (x_0, 0) + d_0 V_F, \tag{57}$$

and the output equation

$$y_F(t) = x_F(t)H_F + d_0\tilde{D}_{BA}\delta(t), \tag{58}$$

where

$$H_F = \begin{bmatrix} -\tilde{H}_A\tilde{D}_{BA} \\ \tilde{H}_B\tilde{D}_A\tilde{D}_{BA} \end{bmatrix}. \tag{59}$$

We note that the matrix T_F has $n_B m_A - 1$ eigenvalues with positive real parts, one eigenvalue at the origin, and $n_A m_B$ eigenvalues with negative real parts [42]. We now provide the following theorem regarding the left and right null vectors of the matrix T_F.

Theorem 2. *The left and right null vectors of the matrix T_F, denoted by x_L and x_R, respectively, are explicitly given as:*

$$x_L = \begin{bmatrix} -\tilde{\pi}\tilde{V}_A\tilde{T}_A^{-1} & \tilde{\pi}\tilde{V}_B\tilde{T}_B^{-1} \end{bmatrix}, \qquad x_R = \begin{bmatrix} \tilde{T}_A^{-1}\tilde{H}_A e \\ \tilde{T}_B^{-1}\tilde{H}_B e \end{bmatrix}. \tag{60}$$

Proof. Let $\tilde{F}_A = F_A \otimes I_{n_B}$ and $\tilde{F}_B = I_{n_A} \otimes F_B$. We note that

$$\begin{aligned} \tilde{\pi}\tilde{F}_A &= (\pi_A \otimes \pi_B)(F_A \otimes I), \\ &= (\pi_A F_A) \otimes \pi_B, \\ &= \pi_A \otimes \pi_B, \\ &= \tilde{\pi}. \end{aligned}$$

Similarly, one can show $\tilde{\pi}\tilde{F}_B = \tilde{\pi}$. Now, we first show that x_L is a left null vector of T_F. For this purpose, we suitably partition $T_F = (T_{F,1}, T_{F,2})$, where $T_{F,1}$ and $T_{F,2}$ have $n_B m_A$ and $n_A m_B$ columns, respectively. We then write

$$\begin{aligned} x_L T_{F,1} &= \tilde{\pi}\tilde{V}_A + \tilde{\pi}\underbrace{\tilde{V}_A\tilde{T}_A^{-1}\tilde{H}_A}_{\tilde{D}_A - \tilde{F}_A}\tilde{D}_B\tilde{D}_{AB}\tilde{V}_A + \tilde{\pi}\underbrace{\tilde{V}_B\tilde{T}_B^{-1}\tilde{H}_B}_{\tilde{D}_B - \tilde{F}_B}\tilde{D}_{AB}\tilde{V}_A, \\ &= \tilde{\pi}\tilde{V}_A + \tilde{\pi}\tilde{D}_A\tilde{D}_B\tilde{D}_{AB}\tilde{V}_A - \tilde{\pi}\tilde{D}_B\tilde{D}_{AB}\tilde{V}_A + \tilde{\pi}\tilde{D}_B\tilde{D}_{AB}\tilde{V}_A - \tilde{\pi}\tilde{D}_{AB}\tilde{V}_A, \\ &= \tilde{\pi}(I + (\tilde{D}_A\tilde{D}_B - I)\tilde{D}_{AB})\tilde{V}_A \\ &= 0. \end{aligned}$$

Above, the last step is based on the definition of $\tilde{D}_{AB}$ in (48). Similarly,

$$\begin{aligned} x_L T_{F,2} &= \tilde{\pi}\underbrace{\tilde{V}_A\tilde{T}_A^{-1}\tilde{H}_A}_{\tilde{D}_A - \tilde{F}_A}\tilde{D}_{BA}\tilde{V}_A + \tilde{\pi}\tilde{V}_B + \tilde{\pi}\underbrace{\tilde{V}_B\tilde{T}_B^{-1}\tilde{H}_B}_{\tilde{D}_B - \tilde{F}_B}\tilde{D}_A\tilde{D}_{BA}\tilde{V}_B, \\ &= \tilde{\pi}\tilde{V}_B + \tilde{\pi}\tilde{D}_A\tilde{D}_{BA}\tilde{V}_B - \tilde{\pi}\tilde{D}_{BA}\tilde{V}_B + \tilde{\pi}\tilde{D}_B\tilde{D}_A\tilde{D}_{BA}\tilde{V}_B - \tilde{\pi}\tilde{D}_A\tilde{D}_{BA}\tilde{V}_B \\ &= \tilde{\pi}(I + (\tilde{D}_B\tilde{D}_A - I)\tilde{D}_{BA})\tilde{V}_B \\ &= 0. \end{aligned}$$

We therefore conclude that $x_L T_F = 0$, i.e., x_L is a left null vector of T_F. Next, we show that x_R is a right null vector of T_F. First note that

$$\begin{aligned} \tilde{F}_A e &= (F_A \otimes I)(e \otimes e), \\ &= (F_A e) \otimes e, \\ &= e \otimes e, \\ &= e. \end{aligned}$$

Similarly, one can show $\tilde{F}_B e = e$. We then suitably partition $T_F = (T_{F,a}^{\mathrm{T}}, T_{F,b}^{\mathrm{T}})^{\mathrm{T}}$ where $T_{F,a}$ and $T_{F,b}$ have $n_B m_A$ and $n_A m_B$ rows, respectively. We then observe

$$(I - \tilde{D}_B\tilde{D}_A)\tilde{D}_B = \tilde{D}_B(I - \tilde{D}_A\tilde{D}_B)$$

and pre-multiplying the above equality by $(I - \tilde{D}_B\tilde{D}_A)^{-1}$ and post-multiplying it by $(I - \tilde{D}_A\tilde{D}_B)^{-1}$, we obtain

$$\tilde{D}_B\tilde{D}_{AB} = \tilde{D}_{BA}\tilde{D}_B.$$

Similarly, one can show

$$\tilde{D}_{AB}\tilde{D}_A = \tilde{D}_A\tilde{D}_{BA}.$$

Using above equalities, we write

$$\begin{aligned}
T_{F,a}x_R &= -\tilde{H}_A e - \tilde{H}_A\tilde{D}_B\tilde{D}_{AB}\underbrace{\tilde{V}_A\tilde{T}_A^{-1}\tilde{H}_A}_{\tilde{D}_A-\tilde{F}_A} e - \tilde{H}_A\tilde{D}_{BA}\underbrace{\tilde{V}_B\tilde{T}_B^{-1}\tilde{H}_B}_{\tilde{D}_B-\tilde{F}_B} e,\\
&= -\tilde{H}_A e - \tilde{H}_A\underbrace{\tilde{D}_B\tilde{D}_{AB}}_{\tilde{D}_{BA}\tilde{D}_B}\tilde{D}_A e + \tilde{H}_A\underbrace{\tilde{D}_B\tilde{D}_{AB}}_{\tilde{D}_{BA}\tilde{D}_B} e - \tilde{H}_A\tilde{D}_{BA}\tilde{D}_B e + \tilde{H}_A\tilde{D}_{BA} e,\\
&= -\tilde{H}_A(I + \tilde{D}_{BA}(\tilde{D}_B\tilde{D}_A - I))e,\\
&= 0.
\end{aligned}$$

Similarly,

$$\begin{aligned}
T_{F,b}x_R &= \tilde{H}_B e + \tilde{H}_B\tilde{D}_{AB}\underbrace{\tilde{V}_A\tilde{T}_A^{-1}\tilde{H}_A}_{\tilde{D}_A-\tilde{F}_A} e + \tilde{H}_B\tilde{D}_A\tilde{D}_{BA}\underbrace{\tilde{V}_B\tilde{T}_B^{-1}\tilde{H}_B}_{\tilde{D}_B-\tilde{F}_B} e,\\
&= \tilde{H}_B e - \tilde{H}_B\tilde{D}_{AB} e + \tilde{H}_B\tilde{D}_{AB}\tilde{D}_A e - \tilde{H}_B\tilde{D}_A\tilde{D}_{BA} e + \tilde{H}_B\tilde{D}_A\tilde{D}_{BA}\tilde{D}_B e,\\
&= \tilde{H}_B(I - \tilde{D}_{AB} + \underbrace{\tilde{D}_{AB}\tilde{D}_A}_{\tilde{D}_A\tilde{D}_{BA}} - \tilde{D}_A\tilde{D}_{BA} + \tilde{D}_A\underbrace{\tilde{D}_{BA}\tilde{D}_B}_{\tilde{D}_B\tilde{D}_{AB}})e,\\
&= \tilde{H}_B(I - (I - \tilde{D}_A\tilde{D}_B)\tilde{D}_{AB})e,\\
&= 0.
\end{aligned}$$

We therefore show that x_R is a right null vector of T_F, which concludes the proof. □

However, for computational efficiency, we should use the identity $(A \otimes B)^{-1} = (A^{-1} \otimes B^{-1})$ for calculating the left and right null vectors from (60). Therefore, the expressions in (60) can be simplified to

$$x_L = \begin{bmatrix} -\pi_A V_A T_A^{-1} \otimes \pi_B & \pi_A \otimes \pi_B V_B T_B^{-1} \end{bmatrix}, \qquad x_R = \begin{bmatrix} T_A^{-1} H_A e \otimes e \\ e \otimes T_B^{-1} H_B e \end{bmatrix}. \tag{61}$$

Now let Q_F be an orthogonal matrix such that

$$Q_F^{\mathrm{T}} T_F Q_F = \begin{bmatrix} T_{F,++} & T_{F,+-} \\ 0 & T_{F,--} \end{bmatrix}, \tag{62}$$

where the eigenvalues of the matrix $T_{F,++}$ and $T_{F,--}$ have non-negative and negative real parts, and they are of size $n_B m_A$ and $n_A m_B$, respectively. In our numerical studies, we use the ordered Schur form-based method described in Section 2 for obtaining the form (62). In the special case of $n_B m_A = 1$, there is only one eigenvalue of T_F with a non-negative real part, and we use the Householder method, also described in Section 2, and in particular the transformation given in (6). Once the decomposition (62) is obtained, we then define the following transformation

$$\tilde{x}_F(t) = x_F(t) Q_F, \tag{63}$$

and partition

$$\tilde{x}_F(t) = \left(\tilde{x}_{F,+}(t), \tilde{x}_{F,-}(t)\right) \tag{64}$$

appropriately so that the row vectors $\tilde{x}_{F,+}(t)$ and $\tilde{x}_{F,-}(t)$ are of size $n_B m_A$ and $n_A m_B$, respectively. From (57) and (62), we first obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{x}_{F,+}(t) = \tilde{x}_{F,+}(t) T_{F,++} \tag{65}$$

but since all the eigenvalues of $T_{F,++}$ have non-negative real parts, the only condition for the analyticity of $y_F^*(s)$ in the closed right-half plane is

$$\tilde{x}_{F,+}(0) = 0. \tag{66}$$

Partitioning Q_F as in (62)

$$Q_F = \begin{bmatrix} Q_{F,++} & Q_{F,+-} \\ Q_{F,-+} & Q_{F,--} \end{bmatrix}, \tag{67}$$

and defining

$$Q_{F,+} = \begin{bmatrix} Q_{F,++} \\ Q_{F,-+} \end{bmatrix}, \qquad Q_{F,-} = \begin{bmatrix} Q_{F,+-} \\ Q_{F,--} \end{bmatrix}, \tag{68}$$

the condition (66) can be reduced to a linear matrix equation in the unknowns x_0 and d_0:

$$\begin{aligned} \tilde{x}_{F,+}(0) &= x_F(0)Q_{F,+}, \\ &= ((x_0, 0) + d_0 V_F)\, Q_{F,+}, \\ &= x_0 Q_{F,++} + d_0 V_F Q_{F,+} = 0. \end{aligned} \tag{69}$$

The equation (69) ensures that the condition (44) is satisfied. In order to express the condition (45) as a linear matrix equation, we first write $\tilde{x}_F(t)$ as

$$\begin{aligned} \tilde{x}_{F,-}(t) &= \tilde{x}_{F,-}(0)\mathrm{e}^{tT_{F,--}} \\ &= \left(x_0 Q_{F,+-} + d_0 V_F Q_{F,-}\right) \mathrm{e}^{tT_{F,--}}. \end{aligned}$$

Finally, the expression for the output vector $y_F(t)$ in (58) can further be simplified to

$$y_F(t) = \left(x_0 Q_{F,+-} + d_0 V_F Q_{F,-}\right) \mathrm{e}^{t\, T_{F,--}} Q_{F,-}^{\mathrm{T}} H_F + d_0 \tilde{D}_{BA}\, \delta(t) \tag{70}$$

Noting that $\int_{0^-}^{\infty} y_F(t) = \tilde{\pi}$ for condition (45), we obtain another linear matrix equation in x_0 and d_0 so that condition (45) is satisfied:

$$\tilde{\pi} = -x_0 Q_{F,+-} T_{F,--}^{-1} Q_{F,-}^{\mathrm{T}} H_F + d_0 \left(-V_F Q_{F,-} T_{F,--}^{-1} Q_{F,-}^{\mathrm{T}} H_F + \tilde{D}_{BA}\right). \tag{71}$$

Combination of (69) and (71) give $(m_A + n_A)n_B$ equations with $(m_A + n_A)n_B$ unknowns. Recall that x_0 is of size $n_B m_A$ and d_0 is of size $n_A n_B$. Solving for x_0 and d_0 from (69) and (71) leads us to a matrix-exponential waiting time distribution:

$$y_F(t) = \tilde{g}_W(t) = v\mathrm{e}^{tT} H + d\delta(t) \tag{72}$$

where

$$v := x_0 Q_{F,+-} + d_0 V_F Q_{F,-} \tag{73}$$

$$T := T_{F,--} \tag{74}$$

$$H := Q_{F,-}^{\mathrm{T}} H_F \tag{75}$$

$$d := d_0 \tilde{D}_{BA}. \tag{76}$$

Note that the density of the steady-state waiting time is written as

$$g_W(t) = \tilde{g}_W(t)e = v\mathrm{e}^{tT} He + de\delta(t), \tag{77}$$

from which one can find the ith moment of the waiting time as follows:

$$E[W^i] = (-1)^{i+1} i! v T^{-(i+1)} He. \tag{78}$$

We provide the overall algorithm in Table 1. We emphasize that the proposed algorithm relies on only an ordered Schur decomposition of a matrix in addition to ordinary vector-matrix operations and therefore it is matrix-analytical and simple to implement using linear algebra software packages.

7. Modified Lindley recurrence

In this section, we study a modified Lindley recurrence relation:

$$S_{k+1} = (S_k - A_k)^+ + B_{k+1}, \quad k \geq 0, \tag{79}$$

where A_k and B_k are MRP-ME distributed and are defined as in Section 5 with the limiting distribution S. Note that the relation holds $S_k = W_k + B_k$ and W_k and S_k denote the workload just before, and respectively right after, the arrival instant of customer k [43]. In queueing systems of interest to the current paper, W corresponds to the steady-state waiting time

Table 1
Algorithm to find $g_W(t)$ given the pair of quadruples (V_A, T_A, H_A, D_A) and (V_B, T_B, H_B, D_B) characterizing the MRP-MEs for interarrival and service times, respectively.

1.	Find the steady-state vectors π_A and π_B using (26) and (28), respectively.
2.	Let $\tilde{\pi} = \pi_A \otimes \pi_B$.
3.	Obtain $\tilde{V}_A$, $\tilde{T}_A$, $\tilde{H}_A$, and $\tilde{D}_A$ by (37).
4.	Obtain $\tilde{V}_B$, $\tilde{T}_B$, $\tilde{H}_B$, and $\tilde{D}_B$ by (40).
5.	Construct $\tilde{D}_{AB}$ and $\tilde{D}_{BA}$ by (48) and (50).
6.	Construct the matrices T_F, V_F, and H_F by using (55), (56) and (59), respectively.
7.	Find an orthogonal matrix Q_F so as to obtain the ordered Schur form of T_F where ordering is done as in (62).
8.	Partition the matrix Q_F as in (67) and (68).
9.	Solve for x_0 and d_0 using the linear equations (69) and (71).
10.	Define v, T, H, and d by (73), (74), (75) and (76).
11.	Write $g_W(t) = ve^{tT}He + de\,\delta(t)$.

whereas S represents a customer's sojourn time [43]. Assume we want to find the distribution of S directly. It is not difficult to show for this case that if one can find x_0 and d_0 such that the following two conditions are satisfied

$$u_F^*(s) \quad \text{is stable,} \tag{80}$$

$$u_F^*(0) = \tilde{\pi} = \pi_A \otimes \pi_B, \tag{81}$$

then

$$u_F(t) = \tilde{g}_S(t),$$

as opposed to the two conditions (44), (45), and the identity (72) that we proved for the ordinary Lindley recurrence (1). Above, $\tilde{g}_S(t)$ is defined as in (30) with W replaced with S. Compared with the algorithmic approach of the previous section, the first linear equation (69) remains intact but the second linear equation (71) needs to be replaced with

$$\tilde{\pi} = -x_0 Q_{F,+-} T_{F,--}^{-1} Q_{F,-}^{\mathrm{T}} \bar{H}_F + d_0 \left(-V_F Q_{F,-} T_{F,--}^{-1} Q_{F,-}^{\mathrm{T}} \bar{H}_F + \tilde{D}_B \tilde{D}_{AB} \right), \tag{82}$$

where

$$\bar{H}_F = \begin{bmatrix} -\tilde{H}_A \tilde{D}_B \tilde{D}_{AB} \\ \tilde{H}_B \tilde{D}_{AB} \end{bmatrix}. \tag{83}$$

Solving these two Eqs. (69) and (82) for x_0 and d_0, it is easy to show that

$$g_S(t) = \tilde{g}_S(t)e = ve^{tT}\bar{H}e + \bar{d}e\delta(t), \tag{84}$$

where v and T are defined in (73) and (74), respectively, and

$$\bar{H} = Q_{F,-}^{\mathrm{T}} \bar{H}_F, \qquad \bar{d} = d_0 \tilde{D}_B \tilde{D}_{AB}.$$

Alternatively, one can find the distribution of W and then, respectively, that of S using the identity $S_k = W_k + B_k$ but a method is still required to obtain a minimal state-space representation for S_k. We believe that the method proposed above can be used for this purpose.

8. Numerical experiments

Numerical experimentation of a queuing system when the arrival and service processes have ME distributions is already carried out in [41]. In the current paper, we will study an $\mathrm{M}^X/\mathrm{ME}/1$ queue, a PH/MRP/1 queue, a MAP/PH/1 queue, and a priority queue involving auto-correlated service times, to validate the generality and effectiveness of the proposed approach. The implementations of the proposed algorithms are in MATLAB 7.0, which is run on an Intel Core2Duo T7200 Windows XP notebook with 2 GB of RAM.

8.1. Example 1: $M^X/ME/1$ queue

We study the batch $\mathrm{M}^X/\mathrm{D}/1$ queue example from [1]. The inter-batch times are exponential with parameter λ. The batch size S has a probability generating function denoted by $h_S^*(z)$ which equals $h_S^*(z) = \sum_{i=1}^{\infty} h_i z^i$. We fix $h_1 = 0.25$, $h_2 = 0.5$, $h_3 = 0.25$, and $h_i = 0, i > 3$ as in [1]. The service time is deterministic with rate μ and we fix μ to 1/5. The parameter ρ gives the utilization of the queue such that $\rho = E[S]\lambda/\mu = 10\lambda$. As described in Section 3, the arrival process A is an MRP-ME described by the quadruple

$$T_A = -\lambda, \qquad V_A = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \qquad H_A = \lambda \begin{bmatrix} h_1 & h_2 + h_3 & 0 \end{bmatrix}, \qquad D_A = \begin{bmatrix} 0 & 0 & 0 \\ \dfrac{h_2}{h_2 + h_3} & 0 & \dfrac{h_3}{h_2 + h_3} \\ 1 & 0 & 0 \end{bmatrix}.$$

Note that we have $n_A = 3$ states and $m_A = 1$ modes. Since the service time is deterministic, it is not of ME type, but we can approximate it by an Erlang-k distribution which is of phase type, or use an approximation to $g_B^*(s) = \mathrm{e}^{-s/\mu}$. The Reference [1] proposes an approximation based on continued fractions of varying orders; the particular one with order two is given as

$$g_B^*(s) \approx \frac{6 - 2s/\mu}{6 + 4s/\mu + s^2/\mu^2}. \tag{85}$$

In case we employ this particular approximation, we can obtain an MRP-ME characterization for the service process B via the MATLAB function `tf2ss.m`:

$$[\mathtt{TB}, \mathtt{HB}, \mathtt{VB}, \mathtt{DB}] = \mathtt{tf2ss}(\begin{bmatrix} 0 & -2/\mu & 6 \end{bmatrix}, \begin{bmatrix} 1/\mu^2 & 4/\mu & 6 \end{bmatrix})$$

which produces

$$T_B = \begin{bmatrix} -0.8 & -0.24 \\ 1 & 0 \end{bmatrix}, \qquad H_B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad V_B = \begin{bmatrix} -0.40 & 0.24 \end{bmatrix}, \qquad D_B = 0.$$

We note that the `tf2ss` function generates a state-space representation out of a proper rational transfer function. We now illustrate the ordered Schur decomposition part of the proposed algorithm for this particular example using MATLAB commands. For this purpose, we first fix $\rho = 0.4$. Note that $m_A = 1$, $n_A = 3$, $m_B = 2$, and $n_B = 1$. We find T_F using (55) as:

$$\begin{pmatrix} .04 & .004 & -.0024 & .012 & -.0072 & 0 & 0 \\ 1 & -.8 & -.24 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -.2667 & .16 & -.8 & -.24 & -.1333 & .08 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -.4 & .24 & 0 & 0 & -.8 & -.24 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

Recall that the matrix T_F has $n_B m_A - 1 = 0$ eigenvalues with positive real parts, one eigenvalue at the origin, and $n_A m_B = 6$ eigenvalues with negative real parts. Since there are no eigenvalues of T_F with positive real parts, we can use the method described in Section 2 and in particular the expression (6). For this purpose, a right null vector of T_F can be found using (61) as $x_R = (-1, 0, -4.1667, 0, -4.1667, 0, -4.1667)^{\mathrm{T}}$ with $\|x_R\|_2 = 7.2858$. Next, by (6), $u = (-8.2858, 0, -4.1667, 0, -4.1667, 0, -4.1667)^{\mathrm{T}}$. By defining the symmetric and orthogonal Householder matrix $Q_F = I - \frac{2uu^{\mathrm{T}}}{u^{\mathrm{T}}u}$ as in (6), we obtain $Q_F^{\mathrm{T}} T_F Q_F =$

$$\begin{pmatrix} 0 & -.5724 & .0031 & -.5735 & .0037 & -.5719 & .0028 \\ 0 & -.8000 & -.7429 & 0 & -.5029 & 0 & -.5029 \\ 0 & .7101 & .0129 & -.2944 & .0156 & -.2876 & .0115 \\ 0 & -.2667 & .1600 & -.8000 & -.2400 & -.1333 & .0800 \\ 0 & -.2899 & .0129 & .7056 & .0156 & -.2876 & .0115 \\ 0 & -.4000 & .2400 & 0 & 0 & -.8000 & -.2400 \\ 0 & -.2899 & .0129 & -.2944 & .0156 & .7124 & .0115 \end{pmatrix},$$

which gives the desired decomposition of (62) with the scalar $T_{F,++} = 0$. Note that we only use matrix vector algebra, without even the need for Schur decomposition for the $\mathrm{M}^X/\mathrm{ME}/1$ queue. Among the existing approaches, Reference [1] relies on transforms, root finding, and transform inversion (similar to [44,12]) which is very different to the method described here.

We now present our results for the $\mathrm{M}^X/\mathrm{D}/1$ queue for varying ρ with the deterministic service time approximated by continued fractions, as in (85), and also by E_{10} and E_{500} distributions, where E_k denotes the Erlang-k distribution. All three approximations use the proposed algorithm and the results are compared against the exact and approximate (using continued fractions) results presented in [1] in Table 2. The proposed algorithm using continued fractions for approximation of the deterministic service time is in perfect accordance (up to six digits) with the results presented in [1] validating accuracy. The Erlangian approximations for the deterministic service time are not as accurate, despite using very high order Erlangian distributions. We also observe that we are able to solve large-scale problems without encountering any computational difficulty. While claiming so, recall that the $\mathrm{M}^X/\mathrm{E}_{500}/1$ problem is large-scale since its solution involves the Schur decomposition of a square matrix of size 1501. We do not give a timing comparison among alternative approaches because there are various alternatives for root finding and transform inversion which were left unspecified in the References [44,1,12]. However, we present the timing results (excluding moment computations) in Table 3 for the Schur decomposition and the Householder approach for this specific problem for various values of ρ for the same $\mathrm{M}^X/\mathrm{E}_{500}/1$ system. Our results show that CPU times are insensitive to system load ρ whereas the Householder approach introduced significant gains for this special case. However, note that the Householder approach only works for the special case $m_A n_B = 1$.

Table 2
The probability mass at zero, mean waiting time, and second moment of the waiting time for the M^X/D/1 queue and its approximants as a function of ρ obtained as in [1] and by the proposed algorithm.

ρ	Measure	Results of [1]		Proposed algorithm		
		Exact	Cont. Frac.	$M^X/E_{10}/1$	$M^X/E_{500}/1$	Cont. Frac.
0.4	$P(W=0)$	.3	.3	.3	.3	.3
	$E[W]$	6.8750	6.8750	7.0417	6.8783	6.8750
	$E[W^2]$	103.9931	103.9931	113.0139	104.1702	103.9931
0.6	$P(W=0)$	.2	.2	.2	.2	.2
	$E[W]$	11.5625	11.5625	11.9375	11.5700	11.5625
	$E[W^2]$	285.7422	285.7422	311.2734	286.2424	285.7422
0.9	$P(W=0)$	.05	.05	.05	.05	.05
	$E[W]$	53.7500	53.7500	56.0000	53.7950	53.7500
	$E[W^2]$	5876.5625	5876.5625	6410.0625	5887.0047	5876.5625

Table 3
CPU times for solving the $M^X/E_{500}/1$ queue using Householder transformations and Schur decomposition for various values of ρ.

ρ	CPU time (s)	
	Householder approach	Schur decomposition
0.9	16.8	76.2
0.99	16.7	75.4
0.999	16.6	76.0

8.2. Example 2: PH/MRP/1 queue

In this numerical example, the interarrival time distributions are assumed to be of phase-type and in particular hyper-exponential with balanced means, as defined in [45]. We fix the squared coefficient of variation to 16 and the mean interarrival time to $1/\rho$. The service times alternate between 0.5 and 1.5 in a deterministic fashion with a mean service time of one leading to a system load of ρ. Although service times are not of MRP type, we propose to use the continued fractions method of [1] for approximating deterministic service times. For this purpose, let (V_1, T_1, H_1, D_1) and (V_2, T_2, H_2, D_2) be MRP-ME characterizations obtained via continued fraction approximations to deterministic service times of 0.5 and 1.5, respectively. Then, we have an MRP-ME characterization of service times:

$$T_B = \begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix}, \qquad V_B = \begin{bmatrix} V_1 & 0 \\ 0 & V_2 \end{bmatrix}, \qquad H_B = \begin{bmatrix} 0 & H_1 \\ H_2 & 0 \end{bmatrix}, \qquad D_B = \begin{bmatrix} 0 & D_1 \\ D_2 & 0 \end{bmatrix}.$$

In particular, we use the second-order continued fraction approximation as given in (85) and the advanced fifth-order approximation to $g_B^*(s) = e^{-s/\mu}$ (see [1]):

$$g_B^*(s) \approx \frac{5(3024 - 1344s/\mu + 252(s/\mu)^2 - 24(s/\mu)^3 + (s/\mu)^4)}{15120 + 8400s/\mu + 2100(s/\mu)^2 + 300(s/\mu)^3 + 25(s/\mu)^4 + (s/\mu)^5}. \tag{86}$$

We compare our results utilizing two different continued fraction approximations against simulations in Table 4 with 99% confidence intervals. Our results clearly show that the fifth-order approximation matches very well with simulation results, whereas the second-order approximation also fared well, especially for higher system loads.

8.3. Example 3: MAP/PH/1 queue

We consider correlated arrivals in the current example. We study the statistical multiplexing of N voice sources with silence detection so that each voice source is modeled as a two-state Interrupted Poisson Process (IPP) with mean off time and the mean on time set to 650 ms and 353 ms, respectively. The mean number of packets generated by each voice source in an on period is set to 22 and the packet sizes are such that voice peak rate in the on state is 32 Kbps. Although packet sizes are fixed, we model them by the E_{20} distribution with the same mean. On the other hand, the multiplexer's service rate is 10 times each voice source's peak rate. In Table 5, we report the probability that an arriving customer finds the queue empty ($P(W=0)$), the mean waiting time in the queue, and the CPU time needed for the voice multiplexing example as a function of the number of voice sources N using the Schur decomposition approach proposed in this paper and the matrix geometric approach outlined in [2] that also derives explicit waiting time expressions for the underlying MAP/PH/1 queue. We also note that we employ the quadratically convergent iterations proposed in [15] with the stopping criterion parameter set to 10^{-10} for finding the associated rate matrix in the MAP/PH/1 queue. All the values we calculate are identical up to six significant digits, so we report them once. We also provide the CPU times needed in the proposed algorithm and the matrix geometric approach of [2] which are calculated from the average of ten runs and using the MATLAB function `cputime`.

Table 4
The probability mass at zero, mean waiting time, and second moment of the waiting time for Example 2 obtained via simulations and the proposed algorithm employing second- and fifth-order continued fraction approximations.

ρ	Measure	Simulation	Analytical (85)	Analytical (86)
0.4	$P(W=0)$	2.7652e−1 ± 1.1574e−4	2.7657e−1	2.7652e−1
	$E[W]$	1.3923 ± 8.2056e−4	1.3948	1.3923
	$E[W^2]$	4.7089 ± 6.0796e−3	4.7257	4.7085
0.6	$P(W=0)$	6.6718e−2 ± 8.8818e−5	6.6514e−2	6.6725e−2
	$E[W]$	7.7044 ± 1.2339e−2	7.6973	7.7041
	$E[W^2]$	1.2310e+2 ± 5.6697e−1	1.2306e+2	1.2309e+2
0.9	$P(W=0)$	8.7953e−3 ± 4.2561e−5	8.5337e−3	8.7989e−3
	$E[W]$	7.0944e+1 ± 5.1958e−1	7.0868e+1	7.0892e+1
	$E[W^2]$	1.0120e+4 ± 1.7221e+2	1.0093e+4	1.0096e+4
0.99	$P(W=0)$	8.1912e−4 ± 2.1858e−5	7.8787e−4	8.1888e−4
	$E[W]$	7.8732e+2 ± 4.4213e+1	7.9160e+2	7.9162e+2
	$E[W^2]$	1.2290e+6 ± 1.9252e+5	1.2538e+6	1.2538e+6

Table 5
The probability mass at zero, mean waiting time, and the second moment of the waiting time for the MAP/E$_{20}$/1 queue as a function of N obtained by the matrix geometric approach utilized in [2] and our proposed algorithm.

N	ρ	CPU time (s)		$P(W=0)$	$E[W]$ (s)	$E[W^2]$ (s^2)
		[2]	Our algorithm			
8	.2816	0.40	0.19	6.541755e−1	5.009202e−4	1.275616e−6
12	.4223	1.50	0.57	5.138854e−1	1.021375e−3	4.857179e−6
16	.5631	3.57	1.08	3.748969e−1	2.558805e−3	4.006403e−5
20	.7039	7.14	1.97	2.406161e−1	9.194948e−3	4.989104e−4
24	.8447	13.01	3.27	1.171852e−1	4.383876e−2	7.208806e−3
26	.9151	17.77	4.02	6.139214e−2	1.201707e−1	4.096586e−2
28	.9854	26.25	4.94	1.005555e−2	1.004764	2.145297

The results are indicative of similar computational complexities of the two approaches, although we are led to believe that the proposed approach is slightly better than the method utilizing the matrix geometric approach presented in [2] when one uses MATLAB. However, the real advantage in using the proposed approach is that the expression for the waiting time distribution is matrix-exponential and all related moments can be derived algorithmically. Moreover, the algorithm applies to many other queueing systems as well. In the matrix geometric approach, it is the steady-state queue length probabilities that have a relatively simple expression (i.e., matrix geometric). On the other hand, it is generally relatively difficult to calculate the waiting times and their moments out of the steady-state queue length probabilities in this approach. As an example, Heindl was able to derive the first two moments for a MAP/PH/1 queue in [2] which were already cumbersome.

8.4. *Example 4: Non-preemptive strict priority queue*

We assume a non-preemptive strict priority scheduling system with two infinite-capacity queues, namely the high and low priority queues, with Poisson arrivals and exponential service times. The arrival rate to the high priority (low priority) queue is λ_h (λ_l) and the service rate of the high priority (low priority) queue is μ_h (μ_l). We define $\rho_h = \lambda_h/\mu_h$ and $\rho_l = \lambda_l/\mu_l$ and we assume $\rho_h + \rho_l < 1$. In this system, high priority jobs have non-preemptive priority over low priority jobs. That is, a high priority job can move ahead of all low priority jobs waiting in the queue for service, but low priority jobs in service are not interrupted by high priority jobs. We are interested in the waiting time distribution for low priority jobs. We note that the mean waiting time for low priority jobs, denoted by $E[W_l]$, is well-known for this system and is given by [46]:

$$E[W_l] = \frac{\lambda_h/\mu_h^2 + \lambda_l/\mu_l^2}{(1-\rho_h)(1-\rho_h-\rho_l)}. \tag{87}$$

Let us now focus on the low priority queue whose service process is governed by the queue occupancy for the high priority queue. For the sake of convenience, let us make the assumption that the high priority queue is of size K. In this case, it is not difficult to show that when the low priority queue is not empty, the service process is modeled with a Markovian service process. To see this, let the pair $(i(t), j(t))$ denote the state of the service process at time t, where $i(t) = 0, 1, \ldots, K$ denotes the number of high priority jobs waiting in the high priority queue or in service, and $j(t) = 0$ designates a low priority job in service and $j(t) = 1$ otherwise, all at time t. Note that, the state $(0, 0)$ is never visited. From state $(i, 0)$, $1 \le i < K$, we will move to state $(i+1, 0)$ with rate λ_h and from state $(i, 0)$, $1 < i \le K$, we move to state $(i-1, 0)$ with rate μ_h. When in state $(1, 0)$, we transit to state $(0, 1)$ with rate μ_h. On the other hand, we transit to state $(i+1, 1)$ with rate λ_h from state $(i, 1)$, $0 \le i < K$. Finally, we transit from state $(i, 1)$, $0 < i \le K$ to state $(i, 0)$ with rate μ_l with a service completion and also from state $(0, 1)$ to the same state with rate μ_l with a service completion. Obviously, this description

Table 6
$E[W_l]$ obtained using the formula in (87) and the proposed algorithm as a function ρ_l, ρ_h, and K, for $\mu_h = \mu_l = 1$.

K	ρ_h	ρ_l	$E[W_l]$	
			Formula (87)	Proposed algorithm
10	0.2	0.7	11.25	10.8345
		0.799	1.2487e+3	1.2482e+3
		0.79999	1.25e+5	1.2411e+5
100	0.2	0.7	11.25	10.8343
		0.799	1.2487e+3	1.2483e+3
		0.79999	1.25e+5	1.25e+5
10	0.8	0.1	45	11.8852
		0.199	4.9950e+3	119.5920
		0.19999	5e+5	125.1572
100	0.8	0.1	45	24.4364
		0.199	4.9950e+3	4.9743e+3
		0.19999	5e+5	4.9989e+5

completes the characterization of the Markovian service process of the low priority queue unless the low priority queue is empty. We obtain the distribution of W_l and in particular $E[W_l]$, assuming that the Lindley equation (1) is satisfied for Poisson arrivals and for MSP departures as characterized above, irrespective of the low priority queue status. We compare our findings against the ones obtained using the formula given in (87). It is clear that these two values should match as $K \to \infty$ and $\rho_h + \rho_l \to 1$. The reason for the latter is that, in this case, the probability that the low priority queue is zero approaches to zero and the low priority waiting time is then well governed by the Lindley equation (1). However, in general B_k in the Lindley equation (1) will be dependent on whether the low priority queue is empty or not. We fix $\mu_h = \mu_l = 1$ and the $E[W_l]$ results are presented in Table 6 as a function ρ_l, ρ_h, and K. Our findings verify our conjecture. In this example, we not only validate our proposed algorithm for Markovian service processes in a certain asymptotic regime, but we also emphasize the importance of the service model with auto-correlated service times when the queue is empty.

9. Conclusions

In this paper, we introduce a stochastic model, namely the MRP-ME, which is a Markov renewal process with a matrix-exponential kernel. This model is a superset of the well-known phase-type and matrix-exponential-type renewal processes, as well as the more general Markovian and rational arrival processes. An MRP-ME is also general enough to model batch arrivals (or services) as demonstrated in the numerical example involving the $M^X/ME/1$ queue. We study the steady-state waiting time in a semi-Markov queue with infinite-capacity in which the interarrivals and services are both modeled with MRP-MEs. Without having to solve for the steady-state queue lengths by matrix geometric techniques, we introduce an algorithm that directly finds the waiting time distribution which is in matrix-exponential form. The algorithm to obtain the parameters of the matrix-exponential form is relatively easy to implement and the numerical engine relies on the ordered Schur decomposition whose various stable and efficient implementations exist in the literature. The numerical examples we present lead us to believe that the proposed algorithm is a promising candidate for a wide range of Markov renewal queueing problems. As future work, we list the possibility of allowing inter-dependence between arrivals and services, finite capacity queues, more general service models of Markov renewal type, and the study of relationship between the proposed model and the existing batch arrival and service models.

Acknowledgment

This work was supported in part by The Science and Research Council of Turkey (Tübitak) under project EEEAG-106E046. A preliminary version of this paper appeared in [47].

References

[1] M.L. Chaudhry, D.W. Choi, K.C. Chae, Computational analysis of stationary waiting-time distributions of $GI^X/R/1$ and $GI^X/D/1$ queues, Probab. Engrg. Inform. Sci. 19 (1) (2005) 121–140.

[2] A. Heindl, Traffic-Based Decomposition of General Queueing Networks with Correlated Input Processes, Shaker Verlag, Aachen, Germany, 2001, Ph.D. Thesis.

[3] U.N. Bhat, Queueing systems with first-order dependence, J. Oper. Res. Soc. India 6 (1969) 1–24.

[4] I.J.-B.F. Adan, V.G. Kulkarni, Single-server queue with Markov-dependent inter-arrival and service times, Queueing Syst. 45 (2) (2003) 113–134.

[5] G.U. Hwang, K. Sohraby, Performance of correlated queues: The impact of correlated service and inter-arrival times, Performance Evaluation 55 (1–2) (2004) 129–145.

[6] J.H.A. de Smit, Explicit Wiener-Hopf factorizations for the analysis of multidimensional queues, in: J. Dshalalow (Ed.), Advances in Queueing: Theory, Methods and Open Problems, CRC Press, Boca Raton, FL, 1995, pp. 293–311.

[7] T. Ozawa, Analysis of queues with Markovian service processes, Stoch. Mod. 4 (2004) 391–413.

[8] E. Çınlar, Introduction to Stochastic Processes, Prentice Hall, 1975.

[9] J.H. de Smit, The single server semi-Markov queue, Stochastic Process. Appl. 22 (1986) 37–50.

[10] B. Sengupta, The semi-Markovian queue: Theory and applications, Commun. Statist. Stoch. Models 6 (3) (1990) 383–413.

[11] J. Kumaran, K. Mitchell, A. van de Liefvoort, A spectral approach to compute performance measures in a correlated single server queue, Performance Evaluation Review 33 (2) (2005) 12–14.

[12] D.M. Lucantoni, The BMAP/G/1 queue: A tutorial, in: L. Donatiello, R. Nelson (Eds.), Models and Techniques for Performance Evaluation of Computer and Communication Systems, Springer-Verlag, 1993, pp. 330–358.

[13] M.F. Neuts, Structured Stochastic Matrices of $M/G/1$ Type and Their Applications, Marcel Dekker, Inc, New York, 1989.

[14] G. Latouche, V. Ramaswami, A logarithmic reduction algorithm for quasi-birth-death processes, J. Appl. Probab. 30 (1993) 650–674.

[15] V. Naoumov, U. Krieger, D. Wagner, Analysis of a multi-server delay-loss system with a general Markovian arrival process, in: S. Chakravarthy, A. Alfa (Eds.), Matrix-analytical methods in Stochastic models, Marcel Dekker, 1997, pp. 43–66.

[16] D. Bini, B. Meini, On the solution of a nonlinear matrix equation arising in queueing problems, SIAM J. Matrix. Anal. Appl. 17 (1996) 906–926.

[17] N. Akar, K. Sohraby, An invariant subspace approach in M/G/1 and G/M/1 type Markov chains, Commun. Statist. - Stoch. Models 13 (3) (1997) 381–416.

[18] A. Riska, E. Smirni, Exact aggregate solutions for M/G/1-type Markov processes, in: SIGMETRICS '02: Proceedings of the 2002 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, ACM Press, New York, NY, USA, 2002.

[19] V. Ramaswami, A stable recursion for the steady state vector in Markov chains of M/G/1 type, Commun. Statist.- Stoch. Models 4 (1988) 183–263.

[20] G. Latouche, V. Ramaswami, Introduction to Matrix Analytic Methods in Stochastic Modeling, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1999.

[21] S.R. Mahabhashyam, N. Gautam, On queues with Markov modulated service rates, Queueing Syst. Theory Appl. 51 (1-2) (2005) 89–113.

[22] M.F. Neuts, Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach, The Johns Hopkins University Press, 1981.

[23] L.R. Lipsky, Queueing Theory: A Linear Algebraic Approach, MacMillan, 1992.

[24] D.M. Lucantoni, K.S. Meier-Hellstern, M.F. Neuts, A single-server queue with server vacations and a class of non-renewal arrival processes, Adv. Appl. Prob. 22 (1990) 676–705.

[25] S. Asmussen, M. Bladt, Point processes with finite-dimensional conditional probabilities, Stochastic Process Appl. 82 (1) (1999) 127–142.

[26] N. Akar, A matrix analytical method for the discrete time Lindley equation using the generalized Schur decomposition, in: SMCtools '06: Proceeding from the workshop on tools for solving structured Markov chains, 2006.

[27] Z. Bai, J. Demmel, M. Gu, Inverse free parallel spectral divide and conquer algorithms for nonsymmetric eigenproblems, Numer. Math. 76 (1997) 389–396.

[28] Z. Bai, J. Demmel, Inverse free parallel spectral divide and conquer algorithms for nonsymmetric eigenproblems, Computer Science Division Report CSD-94-793, University of California at Berkeley (Feb. 1994).

[29] G.H. Golub, C.F. van Loan, Matrix Computations, 3rd ed., The Johns Hopkins University Press, 1996.

[30] Z. Bai, J.W. Demmel, On swapping diagonal blocks in real Schur form, Linear Algebra Appl. 186 (1993) 73–95.

[31] J.H. Brandts, Matlab code for sorting real Schur forms, Numer. Linear Algebra Appl. 9 (3) (2002) 249–261.

[32] J. Demmel, Non-Hermitian Eigenproblems, in: Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, H. van der Vorst (Eds.), Templates for the solution of algebraic eigenvalue problems: A practical guide, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.

[33] P.V. Dooren, The basics of developing numerical algorithms, IEEE Control Systems Mag. 24 (1) (2004) 18–27.

[34] LAPACK Users's Guide, second edition, 1995.

[35] J.W. Eaton, GNU Octave: A high-level interactive language for numerical computations, Bristol, United Kingdom (2005).

[36] The MATH WORKS Inc., MATLAB 7.0.0.19901 (R14) (2005).

[37] T. Kailath, Linear Systems, Prentice Hall, 1980.

[38] D.M. Lucantoni, New results for the single server queue with a batch Markovian arrival process, Stoch. Models 7 (1991) 1–46.

[39] H. Masuyama, Studies on algorithmic analysis of queues with batch Markovian arrival streams, Ph.D. Thesis, Kyoto University, Japan (2003).

[40] A.V. Oppenheim, A.S. Willsky, S.H. Nawab, Signals & Systems, 2nd ed., Prentice-Hall, Inc, Upper Saddle River, NJ, USA, 1996.

[41] N. Akar, Solving the ME/ME/1 queue with state-space methods and the matrix sign function, Perform. Eval. 63 (2) (2006) 131–145.

[42] J. Kumaran, K. Mitchell, A.V.D. Liefvoort, An efficient solution to the waiting time distribution in a correlated single server queue, in: Proc. ITC'19, Beijing, China, 2005.

[43] R. Bekker, O.J. Boxma, An M/G/1 queue with adaptable service speed, Stoch. Mod. 23 (3) (2007) 373–396.

[44] P.J. Burke, Delays in single-server queues with batch input, Oper. Res. 23 (4) (1975) 830–833.

[45] H.C. Tijms, Stochastic Modelling and Analysis: A Computational Approach, John Wiley and Sons, 1986.

[46] D. Bertsekas, R. Gallager, Data Networks, Prentice-Hall, Inc, Upper Saddle River, NJ, USA, 1987.

[47] N. Akar, K. Sohraby, Solving the single server semi-Markov queue with matrix exponential kernel matrices for interarrivals and services, in: Valuetools '06: Proceedings of the 1st international conference on Performance evaluation methodologies and tools, 2006.

N. Akar received the B.S. degree from Middle East Technical University, Turkey, in 1987 and M.S. and Ph.D. degrees from Bilkent University, Turkey, in 1989 and 1994, respectively, all in electrical and electronics engineering. From 1994 to 1996, he was a visiting scholar and a visiting assistant professor in the Computer Science Telecommunications program at the University of Missouri-Kansas City.

He joined the Technology Planning and Integration group at Long Distance Division, Sprint, Overland Park, Kansas, in 1996, where he held a senior member of technical staff position from 1999 to 2000. Since 2000, he has been with Bilkent University, currently as an associate professor. He is actively involved in European Commission FP7 projects BONE and Wimagic. Recently, Nail Akar has been one of the technical program co-chairs of the 2008 IEEE IP Operations and Management Workshop and the 2008 ICCCN Track on QoS Control and Traffic Modeling. His current research interests include performance analysis of computer and communication networks, queueing systems, traffic engineering, traffic control and resource allocation, wireless and optical networks.

K. Sohraby is currently the Curators' Professor of Computer Science and Electrical Engineering at the University of Missouri-Kansas City. He received the B.Eng. and M.Eng. degrees from McGill University, Montreal, Canada, in 1979 and 1981, respectively, and the Ph.D. degree from the University of Toronto, Toronto, Canada, in 1985, all in electrical engineering. His current research interests include design, analysis and control of high-speed computer and communications networks, traffic management and analysis, multimedia networks, networking aspects of wireless and mobile communications, analysis of algorithms, parallel processing and large-scale computations. Refer to http://www.sce.umkc.edu/~sohrabyk/for a detailed biography.