

A Matrix Analytical Method for the Discrete Time Lindley Equation using the Generalized Schur Decomposition

Nail Akar
Electrical and Electronics Eng.
Bilkent University
Ankara, Turkey
akar@ee.bilkent.edu.tr

ABSTRACT

In this paper, we study the discrete time Lindley equation governing an infinite size GI/GI/1 queue. In this queuing system, the arrivals and services are independent and identically distributed but they obey a discrete time matrix geometric distribution not necessarily with finite support. Our GI/GI/1 model allows geometric batch arrivals and also treats late, early, and hybrid arrival models in a unified manner. We reduce the problem of finding the steady state probabilities for the Lindley equation to finding the generalized ordered Schur form of a matrix pair (E, A) where the size of these matrices are the sum, not the product, of the orders of individual arrival and service distributions. The approach taken in this paper is purely matrix analytical and we obtain a matrix geometric representation for the related quantities (queue lengths or waiting times) for the discrete time GI/GI/1 queue using this approach.

Categories and Subject Descriptors

G.3 [Mathematics of Computing]: Probability and Statistics—Markov processes, Queueing theory; G.1.3 [Numerical Analysis]: Numerical Linear Algebra; C.4 [Computer Systems Organization]: Performance of Systems—Modeling Techniques, Performance attributes

General Terms

Algorithms, Performance

Keywords

Lindley equation, discrete-time queues, matrix geometric distribution, generalized ordered Schur decomposition

1. INTRODUCTION

In this paper, we consider a discrete time queue with infinite size. The time axis is divided into slots of equal length which is taken as one unit with slot n denoting the time

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SMCTOOLS'06 October 10, 2006, Pisa, Italy.

Copyright 2006 ACM 1-59593-506-1/06/10...\$5.00

interval $[n, n + 1)$. We study the following GI/GI/1 discrete time Lindley equation (recursion) [18]:

$$q(n + 1) = \max(q(n) + a(n) - b(n), 0) + c(n), n \geq 0, \quad (1)$$

where $a(n)$, $b(n)$, and $c(n)$ are independent and identically distributed non-negative random sequences with probability mass functions (PMF) $p_a(k)$, $p_b(k)$, and $p_c(k)$, respectively. In most discrete time queueing systems including GI/GI/1 queues, queue lengths or waiting times satisfy a Lindley equation (1) of some form; see [14] and [7] for a detailed treatment of discrete time queueing models including late and early arrival models. In this study, we are interested in the probability mass function (PMF) of the limiting random variable in the recursion (1):

$$p_q(k) = \Pr\{q = k\}, k \geq 0, \quad (2)$$

where $q = \lim_{n \rightarrow \infty} q(n)$. We assume the limiting random variable q exists throughout this paper which is true when $E[a(n)] + E[c(n)] < E[b(n)]$. We note that the underlying Markov chain has a probability transition matrix $P = \{P_{ij}\}$, where

$$\begin{aligned} P_{ij} &= \Pr\{a(n) - b(n) \leq -i\} \Pr\{c(n) = j\} \\ &+ \sum_{l=0}^{j-1} \Pr\{c(n) = l\} \Pr\{a(n) - b(n) = j - i - l\} \end{aligned} \quad (3)$$

In case the process $b(n)$ is of finite support, then the Markov chain above is an M/G/1 type chain with multiple boundaries and with scalar entries [11],[2]. When the processes $a(n)$ and $b(n)$ are both of finite support then we have a G/M/1 type Markov chain with multiple boundary levels and scalar entries [11]. However, we do not impose finite support constraints on the related distributions in the current paper.

There is a vast amount of literature and a variety of approaches for discrete time queues which is generally known to be difficult. Most of the existing results rely on Wiener Hopf factorizations. The reference [12] gives a fast iterative procedure for the Wiener Hopf factorization when the involved distributions are of finite support and the computational complexity per iteration is relatively low. A similar factorization approach is used in [13] to solve the more general SMP/G/1 queue with semi-Markov arrivals. On the other hand, the reference [29] uses a root finding method to study the GI/G/1 system in discrete time. The $GI^X/G/1$ queue with batch arrivals is studied in discrete time using a similar method in [9].

The alternative approach to discrete time queues is the matrix analytical approach that relies on matrix and vector operations as opposed to polynomial operations and root finding [23]. However, most matrix analytical methods are well-suited more for the M/G/1 and G/M/1 type Markov chains [23],[24] and also for those with multiple boundary levels [11] and not for the Markov chains in the form (3). If the problem faced is of M/G/1 or G/M/1 type, then one can use numerically efficient algorithms like the cyclic reduction algorithm [6], the invariant subspace approach [1], and the technique proposed in [26]. When the arrival and service processes possess finite support PMFs then the probability transition matrix of interest turns out to be of Quasi Birth and Death (QBD) type [3] for which quadratically convergent matrix geometric methods exist like the logarithmic reduction algorithm for [17], the invariant subspace approach [1] and the iterative scheme of [22]. Recently, a matrix analytical algorithm is provided for solving general Markov chains of G/G/1 type [27]. The G/G/1 type Markov chains with block banded structures are studied using matrix geometric methods in [28].

In this paper, we will assume that all the three processes $a(n)$, $b(n)$, and $c(n)$ in the Lindley equation (1) have matrix geometric distributions. Such distributions are more general than discrete phase type processes and finite support distributions where the latter can also be viewed as a phase type distribution.

Our main result is that the limiting random variable q in this case also has a matrix geometric distribution and finding the parameters of this distribution is shown to reduce to solving a generalized spectral divide and conquer (GSDC) problem applied on a square matrix pair of size being the sum of the order of the individual processes. Given a real matrix pair (E, A) , the GSDC problem of interest in this paper is finding orthogonal matrices Q and Z such that

$$Q^T E Z = \begin{bmatrix} E_{oo} & E_{oi} \\ 0 & E_{ii} \end{bmatrix}, Q^T A Z = \begin{bmatrix} A_{oo} & A_{oi} \\ 0 & A_{ii} \end{bmatrix},$$

where the generalized eigenvalues of the pair (E_{oo}, A_{oo}) are exactly the same as those of (E, A) that are outside the unit disk. Similarly, the generalized eigenvalues of the pair (E_{ii}, A_{ii}) are exactly the same as those of (E, A) that are inside the unit disk. The advantages of the proposed approach are

- The generality of the model allows one to use a single unifying algorithm for different well-known discrete time GI/GI/1 queueing models including late and early arrival models.
- We do not construct structured Markov chains to solve for the limiting distribution. Studying the Lindley equation directly without having to construct the structured Markov chain offers potential advantages since our experience with the numerical analysis of queues leads us to believe that one of the hard parts even for a sophisticated user is to form or represent the blocks or submatrices of the associated Markov chain. This feature of our proposed approach may allow researchers and practitioners to use the proposed algorithms with little additional effort if their queueing analysis problem fits in the framework defined through the Lindley recursion (1).
- The approach does not use root finding and benefits from being purely matrix analytical as explained in [23] and [24].
- GSDC for GI/GI/1 queues applies on matrices of additive size whereas a number of existing matrix analytical methods for GI/GI/1 queues operate on matrices of multiplicative size [3].
- We propose the use the generalized ordered real Schur decomposition as the numerical engine for solving the GSDC which is known to be effectively used for solving Riccati equations for decades [15].
- Our model allows probability masses at the origin for all the involved processes and therefore it is also possible to address batch arrivals and services with geometric batch sizes.
- In addition to the GSDC, the proposed algorithm is based on only matrix-matrix and matrix-vector operations, and solution of linear equations and is therefore quite easy to implement.

However, we note the real advantage of the proposed algorithm of this paper is when the involved distributions have infinite support or when they can be modeled or approximated well by infinite support matrix geometric distributions with much smaller orders.

The remainder of the paper is organized as follows. Section 2 provides preliminaries and notation used throughout the paper. We provide the matrix analytical solution for the Lindley equation in Section 3. Section 4 provides numerical examples to validate the effectiveness of the proposed approach. We conclude in the final section.

2. PRELIMINARIES AND NOTATION

Let z be a complex number and \mathbb{C} denote the complex plane. The open unit disk in the complex plane, denoted by Δ^i , is defined as $\{z \in \mathbb{C} : |z| < 1\}$. The unit circle is represented by $\Delta^1 = \{z \in \mathbb{C} : |z| = 1\}$. The complement of these two sets is denoted by Δ^o . Above, the symbols i and o stand for “inside” and “outside” the unit circle, respectively. We use uppercase (lowercase) letters to denote matrices (vectors or scalars). I and e denote the identity matrix and a column matrix of ones of appropriate sizes, respectively. For a real matrix A , A^T denotes A transposed and A is orthogonal if $A^T A = I$. For a given real, non-symmetric matrix A and a region \mathcal{D} of the complex plane \mathbb{C} , there is an orthogonal matrix Q such that

$$Q^T A Q = \begin{bmatrix} A_{\mathcal{D}\mathcal{D}} & A_{\mathcal{D}\bar{\mathcal{D}}} \\ 0 & A_{\bar{\mathcal{D}}\bar{\mathcal{D}}} \end{bmatrix}, \quad (4)$$

where the eigenvalues of $A_{\mathcal{D}\mathcal{D}}$ are exactly the same as the eigenvalues of A in \mathcal{D} . Here $\mathcal{D} \cap \bar{\mathcal{D}} = \emptyset$, $\mathcal{D} \cup \bar{\mathcal{D}} = \mathbb{C}$. This problem is called the ordinary spectral divide and conquer (SDC) problem [5]. Let A and E be two $n \times n$ matrices. A complex scalar λ and a nonzero row vector x satisfying

$$xA = \lambda xE, \quad x \neq 0,$$

are called a generalized eigenvalue and the generalized left eigenvector for the matrix pair (E, A) associated with λ , respectively. When $E = I$, we have the definition of an ordinary eigenvalue of the matrix A . $\lambda(E, A)$ denotes the

set of all generalized eigenvalues of (E, A) . For a given pair of real matrices E and A , one can find orthogonal matrices Q and Z such that

$$Q^T E Z = \begin{bmatrix} E_{\mathcal{D}\mathcal{D}} & E_{\mathcal{D}\bar{\mathcal{D}}} \\ 0 & E_{\bar{\mathcal{D}}\bar{\mathcal{D}}} \end{bmatrix}, Q^T A Z = \begin{bmatrix} A_{\mathcal{D}\mathcal{D}} & A_{\mathcal{D}\bar{\mathcal{D}}} \\ 0 & A_{\bar{\mathcal{D}}\bar{\mathcal{D}}} \end{bmatrix}, \quad (5)$$

where $\lambda(E_{\mathcal{D}\mathcal{D}}, A_{\mathcal{D}\mathcal{D}})$ is exactly the same as $\lambda(E, A)$ restricted to the region \mathcal{D} [5]. A real square matrix pair (E, A) of size n can be transformed using orthogonal matrices Q_1 and Z_1 into the so-called generalized real Schur form by writing $Q_1^T E Z_1 = E_1, Q_1^T A Z_1 = A_1$ where A_1 is quasi-upper triangular, which means that the matrix A_1 has either 1-by-1 or 2-by-2 diagonal blocks on the diagonal and E_1 is diagonal. By reordering the blocks by orthogonal transformations, the generalized eigenvalues are made to appear in any order and one can obtain orthogonal matrices Q and Z such that the identity (5) is satisfied and the matrices $A_{\mathcal{D}\mathcal{D}}$ and $A_{\bar{\mathcal{D}}\bar{\mathcal{D}}}$ are quasi-upper triangular, $E_{\mathcal{D}\mathcal{D}}$ and $E_{\bar{\mathcal{D}}\bar{\mathcal{D}}}$ are diagonal, and the generalized eigenvalues of $(E_{\mathcal{D}\mathcal{D}}, A_{\mathcal{D}\mathcal{D}})$ are the same as those of (E, A) in \mathcal{D} [4],[10]. This form is called the generalized ordered real Schur form and the operation to obtain this form is called the generalized ordered real Schur decomposition (GORS). The ordered generalized real Schur form implementations are available in various platforms in LAPACK [4], OCTAVE, and MATLAB 7.0 [21]. Of particular interest to the current paper is when $\mathcal{D} = \Delta^\circ \cup \Delta^1$.

Let $x(k)$ be a vector function of the discrete variable $k \in \mathbb{Z}$. The two sided z-transform of $x(k)$ is given by

$$x^*(z) = \sum_{-\infty}^{+\infty} z^k x(k). \quad (6)$$

We use the $*$ notation for z-transforms throughout this paper. The unit impulse function $\delta(k)$ is defined as $\delta(0) = 1, \delta(k) = 0, k \neq 0$. We note that existence of unit impulse term in the PMF is indicative of a probability mass at the origin. The degree of a polynomial $n^*(z)$ in the indeterminate z is denoted by $\deg(n^*)$. A transform is said to be rational if

$$x^*(z) = \frac{n^*(z)}{d^*(z)},$$

for some polynomials $n^*(z)$ and $d^*(z)$. The rational transform $x^*(z)$ is strictly proper if $\deg(n^*) < \deg(d^*)$ and is proper if $\deg(n^*) = \deg(d^*)$. The poles of the rational function $x^*(z)$ are the roots of the denominator polynomial $d^*(z)$. Any strictly proper rational function $x^*(z)$ can additively be decomposed as

$$x^*(z) = x_{i,1}^*(z) + x_o^*(z),$$

where the poles of $x_{i,1}^*(z)$ and $x_o^*(z)$ reside in $\Delta^i \cup \Delta^1$ and Δ° , respectively. Moreover, this decomposition is unique. If $x_{i,1}^*(z) = 0$ then $x^*(z)$ is called stable. If $x_o^*(z) = 0$, then $x^*(z)$ is called anti-stable.

A linear time-invariant discrete time dynamical system is represented by the following set of difference equations [16]:

$$x(k+1) = x(k)T + u(k)v, k \geq 0, \quad (7)$$

$$y(k) = x(k)h + u(k)d, \quad (8)$$

where $u(k)$ and $y(k)$ denote the input and output, respectively, $x(k) = (x_1(k), \dots, x_m(k))$ is called the state vector and its components are called the state variables, or simply the states. In this representation, v is a row vector of

size m , T is $m \times m$, h is a column vector of size m , and d is a scalar. Considering zero initial state, i.e., $x(0) = 0$, the transfer function $g^*(z)$ between the input and output is written as [16]:

$$y^*(z) = u^*(z)g^*(z) = u^*(z)(v(z^{-1}I - T)^{-1}h + d), \quad (9)$$

where $u^*(z)$ and $y^*(z)$ are the z-transforms of the input and output, 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 function $g^*(z)$ if (9) holds [16]. 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 or irreducible if one cannot satisfy the identity (9) with a lesser order.

A discrete phase type (DPH) distribution is the distribution of the time until absorption in a discrete state discrete time Markov chain (DTMC) with m transient states and one absorbing state. Let the transient states be numbered as $1, 2, \dots, m$ and the absorbing state as $m+1$. The one step probability transition matrix of this DTMC can then suitably be partitioned as

$$P = \begin{bmatrix} T & h \\ 0 & 1 \end{bmatrix}, \quad (10)$$

with an $m \times m$ matrix T and an $m \times 1$ vector h . The initial probability vector can also be partitioned as (v, d) for a $1 \times m$ row vector v and a scalar d . It is easy to show that the DPH-distributed random variable x has a PMF $p_x(k), k \geq 0$ which is of the form

$$p_x(k) = P(x = k) = \begin{cases} vT^{k-1}h & k \geq 1 \\ d & k = 0 \end{cases} \quad (11)$$

and a probability generating function (PGF) $p_x^*(z)$ of the form

$$p_x^*(z) = v(z^{-1}I - T)^{-1}h + d, \quad (12)$$

$$= vz(I - zT)^{-1}h + d. \quad (13)$$

Note that $p_x^*(z)$ is rational and can be written as

$$p_x^*(z) = \frac{n_x^*(z)}{d_x^*(z)},$$

for a numerator polynomial $n_x^*(z)$ and a denominator polynomial $d_x^*(z)$ with $\deg(n_x^*) \geq \deg(d_x^*)$. Sometimes, it is desirable to write

$$p_x^*(z^{-1}) = \frac{\bar{n}_x^*(z)}{\bar{d}_x^*(z)},$$

for a numerator polynomial $\bar{n}_x^*(z)$ and a denominator polynomial $\bar{d}_x^*(z)$ with $\deg(\bar{n}_x) \leq \deg(\bar{d}_x) = \deg(n_x^*)$. Note that $\bar{d}_x^*(z) = \det(zI - T)$. A random variable is said to possess a matrix geometric (MG) distribution if the PMF is of the same form (11) but its parameters v, T , and h do not necessarily carry the same probabilistic interpretation. The size of the matrix T is called the order of the MG distribution. Moreover, an MG distribution is characterized by the ordered quadruple (v, T, h, d) and this representation is irreducible if one cannot find a representation with a smaller order. We note that matrix geometric distributions generalize DPH distributions in the same way as matrix exponential distributions generalize continuous phase type distributions [19]. It is also clear that MG distributions possess rational

z-transforms. The case of $d \neq 0$ can also be visualized by allowing batch arrivals at arrival epochs with a geometric batch size distribution of $d^{i-1}(1-d)$, $i \geq 1$ and interarrival times thus modeled as an MG distribution $(v/(1-d), T, t, 0)$. The factorial moments of matrix geometric distributions can be found by differentiating (13) successively with respect to z and setting $z = 1$. For example the first two factorial moments of an MG-distributed random variable x can be found through the following expressions:

$$\begin{aligned} E[x] &= v(I-T)^{-2}h \\ E[x^2 - x] &= 2v(I-T)^{-3}Th \end{aligned} \quad (14)$$

Let x_1 and y_1 be two independent random variables with MG-distributions characterized with the quadruples

$$(v_1, T_1, h_1, d_1) \text{ and } (v_2, T_2, h_2, d_2),$$

respectively. The random variable $x = x_1 + x_2$ is then also MG-distributed with the characterizing quadruple

$$\left([v_1, d_1 v_2], \begin{bmatrix} T_1 & h_1 v_2 \\ 0 & T_1 \end{bmatrix}, [h_1 d_2, h_2]^T, d_1 d_2 \right) \quad (15)$$

3. DISCRETE TIME LINDLEY EQUATION

Let us visit back the Lindley equation (1). We assume that the processes $a(n)$, $b(n)$, and $c(n)$ are MG-distributed and they are characterized by the irreducible quadruples (v_a, T_a, h_a, d_a) , (v_b, T_b, h_b, d_b) , and (v_c, T_c, h_c, d_c) , with orders m_a , m_b , and m_c , and PGFs $p_a^*(z)$, $p_b^*(z)$, and $p_c^*(z)$, respectively. In particular, we write

$$p_b^*(z) = \frac{n_b^*(z)}{d_b^*(z)}, p_b^*(z^{-1}) = \frac{\bar{n}_b^*(z)}{d_b^*(z)}.$$

For the development of this paper, we first need the following theorem whose proof is provided in the appendix.

THEOREM 1. *Let x and y be two independent non-negative random variables with matrix geometric PMFs or equivalently with rational PGFs:*

$$p_x^*(z) = \frac{n_x^*(z)}{d_x^*(z)}, \deg(n_x^*) \geq \deg(d_x^*), \quad (16)$$

and

$$\begin{aligned} p_y^*(z) &= \frac{n_y^*(z)}{d_y^*(z)}, \deg(n_y^*) \geq \deg(d_y^*), \\ p_y^*(z^{-1}) &= \frac{\bar{n}_y^*(z)}{d_y^*(z)}, \deg(\bar{n}_y^*) \leq \deg(d_y^*), \end{aligned}$$

respectively. Then there exists a unique polynomial $u^*(z)$ of degree $\deg(n_y^*)$ with $u^*(1) = 0$ such that the random variable defined by $t = \max(x - y, 0)$ has a rational PGF $p_t^*(z)$ expressed by

$$p_t^*(z) = p_x^*(z)p_y^*(z^{-1}) - \frac{u^*(z)}{d_y^*(z)}. \quad (17)$$

Conversely, if one can find a polynomial $u^*(z)$ with degree $\deg(n_y^*)$ satisfying $u^*(1) = 0$ and the right hand side of (17) having all its poles in \mathbb{C}^o then identity (17) gives the expression for the PGF of the random variable t .

Applying the result of Theorem 1 to the Lindley equation (1), we immediately obtain the existence of a polynomial

$u^*(z)$ with degree m_b and with $u^*(1) = 0$ such that

$$p_q^*(z) = \left(p_q^*(z)p_a^*(z)p_b^*(z^{-1}) - \frac{u^*(z)}{d_b^*(z)} \right) p_c^*(z). \quad (18)$$

Conversely, if one can find $u^*(z)$ with degree m_b and with $u^*(1) = 0$ satisfying (18) for a stable $p_q^*(z)$ then (18) gives the solution for the PGF of the limiting distribution q . Theorem 1 can directly be used to find $p_q^*(z)$ by root finding and polynomial factorization. However, in this paper we seek a matrix analytical solution that only uses the matrix geometric representations of the individual processes driving the Lindley equation as opposed to an approach that uses transforms. Therefore, we will use Theorem 1 to validate the matrix analytical approach but not for numerical calculations. Towards this goal, we first define the following linear time-invariant discrete time dynamical system, say system \mathcal{A} , for $k \geq 0$ associated with the individual discrete time process $a(n)$:

$$x_a(k+1) = x_a(k)T_a + u_a(k)v_a, x_a(0) = 0, \quad (19)$$

$$y_a(k) = x_a(k)h_a + u_a(k)d_a \quad (20)$$

Here x_a , u_a , and y_a is the state, input, and output of system \mathcal{A} , respectively. We then define another dynamical system denoted by \mathcal{B} for $k \geq 0$ associated with the discrete time process $b(n)$:

$$x_b(k+1)T_b = x_b(k) + u_b(k+1)v_b, x_b(0) = x_0, \quad (21)$$

$$y_b(k) = -x_b(k)h_b + u_b(k)d_b + \delta(k)d_0 \quad (22)$$

Since T_b can be singular, system \mathcal{B} is not an ordinary system but is rather called a discrete time descriptor system and x_b , u_b , and y_b is the descriptor, input, and output of descriptor system \mathcal{B} , respectively [20]. The system parameters x_0 and d_0 are not known yet but they are to be determined. Finally, we define a dynamical system denoted by \mathcal{C} for $k \geq 0$ associated with the individual discrete time process $c(n)$:

$$x_c(k+1) = x_c(k)T_c + u_c(k)v_c, x_c(0) = 0, \quad (23)$$

$$y_c(k) = x_c(k)h_c + u_c(k)d_c \quad (24)$$

Above, x_c , u_c , and y_c is the state, input, and output of system \mathcal{C} , respectively. We then interconnect the three systems in the following feedback configuration:

$$u_a(k) = y_c(k), \quad (25)$$

$$u_b(k) = y_a(k), \quad (26)$$

$$u_c(k) = y_b(k). \quad (27)$$

We will now show that $y_c^*(z)$ in this feedback configuration satisfies the expression (18) for $p_q^*(z)$. In order to show this, we first note from (19) and (20) that

$$y_a^*(z) = u_a^*(z)p_a^*(z), \quad (28)$$

and from (22)

$$y_b^*(z) = -x_b^*(z)h_b + u_b^*(z)d_b + d_0. \quad (29)$$

Furthermore, using (21) and the basic properties of z-transforms we show that

$$z^{-1}(x_b^*(z) - x_0)T_b = x_b^*(z) + z^{-1}(u_b^*(z) - u_b(0))v_b. \quad (30)$$

Using (29) and (30), it is not difficult to show

$$\begin{aligned} y_b^*(z) &= u_b^*(z)(v_b(Iz - T_b)^{-1}h_b + d_b), \\ &\quad - (u_b(0)v_b - x_0T_b)(Iz - T_b)^{-1}h_b + d_0 \\ &= u_b^*(z)p_b^*(z^{-1}) - \bar{x}_0(Iz - T_b)^{-1}h_b + d_0, \end{aligned} \quad (31)$$

where $\bar{x}_0 = u_b(0)v_b - x_0T_b$. In the final step, we write $y_c^*(z)$:

$$\begin{aligned} y_c^*(z) &= y_b^*(z)p_c^*(z), \\ &= (u_b^*(z)p_b^*(z^{-1}) - \bar{x}_0(Iz - T_b)^{-1}h_b + d_0)p_c^*(z), \\ &= (y_c^*(z)p_a^*(z)p_b^*(z^{-1}) \\ &\quad - \bar{x}_0(Iz - T_b)^{-1}h_b + d_0)p_c^*(z). \end{aligned} \quad (32)$$

The above identity is the same as (18) with the choice of $u^*(z) = \bar{x}_0 \text{adj}(Iz - T_b)h_b - d_0 \det(Iz - T_b)$. In other words, if the system \mathcal{B} can be forced to have an initial condition x_0 and if a unit impulse term $d_0\delta(k)$ is added to the same system's output in such a manner that the feedback system and in particular $y_c^*(z)$ stays stable, then it is true that

$$y_c^*(z) = p_q^*(z). \quad (33)$$

From now on, we will use $p_q(k)$ instead of $y_c(k)$. The next step is to find x_0 and d_0 . For this purpose, we first need to write the inputs to each system as a function of individual system states. As a first step, let us begin with the input $u_a(k), k \geq 0$ to the system \mathcal{A} :

$$\begin{aligned} u_a(k) &= x_c(k)h_c + u_c(k)d_c, \\ &= x_c(k)h_c + (-x_b(k)h_b + u_b(k)d_b + \delta(k)d_0)d_c, \\ &= x_c(k)h_c - x_b(k)h_bd_c \\ &\quad + (x_a(k)h_a + u_a(k)d_a)d_b d_c + \delta(k)d_0d_c, \\ &= x_a(k)h_a d_b d_c d_{abc} - x_b(k)h_b d_c d_{abc} \\ &\quad + x_c(k)h_c d_{abc} + \delta(k)d_0d_c d_{abc}, \end{aligned} \quad (34)$$

where

$$d_{abc} = (1 - d_a d_b d_c)^{-1}. \quad (35)$$

One can similarly use the same arithmetic for $u_b(k)$ and $u_c(k)$ to be able to write the following expressions:

$$\begin{aligned} u_b(k) &= x_a(k)h_a d_{abc} - x_b(k)h_b d_c d_a d_{abc} \\ &\quad + x_c(k)h_c d_a d_{abc} + \delta(k)d_0d_c d_a d_{abc}, k \geq 0, \\ u_c(k) &= x_a(k)h_a d_b d_{abc} - x_b(k)h_b d_{abc} \\ &\quad + x_c(k)h_c d_a d_b d_{abc} + \delta(k)d_0d_{abc}, k \geq 0. \end{aligned} \quad (36)$$

We are now ready to write the state evolution equations (19), (21), and (23), in terms of only the system states. Defining the extended state for the feedback configuration

$$x_f(k) = [x_a(k) \quad x_b(k) \quad x_c(k)], \quad (37)$$

it is not difficult to show for $k \geq 0$ that

$$x_f(k+1)E_f = x_f(k)T_f + \delta(k)d_0v_f, \quad (38)$$

$$p_q(k) = x_f(k)h_f + \delta(k)d_0d_f, \quad (39)$$

where

$$E_f = \begin{bmatrix} I & -h_a d_{abc} v_b & 0 \\ 0 & T_b + h_b d_c d_a d_{abc} v_b & 0 \\ 0 & -h_c d_a d_{abc} v_b & I \end{bmatrix}, \quad (40)$$

$$T_f = \begin{bmatrix} T_a + h_a d_b d_c d_{abc} v_a & 0 & h_a d_b d_{abc} v_c \\ -h_b d_c d_{abc} v_a & I & -h_b d_{abc} v_c \\ h_c d_{abc} v_a & 0 & T_c + h_c d_a d_b d_{abc} v_c \end{bmatrix}, \quad (41)$$

$$v_f = [d_c d_{abc} v_a \quad 0 \quad d_{abc} v_c], \quad (42)$$

$$h_f = \begin{bmatrix} h_a d_b d_c d_{abc} \\ -h_b d_c d_{abc} \\ h_c d_{abc} \end{bmatrix}, \quad (43)$$

$$d_f = d_c d_{abc}, \quad (44)$$

and

$$x_f(0) = [0 \quad x_0 \quad 0].$$

We also define

$$T_{f,b} = [-h_b d_c d_{abc} v_a \quad I \quad -h_b d_{abc} v_c], \quad (45)$$

and

$$h_{f,b} = -h_b d_c d_{abc}. \quad (46)$$

Now let

$$z_f(k) = x_f(k+1), k \geq 0, \quad (47)$$

then we have the following autonomous system for $k \geq 0$

$$z_f(k+1)E_f = z_f(k)T_f, \quad (48)$$

$$p_q(k+1) = z_f(k)h_f. \quad (49)$$

Recalling $z_f(0) = x_f(1)$ and using (38) we first have

$$z_f(0)E_f = x_0 T_{f,b} + d_0 v_f. \quad (50)$$

We also note that the matrix pair (E, A) has $m_a + m_c$ generalized eigenvalues in Δ^i , one eigenvalue at $\lambda = 1$, and the remaining $m_b - 1$ in Δ^o including the ones at infinity. Then let Q_f and Z_f be orthogonal matrices such that we obtain a suitable ordered generalized real Schur form of the matrix pair (E_f, A_f) :

$$Q_f^T E_f Z_f = \begin{bmatrix} E_{f,oo} & E_{f,oi} \\ 0 & E_{f,ii} \end{bmatrix}, Q_f^T A_f Z_f = \begin{bmatrix} A_{f,oo} & A_{f,oi} \\ 0 & A_{f,ii} \end{bmatrix}, \quad (51)$$

where the ordering is done such that the generalized eigenvalues of the matrix pair $(E_{f,oo}, A_{f,oo})$ are exactly the same as those of the matrix pair (E, A) outside the unit disk and also including the one at $\lambda = 1$. Therefore the matrices $E_{f,oo}$ and $A_{f,oo}$ are of size m_b . Applying the transformation

$$\tilde{z}_f(k) = z_f(k)Q_f,$$

and partitioning $\tilde{z}_f(k)$ suitably as

$$\tilde{z}_f(k) = [\tilde{z}_{f,o}(k) \quad \tilde{z}_{f,i}(k)],$$

we conclude that for $k \geq 0$

$$\tilde{z}_{f,o}(k+1)E_{f,oo} = \tilde{z}_{f,o}(k)A_{f,oo}. \quad (52)$$

However, in order for $\tilde{z}_{f,o}(k)$ to be summable, the only possibility is that

$$\tilde{z}_{f,o}(0) = 0, \quad (53)$$

since otherwise $\sum_{k=0}^{\infty} p_q(k)$ can never be one. Partitioning

$$Q_f = [Q_{f,o} \quad Q_{f,i}], \quad (54)$$

the identity (53) reduces to

$$z_f(0)Q_{f,o} = 0 \quad (55)$$

Finally by (51), we show that

$$\tilde{z}_{f,i}(k+1)E_{f,ii} = \tilde{z}_{f,i}(k)A_{f,ii}. \quad (56)$$

Since the generalized eigenvalues of the pair $(E_{f,ii}, A_{f,ii})$ lie inside the unit disk Δ^i , $E_{f,ii}$ is nonsingular. Therefore we can rewrite (56)

$$\begin{aligned} \tilde{z}_{f,i}(k) &= \tilde{z}_{f,i}(0)T^k, \\ &= z_f(0)Q_{f,i}T^k, \end{aligned} \quad (57)$$

where

$$T = A_{f,ii}E_{f,ii}^{-1}. \quad (58)$$

Moreover, for $k \geq 0$

$$\begin{aligned} p_q(k+1) &= z_f(k)h_f, \\ &= \tilde{z}_{f,i}(k)Q_{f,i}^T h_f, \\ &= z_f(0)Q_{f,i}T^k Q_{f,i}^T h_f. \end{aligned} \quad (59)$$

Finally, the limiting probabilities should add up to 1 which gives us the final normalization equation:

$$\begin{aligned} 1 &= p_q(0) + \sum_{k=0}^{\infty} p_q(k+1), \\ &= x_0 h_{f,b} + d_0 d_f + z_f(0)Q_{f,i}(I-T)^{-1}Q_{f,i}^T h_f. \end{aligned} \quad (60)$$

Combining the equations (50),(55), and (60), we obtain a matrix equation of size $m_a + 2m_b + m_c + 1$ in the extended unknown vector

$$y = [z_f(0) \quad x_0 \quad d_0] \quad (61)$$

as

$$y \begin{bmatrix} E_f & Q_{f,o} & Q_{f,i}(I-T)^{-1}Q_{f,i}^T h_f \\ -T_{f,b} & 0 & h_{f,b} \\ -v_f & 0 & d_f \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^T. \quad (62)$$

Solving the linear matrix equation (62) for the unknowns $z_f(0)$, x_0 , and d_0 , we obtain the parameters of the matrix geometric distribution of the limiting distribution q :

$$p_q(k) = P(q = k) = \begin{cases} vT^{k-1}h, & k \geq 1 \\ d, & k = 0 \end{cases} \quad (63)$$

where $v = z_f(0)Q_{f,i}$, $h = h_f$, and $d = x_0 h_{f,b} + d_0 d_f$. Although the development of the overall algorithm might be elaborate, the algorithm itself is relatively simple and is given in Table 1 for the sake of reference.

Table 1: Algorithm to find $p_q(k), k \geq 0$ given the triple of quadruples (v_a, T_a, h_a, d_a) , (v_b, T_b, h_b, d_b) , and (v_c, T_c, h_c, d_c) characterizing the MG-distributed processes $a(n)$, $b(n)$, and $c(n)$, respectively, in the Lindley equation (1)

1. Define d_{abc} via (35).
2. Define E_f , T_f , v_f , h_f , and d_f , as in (40),(41),(42), (43), and (44), respectively.
3. Define $T_{f,b}$ and $h_{f,b}$ as in (45) and (46), respectively.
4. Obtain the matrices Q_f and Z_f that put the matrix pair (E_f, A_f) into the generalized ordered real Schur form with ordering described as in (51).
5. Partition Q_f as in (54).
6. Define T through (58).
7. Solve the linear matrix equation (62) for the unknown vector y .
8. Obtain the parameters of the matrix geometric distribution for the random variable q via (63).

4. NUMERICAL EXAMPLES

We first take a classical GI/GI/1 discrete time queueing example from [12]. The interarrival times are denoted by $b(n)$ and its distribution is uniform in the interval $[1, B_{max}]$.

The service times are denoted by $a(n)$ and this distribution is also uniform in the interval $[1, A_{max}]$. Such finite support distributions are actually discrete phase-type and phase-type representations for finite support distributions can easily be found as in [3]. The waiting time in the queue denoted by $q(n)$ satisfies the Lindley equation (1) with $c(n) = 0$ and we are interested in finding the equilibrium distribution. The proposed algorithm in Table 1 can now be used by taking $(v_c, T_c, h_c, d_c) = (\emptyset, \emptyset, \emptyset, 1)$ where \emptyset denotes the empty set. We compare the findings of this paper with those presented in [12] using the iterative algorithm for Wiener Hopf factorization. We write $E[q]$, $Var[q]$, and $p_q(\cdot)$ evaluated at various points for the case of $A_{max} = 50$ and $B_{max} = 25$ which corresponds to a utilization $\rho = E[a(n)]/E[b(n)] = 0.5098$ in Table 2. The proposed algorithm is named as GORSRD due to its numerical engine based on the generalized ordered real Schur decomposition. We use MATLAB 7.0 and its functions `qz.m` for the generalized real Schur form and `ordqz.m` for its suitable ordering. The results agree up to three digits. **table**

Table 2: The numerical results obtained with two algorithms proposed for a discrete time queueing example with $A_{max} = 50$ and $B_{max} = 25$.

Performance Measure	[12]	GORSRD
$E[q]$	3.720	3.7199
$Var[q]$	58.57	58.5738
$p_q(0)$	0.687	0.6871
$p_q(1)$	0.019	0.0193
$p_q(2)$	0.019	0.0187
$p_q(3)$	0.018	0.0182
$p_q(4)$	0.017	0.0176
$p_q(5)$	0.017	0.0170
$p_q(6)$	0.016	0.0164
$p_q(10)$	0.014	0.0138
$p_q(20)$	0.007	0.0066
$p_q(30)$	0.002	0.0018

We also vary the values of A_{max} and B_{max} to see if we can obtain a solution for large problems and with close to unity utilizations. Table 3 presents our numerical results. Although the results agree, the computational complexity of the GORSRD algorithm is much higher than that of [12] for finite support large GI/G/1 queues. The reason is that GORSRD applies on matrix pairs of size $m = m_a + m_b$ (since $m_c = 0$ in this example) and the computational complexity of the generalized ordered real Schur decomposition is $O(m^3)$. On the other hand, the algorithm provided in reference [12] requires $4m_a m_b$ operations in each iteration and is therefore computationally more efficient for the finite support case.

The real advantage of using GORSRD remains to be seen when infinite support matrix geometric distributions are to be used when such distributions are either available or used as a means of approximating empirical data. For this purpose, we use a negative binomial distribution x with parameters r and p whose PMF is then of the following form [8]:

$$p_x(k) = \binom{r+k-1}{k} p^r (1-p)^k, k \geq 0.$$

Table 3: The numerical results using two algorithms for a discrete time queueing example with varying A_{max} and B_{max} .

A_{max}	B_{max}	ρ	$E[q]$ ([12])	$E[q]$ (GORSD)
50	25	0.5098	3.720	3.7199
50	40	0.8039	24.9	24.8725
49	48	0.980	381.0	380.9885
66	65	0.9851	700.1	700.1239
99	98	0.99	1596.6	1594.6
200	180	0.9005	260.4	260.4388
500	450	0.9002	651.1	651.1151
1000	900	0.9001	1302.2	1302.2

The PGF of the distribution x is then easy to write

$$p_x^*(z) = \left(\frac{pz}{z - (1-p)} \right)^r$$

with $E[x] = \frac{r(1-p)}{p}$ and $Var[x] = \frac{r(1-p)}{p^2}$. The parameter r gives the order of the distribution and given r and $E[x]$, one can fit a negative binomial distribution by setting $p = r/(r + E[x])$. In this numerical example, we assume that both processes $a(n)$ and $b(n)$ possess negative binomial distributions with the same but varying r parameter. Moreover, $E[b] = 10$ and $E[a] = \rho E[b]$ for a load parameter $\rho < 1$. Obviously, negative binomial distributions are infinite support and the iterative technique of [13] cannot be directly used. However, one can find a straightforward finite support distribution approximation $p_x^\varepsilon(\cdot)$ for a desired accuracy ε to an infinite support distribution $p_x(\cdot)$ by

$$p_x^\varepsilon(k) = \begin{cases} \frac{p_x(k)}{\sum_{k=K_x^\varepsilon+1}^{\infty} p_x(k)} & k \leq K_x^\varepsilon, \\ 0 & k > K_x^\varepsilon, \end{cases}$$

where K_x^ε is the smallest integer such that $\sum_{k=0}^{K_x^\varepsilon} p_x(k) > 1 - \varepsilon$. Once the finite support approximation is obtained, one can use the iterations in [13] for finding an approximation to the limiting distribution q in the Lindley equation (1). Given a desired accuracy ε , the computational cost for the algorithm of [13] for the Lindley equation in (1) is $O(K_a^\varepsilon K_b^\varepsilon)$ per iteration. Let *ITER* denote the number of iterations required for a desired accuracy of 10^{-9} . On the other hand, our proposed algorithm uses the matrix geometric representation for the negative binomial distribution that corresponds to the sum of r geometric random variables each with PMF of the form $p(1-p)^k, k \geq 0$. Such a representation is easily obtainable using the method described in (15). It is also clear that the computational complexity of the proposed GORS method in this case is $O(r^3)$ and we expect to achieve relatively efficient solutions with GORS when the parameter r is small. We present our results in terms of $p_q(0)$, $E[q]$, and the CPU time in seconds in Table 4 for two different values of r and for two different values of ρ . The results are obtained using MATLAB 7.0 running on a Linux Redhat Intel XEON-based workstation. When the approximation accuracy parameter $\varepsilon \rightarrow 0$ then the results using the algorithm in [13] exactly match with those obtained from GORS up to 4 digits. We did not encounter any stability problems in using the two algorithms for any parameter set. We also observed that increasing the load to 0.99 did not have any adverse effect on both algorithms. When the distribution parameter r is 4, then $r^3 \ll K_a^\varepsilon K_b^\varepsilon$ and we obtain

relatively efficient results with GORS. On the other hand, when $r = 64$ then $r^3 \gg K_a^\varepsilon K_b^\varepsilon$ and the algorithm in [13] is more efficient in terms of CPU time. Therefore, we are led to believe that GORS is a viable alternative for discrete time queues especially driven by MG-distributed infinite support distributions with moderate order.

5. CONCLUSIONS

In this paper, we introduce a new algorithmic method for solving the limiting distribution of discrete time Lindley's equation which is driven by processes which are not auto-correlated and which are modeled by general matrix geometric distributions. We use state space methods of system theory which are purely matrix analytical. We avoid transform domain calculations such as root finding or inverse z-transformation. The numerical engine of the proposed algorithm is the generalized ordered real Schur decomposition which is available in various platforms for public use and which has been effectively used to solve Riccati equations for decades. This decomposition is to apply on a matrix pair with size being the sum of the order of the individual matrix geometric-distributed processes as opposed to their product, which is a clear advantage over some of the matrix analytical methods proposed for discrete time queues. However, we note the real advantage of the proposed algorithm of this paper is when the involved matrix geometric distributions have infinite support or they can be approximated well by infinite support matrix geometric distributions with much smaller orders. Another advantage of the proposed algorithm is the matrix geometric representation of the limiting distribution via which moments or any related quantity can easily be found. We plan on extending our results to semi-Markov arrivals and services and also studying matrix geometric modeling using empirical data.

Acknowledgments

This work was supported in part by the Scientific and Technical Research Council of Turkey (TÜBİTAK) under projects EEEAG-101E025, EEEAG-105E065, and EEEAG-106E046.

6. REFERENCES

- [1] N. Akar and K. Sohraby. An invariant subspace approach in M/G/1 and G/M/1 type Markov chains. *Commun. Statist. - Stochastic Models*, 13(3):381–416, 1997.
- [2] N. Akar and K. Sohraby. Matrix-geometric solutions in m/g/1 type markov chains with multiple boundaries: a generalized state-space approach. In *Proc. ITC'15*, pages 487–496, 1997.
- [3] A. S. Alfa. Combined elapsed time and matrix-analytic method for the discrete time GI/G/1 and GI^X/G/1 systems. *Queueing Systems*, 45:5–25, 2003.
- [4] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. D. Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK's User's Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, third edition, 1999.
- [5] Z. Bai and J. Demmel. Inverse free parallel spectral divide and conquer algorithms for nonsymmetric eigenproblems. Computer Science Division Report

Table 4: The two performance measures $p_q(0)$ and $E[q]$, and the CPU time in seconds in for $r = 4, 64$ and for $\rho = 0.6, 0.99$ using the two algorithms

r	ρ	Algorithm in [13]							GORSD		
		ε	$ITER$	K_a^ε	K_b^ε	$p_q(0)$	$E[q]$	CPU time	$p_q(0)$	$E[q]$	CPU time
4	0.6	10^{-2}	14	18	28	$6.2047 \cdot 10^{-1}$	2.377	$3.9 \cdot 10^{-2}$	$6.1000 \cdot 10^{-1}$	2.647	$8.3 \cdot 10^{-4}$
		10^{-5}	13	35	54	$6.1002 \cdot 10^{-1}$	2.646	$1.3 \cdot 10^{-1}$			
		10^{-8}	13	50	77	$6.1000 \cdot 10^{-1}$	2.647	$2.3 \cdot 10^{-1}$			
4	0.99	10^{-2}	19	28	28	$1.6938 \cdot 10^{-2}$	342.4	$6.6 \cdot 10^{-2}$	$1.7988 \cdot 10^{-2}$	341.9	$8.6 \cdot 10^{-4}$
		10^{-5}	20	53	54	$1.8002 \cdot 10^{-2}$	341.5	$2.8 \cdot 10^{-1}$			
		10^{-8}	18	76	77	$1.7988 \cdot 10^{-2}$	341.9	$4.8 \cdot 10^{-1}$			
64	0.6	10^{-2}	10	13	19	$8.1930 \cdot 10^{-1}$	0.544	$1.5 \cdot 10^{-2}$	$8.1285 \cdot 10^{-1}$	0.592	$1.2 \cdot 10^{-1}$
		10^{-5}	10	20	28	$8.1287 \cdot 10^{-1}$	0.592	$2.4 \cdot 10^{-2}$			
		10^{-8}	10	26	35	$8.1285 \cdot 10^{-1}$	0.592	$3.5 \cdot 10^{-2}$			
64	0.99	10^{-2}	19	19	19	$3.1498 \cdot 10^{-2}$	112.3	$3.5 \cdot 10^{-2}$	$3.2427 \cdot 10^{-2}$	112.2	$1.1 \cdot 10^{-1}$
		10^{-5}	18	28	28	$3.2424 \cdot 10^{-2}$	112.2	$6.1 \cdot 10^{-2}$			
		10^{-8}	18	35	35	$3.2427 \cdot 10^{-2}$	112.2	$9.7 \cdot 10^{-2}$			

CSD-94-793, University of California at Berkeley, Feb. 1994.

[6] D. Bini and B. Meini. On the solution of a nonlinear matrix equation arising in queueing problems. *SIAM Jour. on Matrix. Analy. Appl.*, 17:906–926, 1996.

[7] H. Bruneel and B. G. Kim. *Discrete-Time Models for Communication Systems Including ATM*. Kluwer Academic Publishers, Norwell, MA, USA, 1992.

[8] G. Casella and R. L. Berger. *Statistical Inference*. Wadsworth and Brooks/Cole, Belmont, CA, 1990, 1990.

[9] M. Chaudhry and U. C. Gupta. Computing waiting-time probabilities in the discrete-time queue: GI^X/G/1. *Perf. Eval.*, 43:123–131, 2001.

[10] J. Demmel and B. Kagstrom. The generalized Schur decomposition of an arbitrary pencil $A - zB$: Robust software with error bounds and applications, Part I - theory and algorithms. *ACM Trans. Math. Softw.*, 19(2):160–174, 1993.

[11] H. R. Gail, S. L. Hantler, and P. G. Taylor. M/G/1 and G/M/1 type Markov chains with multiple boundary levels. *Adv. appl. Prob.*, 29(3):733–758, 1997.

[12] W. K. Grassmann and J. L. Jain. Numerical solutions of the waiting time distribution and idle time distribution of the arithmetic GI/G/1 queue. *Operations Research*, 37(1):141–149, 1989.

[13] G. Hasslinger. Waiting time, busy periods and output models of a server analyzed via WienerHopf factorization. *Performance Evaluation*, 40:3–26, 2000.

[14] J. H. Hunter. *Mathematical Techniques of Applied Probability Vol. 2 Discrete Time Models: Techniques and Applications*. Academic Press, 1983.

[15] W. F. A. III and A. J. Laub. Generalized eigenproblem algorithms and software for algebraic riccati equations. *Proc. of IEEE*, 72:1746–1754, 1984.

[16] T. Kailath. *Linear Systems*. Prentice Hall, 1980.

[17] G. Latouche and V. Ramaswami. A logarithmic reduction algorithm for quasi-birth-death processes. *J. Appl. Prob.*, 30:650–674, 1993.

[18] D. V. Lindley. The theory of queues in a single server queue. *Proc. Cambridge Philos. Soc.*, 48:277–289, 1952.

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

[20] D. Luenberger. Dynamic equations in descriptor form. *IEEE Trans. Auto. Contr.*, 22(3):312–321, 1977.

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

[22] V. Naoumov, U. Krieger, and D. Wagner. Analysis of a multi-server delay-loss system with a general Markovian arrival process. In S. Chakravathy and A. Alfa, editors, *Matrix-analytical methods in Stochastic models*, pages 43–66. Marcel Dekker, 1997.

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

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

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

[26] A. Riska and 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*, pages 86–96, New York, NY, USA, 2002. ACM Press.

[27] A. Riska, E. Smirni, and G. Ciardo. Exact analysis of a class of GI/G/1-type performability models. *IEEE Trans. Reliability*, 53(2):238–249, 2004.

[28] K. Wuyts, B. van Houdt, R. B. R., and C. Blondia. Matrix geometric analysis of discrete-time queues with batch arrivals and batch departures with applications to B-ISDN. In *International Teletraffic Congress*, 1999.

[29] T. Yang and M. Chaudhry. On the steady-state queue size distributions of the discrete-time GI/G/1 queue. *Adv. Appl. Prob.*, 28:1177–1200, 1996.

APPENDIX

We'll provide a proof for Theorem 1 here. Consider double-sided z -transforms [25] and recall that right-sided PMFs $p(k)$, i.e., $p(k) = 0, k < 0$ possess stable z -transforms, i.e., their poles are in Δ^o . On the other hand, left-sided PMFs $p(k)$, i.e., $p(k) = 0, k > 0$, have anti-stable z -transforms, i.e., their poles are in Δ^i . The $\max(\cdot, 0)$ operator in the space domain then corresponds to taking the anti-stable part out in the transform domain and adding all the probability mass of the anti-stable part to the origin. Note that the PMF of the random variable $s = x - y$ denoted by $p_z(k)$ is double-sided and its PGF is expressed as

$$p_s^*(z) = p_x^*(z)p_y^*(z^{-1}). \quad (64)$$

We know that

$$p_x^*(z) = q_x^*(z) + \frac{l_x^*(z)}{d_x^*(z)}, \deg(l_x^*) < \deg(d_x^*), \quad (65)$$

for a polynomial $q_x^*(z)$ simply by Euclidean division. Rewriting $p_s^*(z)$, we have

$$p_s^*(z) = q_x^*(z) \frac{\bar{n}_y^*(z)}{d_y^*(z)} + \frac{l_x^*(z)}{d_x^*(z)} \frac{\bar{n}_y^*(z)}{d_y^*(z)}. \quad (66)$$

Using Euclidean division for the first term and unique spectral decomposition for the strictly proper second term, we find out that the anti-stable part of $p_s^*(z)$ is a strictly proper rational function of the form $\frac{u^*(z)}{d_y^*(z)}$. Therefore by taking out the anti-stable part and by adding the corresponding probability mass to the origin, we obtain

$$\begin{aligned} p_i^*(z) &= p_s^*(z) - \frac{u_1^*(z)}{d_y^*(z)} + \frac{u_1^*(1)}{d_y^*(1)}, \\ &= p_x^*(z)p_y^*(z^{-1}) - \frac{u^*(z)}{d_y^*(z)}, \end{aligned} \quad (67)$$

where

$$u^*(z) = u_1^*(z) - \frac{u_1^*(1)}{d_y^*(1)} \bar{d}_y^*(z).$$

Note that $u^*(1) = 0$ and $\deg(u^*) = \deg(\bar{d}_y^*) = \deg(n_y^*)$, which completes the if part of the proof. The only if part can be proved by observing the unique spectral decomposition of a strictly proper rational function into its stable and anti-stable parts and tracing back the proof of the if part.