ETAQA Solutions for Infinite Markov Processes with Repetitive Structure

Alma Riska
Seagate Research, 1251 Waterfront Place, Pittsburgh, Pennsylvania 15222, USA, alma.riska@seagate.com
Evgenia Smirni
Department of Computer Science, College of William and Mary, Williamsburg, Virginia 23187-8795, USA, esmirni@cs.wm.edu

We describe the ETAQA approach for the exact analysis of M/G/1 and GI/M/1-type processes, and their intersection, i.e., quasi-birth-death processes. ETAQA exploits the repetitive structure of the infinite portion of the chain and derives a finite system of linear equations. In contrast to the classic techniques for the solution of such systems, the solution of this finite linear system *does not* provide the entire probability distribution of the state space but simply allows for the calculation of the aggregate probability of a finite set of classes of states from the state space, appropriately defined. Nonetheless, these aggregate probabilities allow for the computation of a rich set of measures of interest such as the system queue length or any of its higher moments. The proposed solution approach is exact and, for the case of M/G/1-type processes, it compares favorably to the classic methods as shown by detailed time and space complexity analysis. Detailed experimentation further corroborates that ETAQA provides significantly less expensive solutions when compared to the classic methods.

Key words: M/G/1-type processes; GI/M/1-type processes; quasi-birth-death processes; computer system performance modeling; matrix-analytic methods; Markov chains
History: Accepted by Ed Kao, Area Editor for Computational Probability & Analysis;

---

1. Introduction

Matrix analytic techniques, pioneered by Neuts (1981, 1989) provide a framework that is widely used for the exact analysis of a general and frequently encountered class of queuing models. In these models, the embedded Markov chains are two-dimensional generalizations of elementary GI/M/1 and M/G/1 queues (Kleinrock (1975)), and their intersection, i.e., quasi-birth-death (QBD) processes. GI/M/1 and M/G/1 queues model systems with inter arrival and service times characterized, respectively, by *general* distributions rather than simple exponentials and are often used as the modeling tool of choice in modern computer and communication systems (Nelson (1995); Ramaswami and Wang (1996); Squillante (1998, 2000)). As a consequence, various analytic methodologies for their solution have been developed in Neuts (1989); Latouche (1993); Latouche and Stewart (1995); Meini (1998); Grassman and Stanford (2000).

In this paper, we present ETAQA, an analytic solution technique for the exact analysis of M/G/1-type, GI/M/1-type Markov chains, and their intersection, i.e., quasi-birth-death processes. Neuts (1981) defines various classes of infinite-state Markov chains with a repetitive structure. In all cases, the state space $\mathcal{S}$ is partitioned into the boundary states $\mathcal{S}^{(0)} = \{s_1^{(0)}, \ldots, s_m^{(0)}\}$ and the sets of states representing each repetitive level $\mathcal{S}^{(j)} = \{s_1^{(j)}, \ldots, s_n^{(j)}\}$, for $j \geq 1$. For the M/G/1-type Markov chains, the infinitesimal generator $\mathbf{Q}_{M/G/1}$ has upper block Hessenberg form and Neuts (1989) proposes matrix analytic methods for their solution. The key in the matrix-analytic solution is the computation of an auxiliary matrix called $\mathbf{G}$. Similarly, for Markov chains of the GI/M/1-type, the infinitesimal generator has a lower block Hessenberg form, and Neuts (1981) proposes the very elegant matrix-geometric for their solution. QBD processes with a block tri-diagonal infinitesimal generator can be solved using either methodology, but matrix geometric is the prefered one (see Latouche and Ramaswami (1999)).

The traditional matrix-analytic algorithms are developed based on the concept of stochastic complementation, as explained in Riska and Smirni (2002b) and provide a recursive function for the computation of the probability vector $\boldsymbol{\pi}^{(i)}$ that corresponds to $\mathcal{S}^{(j)}$, for $j \geq 1$. This recursive function is based on $\mathbf{G}$ (for the case of M/G/1-type processes) or $\mathbf{R}$ (for the case of GI/M/1-type processes). Iterative procedures are used for determining $\mathbf{G}$ or $\mathbf{R}$ (see details in Latouche (1993); Meini (1998)). For more details on stochastic complementation and its application for the development of the matrix analytic algorithms, we direct the interested reader to Riska and Smirni (2002b).

ETAQA (which stands for the Efficient Technique for the Analysis of QBD processes by Aggregation) was first introduced in Ciardo and Smirni (1999) for the solution of a *limited* class of QBD processes. This limited class allowed the return from level $\mathcal{S}^{(j+1)}$ to level $\mathcal{S}^{(j)}$, $j \geq 1$, to be directed towards a single state only. This same result was extended in Ciardo et al. (2004) for the solution of M/G/1-type processes with the same restriction, i.e., returns from any higher level $\mathcal{S}^{(j+1)}$ in the Markov chain to its lower level $\mathcal{S}^{(j)}$ have to be directed to a single state only.

Figure 1: Aggregation of an Infinite $\mathcal{S}$ into a Finite Number of Classes of States

In this paper, we adapt the ETAQA approach for the solution of *general* processes of the M/G/1-type, GI/M/1-type, as well as QBDs, i.e., we relax the above strong assumption of returns to a single state only, and provide a general solution approach that works for *any* type of returns to the lower level, i.e., transitions from any state in level $\mathcal{S}^{(j+1)}$ to any state in level $\mathcal{S}^{(j)}$, $j \geq 1$ are allowed. In contrast to the matrix-analytic techniques for solving M/G/1-type and GI/M/1-type processes that use a recursive function for the computation of the probability vectors of each level, ETAQA uses a different treatment: it constructs and solves a finite linear system of $m + 2n$ unknowns, where m is the number of states in the boundary portion of the process and n is the number of states in each of the repetitive "levels" of the state space, and obtain exact solution. Instead of evaluating the stationary probability distribution of *all* states in each of the repetitive levels $\mathcal{S}^{(j)}$ of the state space $\mathcal{S}$, we calculate the *aggregate* stationary probability distribution of n classes of states $\mathcal{T}^{(i)}$, $1 \leq i \leq n$, appropriately defined (see Figure 1). This approach could be perceived as similar to lumpability since an aggregate probability distrbution is computed, or perhaps also stochastic complementation. We stress that the finite system of $m+2n$ linear equations that ETAQA provides is not an infinitesimal generator, so the aggregation of the infinite set $\mathcal{S}$ into a finite number of classes of states does not result to a Markov chain, thus it cannot be considered similar to any lumpability of stochastic complementation techniques.

Yet, the computation of the aggregate probability distribution that we compute with our method is *exact*. Furthermore, this aggregate probability distribution does provide the means for calculating a variety of measures of interest including the expected queue length and any of its higher moments. Although ETAQA does not allow for the exact calculation of the queue length distribution, it provides the means to compute the coefficient of variation (i.e., via the second moment) as well as the skewness of the distribution (i.e., via the third moment), which in turn provide further information about the queuing behavior of the system.

ETAQA results in significantly more efficient solutions than the traditional methods for the M/G/1-type processes. For the case of QBD and GI/M/1-type processes, ETAQA results in solutions that are as efficient as the classic ones. We provide detailed big-O complexity analysis of ETAQA and the most efficient alternative methods. These results are further corroborated via detailed experimentation.

An additional important issue that arises is related to the numerical stability of the method, especially for the case of M/G/1-type processes. Riska and Smirni (2002a), a preliminary version of this paper that focused on M/G/1-type processes only, provides experimental indications that the method is numerically stable. Here, we do not focus on the numerical stability issue, but we instead illustrate that the method generalizes to the solution of M/G/1-type, GI/M/1-type, and QBD processes of any type. The numerical stability of ETAQA and its connection to matrix-analytic methods is explored formally in Stathopoulos et al. (2005), where ETAQA's numerical stability is proven and shown to often be superior to the alternative matrix-analytic solutions.

This paper is organized as follows. In Section 2 we outline the matrix analytic methods for the solution of M/G/1-type, GI/M/1-type, and QBD processes. ETAQA, along with detailed time and storage complexity analysis for the solution of M/G/1-type, GI/M/1-type, and QBD processes is presented in Sections 3, 4, and 5, respectively. We experimentally compare its efficiency with the best known methods in a set of realistic examples (see Section 6) for the case of M/G/1-type processes. Finally, we summarize our findings and report on ETAQA's efficiency in Section 7.

2. Background

In this paper, we assume continuous time Markov chains, or CTMCs, hence we refer to the infinitesimal generator $\mathbf{Q}$, but our discussion applies just as well to discrete time Markov chains, or DTMCs. Neuts (1981) defines various classes of infinite-state Markov chains with a repetitive structure. In all cases, the state space $\mathcal{S}$ is partitioned into the boundary states $\mathcal{S}^{(0)} = \{s_1^{(0)}, \ldots, s_m^{(0)}\}$ and the sets of states $\mathcal{S}^{(j)} = \{s_1^{(j)}, \ldots, s_n^{(j)}\}$, for $j \geq 1$, while $\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(j)}$, are the stationary probability vectors for states in $\mathcal{S}^{(0)}$ and $\mathcal{S}^{(j)}$, for $j \geq 1$.

2.1. M/G/1-type Processes

For the class of M/G/1-type Markov chains, the infinitesimal generator $\mathbf{Q}_{M/G/1}$ is block-partitioned as:

$$\mathbf{Q}_{M/G/1} = \begin{bmatrix} \widehat{\mathbf{L}} & \widehat{\mathbf{F}}^{(1)} & \widehat{\mathbf{F}}^{(2)} & \widehat{\mathbf{F}}^{(3)} & \widehat{\mathbf{F}}^{(4)} & \cdots \\ \widehat{\mathbf{B}} & \mathbf{L} & \mathbf{F}^{(1)} & \mathbf{F}^{(2)} & \mathbf{F}^{(3)} & \cdots \\ \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F}^{(1)} & \mathbf{F}^{(2)} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F}^{(1)} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{1}$$

We use the letters "L", "F", and "B" according to whether they describe "local", 'forward", and "backward" transition rates, respectively, in relation to a set of states $\mathcal{S}^{(j)}$ for $j \geq 1$, and a "$\widehat{\ }$" for matrices related to $\mathcal{S}^{(0)}$.

For the solution of M/G/1-type processes, several algorithms exist in Grassman and Stanford (2000); Bini et al. (2000); Meini (1998); Neuts (1989). These algorithms first compute the matrix $\mathbf{G}$ as the solution of the matrix equation:

$$\mathbf{B} + \mathbf{L}\mathbf{G} + \sum_{j=1}^{\infty} \mathbf{F}^{(j)}\mathbf{G}^{j+1} = \mathbf{0}. \tag{2}$$

The matrix $\mathbf{G}$, which is stochastic if the process is recurrent and irreducible, has an important probabilistic interpretation: an entry (k, l) in $\mathbf{G}$ expresses the conditional probability of the process first entering $\mathcal{S}^{(j-1)}$ through state l, given that it starts from state k of $\mathcal{S}^{(j)}$, as defined in Neuts (1989, page 81). Note that the probabilistic interpretation of $\mathbf{G}$ is the same for both DTMCs and CTMCs. The interpretation in Neuts (1989, page 81) is consistent with the discussion in Latouche and Ramaswami (1999, page 142), where CTMCs are taken into consideration. The $\mathbf{G}$ matrix is obtained by solving iteratively Eq.(2). However, recent advances show that the computation of $\mathbf{G}$ is more efficient when displacement structures are used based on the representation of M/G/1-type processes by means of QBD processes, as discussed in Meini (1998); Bini et al. (2000); Bini and Meini (1998); Latouche and Ramaswami (1999). The most efficient algorithm for the computation of $\mathbf{G}$ is the cyclic reduction algorithm proposed of Bini et al. (2000).

The calculation of the stationary probability vector is based on the recursive Ramaswami (1988)'s formula, which is numerically stable because it entails only additions and multiplications. Neuts (1989); Ramaswami (1988) suggest that subtractions on these type of formulas present the possibility of numerical instability. Ramaswami's formula defines the following recursive relation among stationary probability vectors $\boldsymbol{\pi}^{(j)}$ for $j \geq 0$:

$$\boldsymbol{\pi}^{(j)} = -\left(\boldsymbol{\pi}^{(0)}\widehat{\mathbf{S}}^{(j)} + \sum_{k=1}^{j-1} \boldsymbol{\pi}^{(k)}\mathbf{S}^{(j-k)}\right)\mathbf{S}^{(0)^{-1}} \quad \forall j \geq 1, \tag{3}$$

where $\widehat{\mathbf{S}}^{(j)}$ and $\mathbf{S}^{(j)}$ are defined as follows:

$$\widehat{\mathbf{S}}^{(j)} = \sum_{l=j}^{\infty} \widehat{\mathbf{F}}^{(l)}\mathbf{G}^{l-j}, \quad j \geq 1, \qquad \mathbf{S}^{(j)} = \sum_{l=j}^{\infty} \mathbf{F}^{(l)}\mathbf{G}^{l-j}, \quad j \geq 0 \ \ (\text{letting } \mathbf{F}^{(0)} \equiv \mathbf{L}). \tag{4}$$

Given the above definition of $\boldsymbol{\pi}^{(j)}$ and the normalization condition, a unique vector $\boldsymbol{\pi}^{(0)}$ can be obtained by solving the following system of m linear equations:

$$\boldsymbol{\pi}^{(0)} \left[\; \left(\widehat{\mathbf{L}} - \widehat{\mathbf{S}}^{(1)}\mathbf{S}^{(0)^{-1}}\widehat{\mathbf{B}}\right)^{\diamond} \; \middle| \; \mathbf{1}^T - \left(\sum_{j=1}^{\infty} \widehat{\mathbf{S}}^{(j)}\right)\left(\sum_{j=0}^{\infty} \mathbf{S}^{(j)}\right)^{-1} \mathbf{1}^T \; \right] = [\mathbf{0} \mid 1], \tag{5}$$

where the symbol "$\diamond$" indicates that we discard one (any) column of the corresponding matrix, since we added a column representing the normalization condition. Once $\boldsymbol{\pi}^{(0)}$ is known, we can then iteratively compute $\boldsymbol{\pi}^{(j)}$ for $j \geq 1$, stopping when the accumulated probability mass is close to one. After this point, measures of interest can be computed. Since the relation between $\boldsymbol{\pi}^{(j)}$ for $j \geq 1$ is not straightforward, computation of measures of interest requires generation of the whole stationary probability vector. For a limited set of measures of interest such as first and second moments of queue length, Lucantoni (1983) proposes closed-form formulas that do not require the knowledge of the entire vector $\boldsymbol{\pi}$. However these formulas are very complex.

Meini (1997b) gives an improved version of Ramaswami's formula. Once $\boldsymbol{\pi}^{(0)}$ is known using Eq.(5), the stationary probability vector is computed using matrix-generating functions associated with triangular Toeplitz matrices. A Toeplitz matrix has equal elements in each of its diagonals, which makes these type of matrices easier to handle than fully general matrices. These matrix-generating functions are computed efficiently using fast Fourier transforms (FFTs).

$$\begin{aligned} \tilde{\boldsymbol{\pi}}^{(1)} &= -\mathbf{b} \cdot \mathbf{Y}^{-1} \\ \tilde{\boldsymbol{\pi}}^{(i)} &= -\tilde{\boldsymbol{\pi}}^{(i-1)} \cdot \mathbf{Z}\mathbf{Y}^{-1} \quad i \geq 2, \end{aligned} \tag{6}$$

where $\tilde{\boldsymbol{\pi}}^{(1)} = [\boldsymbol{\pi}^{(1)}, ..., \boldsymbol{\pi}^{(p)}]$ and $\tilde{\boldsymbol{\pi}}^{(i)} = [\boldsymbol{\pi}^{(p(i-1)+1)}, ..., \boldsymbol{\pi}^{(pi)}]$ for $i \geq 2$. Matrices $\mathbf{Y}$, $\mathbf{Z}$, and $\mathbf{b}$ are defined as follows:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{S}^{(0)} & \mathbf{S}^{(1)} & \mathbf{S}^{(2)} & \cdots & \mathbf{S}^{(p-1)} \\ \mathbf{0} & \mathbf{S}^{(0)} & \mathbf{S}^{(1)} & \cdots & \mathbf{S}^{(p-2)} \\ \mathbf{0} & \mathbf{0} & \mathbf{S}^{(0)} & \cdots & \mathbf{S}^{(p-3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{S}^{(0)} \end{bmatrix}, \quad \mathbf{Z} = \begin{bmatrix} \mathbf{S}^{(p)} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} \\ \mathbf{S}^{(p-1)} & \mathbf{S}^{(p)} & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{S}^{(2)} & \mathbf{S}^{(3)} & \cdots & \mathbf{S}^{(p)} & \mathbf{0} \\ \mathbf{S}^{(1)} & \mathbf{S}^{(2)} & \cdots & \mathbf{S}^{(p-1)} & \mathbf{S}^{(p)} \end{bmatrix}, \quad \mathbf{b} = \boldsymbol{\pi}^{(0)} \begin{bmatrix} \widehat{\mathbf{S}}^{(1)} \\ \widehat{\mathbf{S}}^{(2)} \\ \widehat{\mathbf{S}}^{(3)} \\ \vdots \\ \widehat{\mathbf{S}}^{(p)} \end{bmatrix}^T,$$

where p is a constant that defines how many of matrices $\mathbf{S}^{(i)}$ and $\widehat{\mathbf{S}}^{(i)}$ are computed. In the above representation, the matrix $\mathbf{Y}$ is an upper block triangular Toeplitz matrix and the matrix $\mathbf{Z}$ is a lower block triangular Toeplitz matrix.

2.2. GI/M/1-type Processes

For the class of GI/M/1-type Markov chains, the infinitesimal generator $\mathbf{Q}_{GI/M/1}$ is block-partitioned as:

$$\mathbf{Q}_{GI/M/1} = \begin{bmatrix} \widehat{\mathbf{L}} & \widehat{\mathbf{F}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ \widehat{\mathbf{B}}^{(1)} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \mathbf{0} & \cdots \\ \widehat{\mathbf{B}}^{(2)} & \mathbf{B}^{(1)} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \cdots \\ \widehat{\mathbf{B}}^{(3)} & \mathbf{B}^{(2)} & \mathbf{B}^{(1)} & \mathbf{L} & \mathbf{F} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{7}$$

Key to the general solution of the generator in Eq.(7) is the fact that the following geometric relation holds among the stationary probability vectors $\boldsymbol{\pi}^{(j)}$ and $\boldsymbol{\pi}^{(1)}$ for states in $\mathcal{S}^{(j)}$:

$$\boldsymbol{\pi}^{(j)} = \boldsymbol{\pi}^{(1)} \cdot \mathbf{R}^{j-1}, \qquad \forall j \geq 1, \tag{8}$$

where $\mathbf{R}$ is the solution of the matrix equation

$$\mathbf{F} + \mathbf{R} \cdot \mathbf{L} + \sum_{k=1}^{\infty} \mathbf{R}^{k+1} \cdot \mathbf{B}^{(k)} = \mathbf{0}, \tag{9}$$

and can be computed using iterative numerical algorithms. Matrix $\mathbf{R}$, the geometric coefficient, has an important probabilistic interpretation: the entry (k, l) of $\mathbf{R}$ is the expected time spent in the state l of $\mathcal{S}^{(i)}$, before the first visit into $\mathcal{S}^{(i-1)}$, expressed in time unit Δ^i, given the starting state is k in $\mathcal{S}^{(i-1)}$. Δ^i is the mean sojourn time in the state k of $\mathcal{S}^{(i-1)}$ for $i \geq 2$, as defined in Neuts (1981, pages 30-35). Latouche (1993) describes several iterative numerical algorithms for computation of $\mathbf{R}$. Eq.(9) together with the normalization condition are then used to obtain $\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(1)}$ by solving the following system of $m + n$ equations:

$$[\boldsymbol{\pi}^{(0)},\ \boldsymbol{\pi}^{(1)}] \cdot \left[\begin{array}{c|c|c} \widehat{\mathbf{L}}^{\diamond} & \widehat{\mathbf{F}} & \mathbf{1}^T \\ (\sum_{k=1}^{\infty} \mathbf{R}^{k-1} \cdot \widehat{\mathbf{B}}^{(k)})^{\diamond} & \mathbf{L} + \sum_{k=1}^{\infty} \mathbf{R}^{k} \cdot \mathbf{B}^{(k)} & (\mathbf{I} - \mathbf{R})^{-1} \cdot \mathbf{1}^T \end{array} \right] = [\mathbf{0} \mid 1]. \tag{10}$$

For $k \geq 1$, $\boldsymbol{\pi}^{(k)}$ can be obtained numerically from Eq.(8). More importantly, useful performance metrics, such as the expected queue length, is computed exactly in explicit form:

$$\boldsymbol{\pi}^{(1)} \cdot (\mathbf{I} - \mathbf{R})^{-2} \cdot \mathbf{1}^T.$$

2.3. Quasi Birth-Death Processes

The intersection of GI/M/1-type and M/G/1-type processes is the special case of the quasi birth-death (QBD) processes, whose infinitesimal generator $\mathbf{Q}_{QBD}$ is of the block tri-diagonal form:

$$\mathbf{Q}_{QDB} = \begin{bmatrix} \widehat{\mathbf{L}} & \widehat{\mathbf{F}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ \widehat{\mathbf{B}} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{11}$$

While the QBD case falls under both the M/G/1 and the GI/M/1-type processes, it is most commonly associated with GI/M/1-type matrices because it can be solved using the very well-known matrix geometric approach introduced in Neuts (1981), (which we outlined in Section 2.2), and provides simple closed form formulas for measures of interest such as the expected queue length. In the case of QBD processes, Eq.(9) reduces to the matrix quadratic equation

$$\mathbf{F} + \mathbf{R}\cdot\mathbf{L} + \mathbf{R}^2\cdot\mathbf{B} = \mathbf{0}. \tag{12}$$

QBD processes have been studied extensively and several fast algorithms have been proposed for the solution of Eq.(12), with most notable the logarithmic reduction algorithm, proposed by Latouche and Ramaswami (1999). Ramaswami and Latouche (1986) and Ramaswami and Wang (1996) identify several cases that allow for the explicit computation of $\mathbf{R}$.

$\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(1)}$ are obtained by solving the following system of $m+n$ equations:

$$[\boldsymbol{\pi}^{(0)},\ \boldsymbol{\pi}^{(1)}]\cdot\left[\begin{array}{c|c|c} \widehat{\mathbf{L}}^{\diamond} & \widehat{\mathbf{F}} & \mathbf{1}^T \\ \widehat{\mathbf{B}}^{\diamond} & \mathbf{L}+\mathbf{R}\cdot\mathbf{B} & (\mathbf{I}-\mathbf{R})^{-1}\cdot\mathbf{1}^T \end{array}\right] = [\mathbf{0}\mid 1]. \tag{13}$$

Again, the average queue length is given by the same equation as in the GI/M/1 case.

3. ETAQA Solution for M/G/1-type Processes

In Section 2.1, we described the matrix analytic method for the solution of M/G/1-type processes. Here, we present ETAQA, an aggregated technique that computes only $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ and the aggregated probability vector $\boldsymbol{\pi}^{(*)} = \sum_{i=2}^{\infty}\boldsymbol{\pi}^{(i)}$. This approach is exact and very efficient with respect to both its time and space complexity (see the discussion in Section 3.2).

The block partitioning of the infinitesimal generator as shown in Eq.(1) defines a block partitioning of the stationary probability vector $\boldsymbol{\pi}$ as $\boldsymbol{\pi} = [\boldsymbol{\pi}^{(0)},\ \boldsymbol{\pi}^{(1)},\ \boldsymbol{\pi}^{(2)},\ ...]$ with $\boldsymbol{\pi}^{(0)} \in$

$\mathbb{R}^m$ and $\boldsymbol{\pi}^{(i)} \in \mathbb{R}^n$, for $i \geq 1$. First, we rewrite the matrix equality $\boldsymbol{\pi} \cdot \mathbf{Q}_{M/G/1} = \mathbf{0}$ as:

$$\left\{ \begin{array}{llllll} \boldsymbol{\pi}^{(0)}\cdot\widehat{\mathbf{L}} & +\boldsymbol{\pi}^{(1)}\cdot\ \widehat{\mathbf{B}} & & & & = \mathbf{0} \\ \boldsymbol{\pi}^{(0)}\cdot\widehat{\mathbf{F}}^{(1)} & +\boldsymbol{\pi}^{(1)}\cdot\mathbf{L} & +\boldsymbol{\pi}^{(2)}\cdot\ \mathbf{B} & & & = \mathbf{0} \\ \boldsymbol{\pi}^{(0)}\cdot\widehat{\mathbf{F}}^{(2)} & +\boldsymbol{\pi}^{(1)}\cdot\mathbf{F}^{(1)} & +\boldsymbol{\pi}^{(2)}\cdot\mathbf{L} & +\boldsymbol{\pi}^{(3)}\cdot\ \mathbf{B} & & = \mathbf{0} \\ \boldsymbol{\pi}^{(0)}\cdot\widehat{\mathbf{F}}^{(3)} & +\boldsymbol{\pi}^{(1)}\cdot\mathbf{F}^{(2)} & +\boldsymbol{\pi}^{(2)}\cdot\mathbf{F}^{(1)} & +\boldsymbol{\pi}^{(3)}\cdot\mathbf{L} & +\boldsymbol{\pi}^{(4)}\cdot\mathbf{B} & = \mathbf{0} \\ & & \vdots & & & \end{array} \right. . \tag{14}$$

The first step toward the solution of an M/G/1-type process is the computation of matrix $\mathbf{G}$. We assume that $\mathbf{G}$ is available, i.e., it has been computed using an efficient iterative method, e.g., the cyclic reduction algorithm of Bini et al. (2000), or that it can be explicitly obtained if the process falls in one of the cases identified by Ramaswami and Latouche (1986) and Ramaswami and Wang (1996).

Theorem 1 *Given an ergodic CTMC with infinitesimal generator $\mathbf{Q}_{M/G/1}$ having the structure shown in Eq.(1), with stationary probability vector $\boldsymbol{\pi} = [\boldsymbol{\pi}^{(0)},\ \boldsymbol{\pi}^{(1)},\ \boldsymbol{\pi}^{(2)},\ ...]$ the system of linear equations*

$$\mathbf{x} \cdot \mathbf{X} = [1, \mathbf{0}], \tag{15}$$

where $\mathbf{X} \in \mathbb{R}^{(m+2n)\times(m+2n)}$ is defined as follows

$$\mathbf{X} = \left[\begin{array}{c|c|c|c} \mathbf{1}^T & \widehat{\mathbf{L}} & \widehat{\mathbf{F}}^{(1)} - \sum_{i=3}^{\infty} \widehat{\mathbf{S}}^{(i)} \cdot \mathbf{G} & (\sum_{i=2}^{\infty} \widehat{\mathbf{F}}^{(i)} + \sum_{i=3}^{\infty} \widehat{\mathbf{S}}^{(i)} \cdot \mathbf{G})^{\diamond} \\ \mathbf{1}^T & \widehat{\mathbf{B}} & \mathbf{L} - \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G} & (\sum_{i=1}^{\infty} \mathbf{F}^{(i)} + \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G})^{\diamond} \\ \mathbf{1}^T & \mathbf{0} & \mathbf{B} - \sum_{i=1}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G} & (\sum_{i=1}^{\infty} \mathbf{F}^{(i)} + \mathbf{L} + \sum_{i=1}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G})^{\diamond} \end{array} \right], \tag{16}$$

admits a unique solution $\mathbf{x} = [\boldsymbol{\pi}^{(0)},\ \boldsymbol{\pi}^{(1)},\ \boldsymbol{\pi}^{(*)}]$, where $\boldsymbol{\pi}^{(*)} = \sum_{i=2}^{\infty} \boldsymbol{\pi}^{(i)}$.

Proof. We first show that $[\boldsymbol{\pi}^{(0)},\ \boldsymbol{\pi}^{(1)},\ \boldsymbol{\pi}^{(*)}]$ is a solution of Eq.(15) by verifying that it satisfies four matrix equations corresponding to the four sets of columns we used to define $\mathbf{X}$.

(i) The first equation is the normalization constraint:

$$\boldsymbol{\pi}^{(0)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(1)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(*)} \cdot \mathbf{1}^T = 1. \tag{17}$$

(ii) The second set of m equations is the first line in Eq.(14):

$$\boldsymbol{\pi}^{(0)} \cdot \widehat{\mathbf{L}} + \boldsymbol{\pi}^{(1)} \cdot \widehat{\mathbf{B}} = \mathbf{0}. \tag{18}$$

(iii) The third set of n equations is derived beginning from the second line in Eq.(14):

$$\boldsymbol{\pi}^{(0)} \cdot \widehat{\mathbf{F}}^{(1)} + \boldsymbol{\pi}^{(1)} \cdot \mathbf{L} + \boldsymbol{\pi}^{(2)} \cdot \mathbf{B} = \mathbf{0}.$$

Because our solution does not compute explicitly $\boldsymbol{\pi}^{(2)}$, we rewrite $\boldsymbol{\pi}^{(2)}$, such that it is expressed in terms of $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ and $\boldsymbol{\pi}^{(*)}$ only. By substituting $\boldsymbol{\pi}^{(2)}$ in the above equation we obtain:

$$\boldsymbol{\pi}^{(0)} \cdot \widehat{\mathbf{F}}^{(1)} + \boldsymbol{\pi}^{(1)} \cdot \mathbf{L} + \boldsymbol{\pi}^{(*)} \cdot \mathbf{B} - \sum_{j=3}^{\infty} \boldsymbol{\pi}^{(j)} \cdot \mathbf{B} = \mathbf{0}. \tag{19}$$

To compute the sum $\sum_{j=3}^{\infty} \boldsymbol{\pi}^{(j)}$, we use Ramaswami's recursive formula, i.e., Eq.(3), and obtain:

$$\begin{array}{l} \boldsymbol{\pi}^{(3)} = -(\boldsymbol{\pi}^{(0)} \cdot \widehat{\mathbf{S}}^{(3)} + \boldsymbol{\pi}^{(1)} \cdot \mathbf{S}^{(2)} + \boldsymbol{\pi}^{(2)} \cdot \mathbf{S}^{(1)}) \cdot (\mathbf{S}^{(0)})^{-1} \\ \boldsymbol{\pi}^{(4)} = -(\boldsymbol{\pi}^{(0)} \cdot \widehat{\mathbf{S}}^{(4)} + \boldsymbol{\pi}^{(1)} \cdot \mathbf{S}^{(3)} + \boldsymbol{\pi}^{(2)} \cdot \mathbf{S}^{(2)} + \boldsymbol{\pi}^{(3)} \cdot \mathbf{S}^{(1)}) \cdot (\mathbf{S}^{(0)})^{-1} \\ \boldsymbol{\pi}^{(5)} = -(\boldsymbol{\pi}^{(0)} \cdot \widehat{\mathbf{S}}^{(5)} + \boldsymbol{\pi}^{(1)} \cdot \mathbf{S}^{(4)} + \boldsymbol{\pi}^{(2)} \cdot \mathbf{S}^{(3)} + \boldsymbol{\pi}^{(3)} \cdot \mathbf{S}^{(2)} + \boldsymbol{\pi}^{(4)} \cdot \mathbf{S}^{(1)}) \cdot (\mathbf{S}^{(0)})^{-1} \\ \vdots \end{array}, \tag{20}$$

where the matrices $\widehat{\mathbf{S}}^{(i)}$, for $i \geq 3$, and $\mathbf{S}^{(j)}$, for $j \geq 0$ are determined using the definitions in Eq.(4).

From the definition of matrix $\mathbf{G}$ in Eq.(2), it follows that

$$\mathbf{B} = -(\mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(i)} \mathbf{G}^{i}) \cdot \mathbf{G} = -\mathbf{S}^{(0)} \cdot \mathbf{G}.$$

After summing all equations in Eq.(20) by part and multiplying by $\mathbf{B}$, we obtain:

$$\sum_{j=3}^{\infty} \boldsymbol{\pi}^{(j)} \cdot \mathbf{B} = \left(\boldsymbol{\pi}^{(0)} \cdot \sum_{i=3}^{\infty} \widehat{\mathbf{S}}^{(i)} + \boldsymbol{\pi}^{(1)} \cdot \sum_{i=2}^{\infty} \mathbf{S}^{(i)} + \sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)} \cdot \sum_{i=1}^{\infty} \mathbf{S}^{(i)} \right) \cdot (\mathbf{S}^{(0)})^{-1} \cdot \mathbf{S}^{(0)} \cdot \mathbf{G},$$

which further results in:

$$\sum_{j=3}^{\infty} \boldsymbol{\pi}^{(j)} \cdot \mathbf{B} = \boldsymbol{\pi}^{(0)} \cdot \sum_{i=3}^{\infty} \widehat{\mathbf{S}}^{(i)} \cdot \mathbf{G} + \boldsymbol{\pi}^{(1)} \cdot \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G} + \boldsymbol{\pi}^{(*)} \cdot \sum_{i=1}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G}. \tag{21}$$

Substituting Eq.(21) in Eq.(19), we obtain the third set of equations as a function of $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ and $\boldsymbol{\pi}^{(*)}$ only:

$$\boldsymbol{\pi}^{(0)} \cdot \left(\widehat{\mathbf{F}}^{(1)} - \sum_{i=3}^{\infty} \widehat{\mathbf{S}}^{(i)} \cdot \mathbf{G} \right) + \boldsymbol{\pi}^{(1)} \cdot \left(\mathbf{L} - \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G} \right) + \boldsymbol{\pi}^{(*)} \cdot \left(\mathbf{B} - \sum_{i=1}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G} \right) = \mathbf{0}. \tag{22}$$

(iv) Another set of n equations is obtained by summing all lines in Eq.(14) starting from the third line:

$$\boldsymbol{\pi}^{(0)} \cdot \sum_{i=2}^{\infty} \widehat{\mathbf{F}}^{(i)} + \boldsymbol{\pi}^{(1)} \cdot \sum_{i=1}^{\infty} \mathbf{F}^{(i)} + \sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)} \cdot \left(\mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(i)} \right) + \sum_{j=3}^{\infty} \boldsymbol{\pi}^{(j)} \cdot \mathbf{B} = \mathbf{0}.$$

$\mathbf{V}^{(0)}$	$\mathbf{V}^{(1)}$	$\mathbf{V}^{(2)}$	$\mathbf{V}^{(3)}$	$\cdots$
$\widehat{\mathbf{L}}$	$\widehat{\mathbf{F}}^{(1)}$	$\widehat{\mathbf{F}}^{(2)}$	$\widehat{\mathbf{F}}^{(3)}$	$\cdots$
$\widehat{\mathbf{B}}$	$\mathbf{L}$	$\mathbf{F}^{(1)}$	$\mathbf{F}^{(2)}$	$\cdots$
$\mathbf{0}$	$\mathbf{B}$	$\mathbf{L}$	$\mathbf{F}^{(1)}$	$\cdots$
$\mathbf{0}$	$\mathbf{0}$	$\mathbf{B}$	$\mathbf{L}$	$\cdots$
$\mathbf{0}$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{B}$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

$\mathbf{U}$
$\sum_{i=2}^{\infty} \widehat{\mathbf{F}}^{(i)}$
$\sum_{i=1}^{\infty} \mathbf{F}^{(i)}$
$\mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(i)}$
$\mathbf{B} + \mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(i)}$
$\mathbf{B} + \mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(i)}$
$\vdots$

$\mathbf{W}^{(1)}$	$\mathbf{W}^{(2)}$	$\mathbf{W}^{(3)}$	$\cdots$
$\widehat{\mathbf{S}}^{(3)} \cdot \mathbf{G}$	$\widehat{\mathbf{S}}^{(4)} \cdot \mathbf{G}$	$\widehat{\mathbf{S}}^{(5)} \cdot \mathbf{G}$	
$\mathbf{S}^{(2)} \cdot \mathbf{G}$	$\mathbf{S}^{(3)} \cdot \mathbf{G}$	$\mathbf{S}^{(4)} \cdot \mathbf{G}$	
$\mathbf{S}^{(1)} \cdot \mathbf{G}$	$\mathbf{S}^{(2)} \cdot \mathbf{G}$	$\mathbf{S}^{(3)} \cdot \mathbf{G}$	
$-\mathbf{B}$	$\mathbf{S}^{(1)} \cdot \mathbf{G}$	$\mathbf{S}^{(2)} \cdot \mathbf{G}$	
$\mathbf{0}$	$-\mathbf{B}$	$\mathbf{S}^{(1)} \cdot \mathbf{G}$	
$\vdots$	$\vdots$	$\vdots$	

$\mathbf{Y}$
$\widehat{\mathbf{F}}^{(1)} - \sum_{i=3}^{\infty} \widehat{\mathbf{S}}^{(i)} \cdot \mathbf{G}$
$\mathbf{L} - \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G}^{i}$
$\mathbf{B} - \sum_{i=1}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G}^{i}$
$\mathbf{B} - \sum_{i=1}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G}^{i}$
$\mathbf{B} - \sum_{i=1}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G}^{i}$
$\vdots$

$\mathbf{Z}$
$\sum_{i=2}^{\infty} \widehat{\mathbf{F}}^{(i)} + \sum_{i=3}^{\infty} \widehat{\mathbf{S}}^{(i)} \cdot \mathbf{G}$
$\sum_{i=1}^{\infty} \mathbf{F}^{(i)} + \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G}^{i}$
$\mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(i)} + \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G}^{i}$
$\mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(i)} + \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G}^{i}$
$\mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(i)} + \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G}^{i}$
$\vdots$

Figure 2: The Blocks of Column Vectors Used to Prove Linear Independence

Since $\sum_{j=3}^{\infty} \boldsymbol{\pi}^{(j)} \cdot \mathbf{B}$ can be expressed as a function of $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and $\boldsymbol{\pi}^{(*)}$ only (Eq.(21)), the above equation can be rewritten as:

$$\boldsymbol{\pi}^{(0)} \cdot \left(\sum_{i=2}^{\infty} \widehat{\mathbf{F}}^{(i)} + \sum_{i=3}^{\infty} \widehat{\mathbf{S}}^{(i)} \cdot \mathbf{G} \right) + \boldsymbol{\pi}^{(1)} \cdot \left(\sum_{i=1}^{\infty} \mathbf{F}^{(i)} + \sum_{i=2}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G} \right) + \boldsymbol{\pi}^{(*)} \cdot \left(\sum_{i=1}^{\infty} \mathbf{F}^{(i)} + \mathbf{L} + \sum_{i=1}^{\infty} \mathbf{S}^{(i)} \cdot \mathbf{G} \right) = \mathbf{0}. \tag{23}$$

In steps (i) through (iv), we showed that the vector $[\boldsymbol{\pi}^{(0)},\ \boldsymbol{\pi}^{(1)},\ \boldsymbol{\pi}^{(*)}]$ satisfies Eqs. (17), (18), (22), and (23), hence it is a solution of Eq.(15). Now we have to show that this solution is unique. For this, it is enough to prove that the rank of $\mathbf{X}$ is $m + 2n$ by showing that its $m + 2n$ rows are linearly independent.

Since the process with the infinitesimal generator $\mathbf{Q}_{M/G/1}$ is ergodic, we know that the vector $\mathbf{1}^{T}$ and the set of vectors corresponding to all the columns of $\mathbf{Q}_{M/G/1}$ except one, any of them, are linearly independent. We also note that by multiplying a block column of the infinitesimal generator $\mathbf{Q}_{M/G/1}$ with a matrix, we get a block column which is a linear combination of the columns of the selected block column. In our proof, we use multiplication of the block columns with powers of the matrix $\mathbf{G}$.

We begin from the columns of the infinitesimal generator. In Figure 2, we show the blocks of column vectors that we use in our proof. The blocks labeled $\mathbf{V}^{(i)}$ for $i \geq 0$ are the original block columns of $\mathbf{Q}_{M/G/1}$. The block $\mathbf{U}$ is obtained by summing all $\mathbf{V}^{(i)}$ for $i \geq 2$:

$$\mathbf{U} = \sum_{i=2}^{\infty} \mathbf{V}^{(i)}.$$

Blocks $\mathbf{W}^{(i)}$ for $i \geq 1$ are obtained by multiplying the block columns $\mathbf{V}^{(j)}$ for $j \geq i+2$ with the $(j-i+1)^{th}$ power of matrix $\mathbf{G}$ and summing them all together

$$\mathbf{W}^{(i)} = \sum_{j=i}^{\infty} \mathbf{V}^{(j+2)} \cdot \mathbf{G}^{j-i+1}, \quad i \geq 1,$$

which are used to define

$$\mathbf{Y} = \mathbf{V}^{(1)} - \sum_{i=1}^{\infty} \mathbf{W}^{(i)}, \qquad \text{and} \qquad \mathbf{Z} = \mathbf{U} + \sum_{i=1}^{\infty} \mathbf{W}^{(i)}.$$

In the matrix $\mathbf{X}$ defined in Eq.(16), we make use of the three upper blocks of $\mathbf{V}^{(0)}$, $\mathbf{Y}$, and $\mathbf{Z}$. We argue that the rank of the matrix $[\mathbf{V}^{(0)}|\mathbf{Y}|\mathbf{Z}]$ is $m + 2n - 1$ because we obtained $\mathbf{Y}$, and $\mathbf{Z}$ respectively as linear combination of blocks $\mathbf{V}^{(1)}$ and $\mathbf{V}^{(2)}$ with the blocks $\mathbf{W}^{(i)}$ for $i \geq 1$, and none of the columns used to generate $\mathbf{W}^{(i)}$ for $i \geq 1$ is from either $\mathbf{V}^{(1)}$ or $\mathbf{V}^{(2)}$. Recall that $\mathbf{Q}_{M/G/1}$ is an infinitesimal generator, therefore the defect is one and the rank of $[\mathbf{V}^{(0)}|\mathbf{Y}|\mathbf{Z}]$ is exactly $m + 2n - 1$. Substituting one (any) of these columns with a column of $1s$, we obtain the rank of $m + 2n$. □

3.1. Computing Measures of Interest for M/G/1-type Processes

We now consider the problem of obtaining stationary measures of interest once $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and $\boldsymbol{\pi}^{(*)}$ have been computed. Traditionally, such metrics can be calculated using moment generating functions, as in Grassman and Stanford (2000).

Here, we consider measures that can be expressed as the expected reward rate

$$r = \sum_{j=0}^{\infty} \sum_{i \in \mathcal{S}^{(j)}} \boldsymbol{\rho}_i^{(j)} \boldsymbol{\pi}_i^{(j)},$$

where $\boldsymbol{\rho}_i^{(j)}$ is the *reward rate* of state $s_i^{(j)}$. For example, to compute the expected queue length in steady state, where $\mathcal{S}^{(j)}$ represents the system states with j customers in the queue, we let $\boldsymbol{\rho}_i^{(j)} = j$. To compute the second moment of the queue length, we let $\boldsymbol{\rho}_i^{(j)} = j^2$.

Since our solution approach computes $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and $\sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)}$, we rewrite r as

$$r = \boldsymbol{\pi}^{(0)} \boldsymbol{\rho}^{(0)T} + \boldsymbol{\pi}^{(1)} \boldsymbol{\rho}^{(1)T} + \sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)} \boldsymbol{\rho}^{(j)T},$$

where $\boldsymbol{\rho}^{(0)} = [\boldsymbol{\rho}_1^{(0)}, \ldots, \boldsymbol{\rho}_m^{(0)}]$ and $\boldsymbol{\rho}^{(j)} = [\boldsymbol{\rho}_1^{(j)}, \ldots, \boldsymbol{\rho}_n^{(j)}]$, for $j \geq 1$. Then, we must show how to compute the above summation without explicitly using the values of $\boldsymbol{\pi}^{(j)}$ for $j \geq 2$. We can do so if the reward rate of state $s_i^{(j)}$, for $j \geq 2$ and $i = 1, \ldots, n$, is a polynomial of degree k in j with arbitrary coefficients $\mathbf{a}_i^{[0]}, \mathbf{a}_i^{[1]}, \ldots, \mathbf{a}_i^{[k]}$:

$$\forall j \geq 2,\ \forall i \in \{1, 2, \ldots, n\}, \qquad \boldsymbol{\rho}_i^{(j)} = \mathbf{a}_i^{[0]} + \mathbf{a}_i^{[1]} j + \cdots + \mathbf{a}_i^{[k]} j^k. \tag{24}$$

The definition of $\boldsymbol{\rho}_i^{(j)}$ illustrates that the set of measures of interest that we can compute includes any moment of the probability vector $\boldsymbol{\pi}$. The only metrics of interest that we cannot compute using our aggregate approach are those whose reward rates $\boldsymbol{\rho}_i^{(j)}$ for states $s_i^{(j)}$ have different coefficients in their polynomial representation, for different inter-level index $j \geq 2$. The set of measures of interest that cannot be computed by the following methodology does not arise often in practice, since we expect that within each inter-level of the repeating portion of the process the states have similar probabilistic interpretation.

We compute $\sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)} \boldsymbol{\rho}^{(j)T}$ as follows

$$\begin{aligned}
\sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)} \boldsymbol{\rho}^{(j)T} &= \sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)} \left(\mathbf{a}^{[0]} + \mathbf{a}^{[1]} j + \cdots + \mathbf{a}^{[k]} j^k \right)^T \\
&= \sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)} \mathbf{a}^{[0]T} + \sum_{j=2}^{\infty} j \boldsymbol{\pi}^{(j)} \mathbf{a}^{[1]T} + \cdots + \sum_{j=2}^{\infty} j^k \boldsymbol{\pi}^{(j)} \mathbf{a}^{[k]T} \\
&= \mathbf{r}^{[0]} \mathbf{a}^{[0]T} + \mathbf{r}^{[1]} \mathbf{a}^{[1]T} + \cdots + \mathbf{r}^{[k]} \mathbf{a}^{[k]T},
\end{aligned}$$

and the problem is reduced to the computation of $\mathbf{r}^{[l]} = \sum_{j=2}^{\infty} j^l \boldsymbol{\pi}^{(j)}$, for $l = 0, \ldots, k$.

We show how $\mathbf{r}^{[k]}$, $k > 0$, can be computed recursively, starting from $\mathbf{r}^{[0]}$, which is simply $\boldsymbol{\pi}^{(*)}$. Multiplying the equations in (14) from the second line on by the appropriate factor j^k results in

$$\left\{ \begin{array}{lllllllllll}
2^k \boldsymbol{\pi}^{(0)} & \widehat{\mathbf{F}}^{(1)} & + & 2^k \boldsymbol{\pi}^{(1)} & \mathbf{L} & + & 2^k \boldsymbol{\pi}^{(2)} & \mathbf{B} & & & = \mathbf{0} \\
3^k \boldsymbol{\pi}^{(0)} & \widehat{\mathbf{F}}^{(2)} & + & 3^k \boldsymbol{\pi}^{(1)} & \mathbf{F}^{(1)} & + & 3^k \boldsymbol{\pi}^{(2)} & \mathbf{L} & + & 3^k \boldsymbol{\pi}^{(3)} \ \mathbf{B} & = \mathbf{0} \\
 & & & & \vdots & & & & & &
\end{array} \right. .$$

Summing these equations by parts, we obtain

$$\underbrace{\boldsymbol{\pi}^{(0)} \sum_{j=1}^{\infty} (j+1)^k \widehat{\mathbf{F}}^{(j)}}_{\stackrel{\text{def}}{=}\ \widehat{\mathbf{f}}} + \underbrace{\boldsymbol{\pi}^{(1)} \left(2^k \mathbf{L} + \sum_{j=1}^{\infty} (j+2)^k \mathbf{F}^{(j)} \right)}_{\stackrel{\text{def}}{=}\ \mathbf{f}} +$$

$$\sum_{h=2}^{\infty} \boldsymbol{\pi}^{(h)} \left(\sum_{j=1}^{\infty} (j+h+1)^k \mathbf{F}^{(j)} + (h+1)^k \mathbf{L} \right) + \underbrace{\sum_{h=2}^{\infty} h^k \boldsymbol{\pi}^{(h)}}_{=\ \mathbf{r}^{[k]}} \mathbf{B} = \mathbf{0},$$

which can then be rewritten as

$$\sum_{h=2}^{\infty} \boldsymbol{\pi}^{(h)} \left[\left(\sum_{j=1}^{\infty} \sum_{l=0}^{k} \begin{pmatrix} k \\ l \end{pmatrix} (j+1)^l h^{k-l} \mathbf{F}^{(j)} \right) + \left(\sum_{l=0}^{k} \begin{pmatrix} k \\ l \end{pmatrix} 1^l h^{k-l} \mathbf{L} \right) \right] + \mathbf{r}^{[k]} \mathbf{B} = -\widehat{\mathbf{f}} - \mathbf{f}.$$

Exchanging the order of summations, we obtain

$$\sum_{l=0}^{k} \begin{pmatrix} k \\ l \end{pmatrix} \underbrace{\sum_{h=2}^{\infty} \boldsymbol{\pi}^{(h)} h^{k-l}}_{=\ \mathbf{r}^{[k-l]}} \left(\mathbf{L} + \sum_{j=1}^{\infty} (j+1)^l \mathbf{F}^{(j)} \right) + \mathbf{r}^{[k]} \mathbf{B} = -\widehat{\mathbf{f}} - \mathbf{f}.$$

Finally, isolating the case $l = 0$ in the outermost summation we obtain

$$\mathbf{r}^{[k]} \left(\mathbf{B} + \mathbf{L} + \sum_{j=1}^{\infty} \mathbf{F}^{(j)} \right) = -\widehat{\mathbf{f}} - \mathbf{f} - \sum_{l=1}^{k} \begin{pmatrix} k \\ l \end{pmatrix} \mathbf{r}^{[k-l]} \left(\mathbf{L} + \sum_{j=1}^{\infty} (j+1)^l \mathbf{F}^{(j)} \right),$$

which is a linear system of the form $\mathbf{r}^{[k]}(\mathbf{B} + \mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(j)}) = \mathbf{b}^{[k]}$, where the right-hand side $\mathbf{b}^{[k]}$ is an expression that can be effectively computed from $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and the vectors $\mathbf{r}^{[0]}$ through $\mathbf{r}^{[k-1]}$. However, the rank of $(\mathbf{B} + \mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(j)})$ is $n - 1$. This is true because $(\mathbf{B} + \mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(j)})$ is an infinitesimal generator with rank $n - 1$, so the above system is under-determined. We drop any of the columns of $\mathbf{B} + \mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(j)}$, resulting in

$$\mathbf{r}^{[k]} (\mathbf{B} + \mathbf{L} + \sum_{i=1}^{\infty} \mathbf{F}^{(j)})^{\diamond} = (\mathbf{b}^{[k]})^{\diamond}, \tag{25}$$

and obtain one additional equation for $\mathbf{r}^{[k]}$ by using the flow balance equations between $\cup_{l=0}^{j} \mathcal{S}^{(l)}$ and $\cup_{l=j+1}^{\infty} \mathcal{S}^{(l)}$ for each $j \geq 1$ and multiplying them by the appropriate factor j^k,

$$\begin{cases} 2^k \boldsymbol{\pi}^{(0)} \displaystyle\sum_{l=2}^{\infty} \widehat{\mathbf{F}}^{(l)} \mathbf{1}^T + 2^k \boldsymbol{\pi}^{(1)} \sum_{l=1}^{\infty} \mathbf{F}^{(l)} \mathbf{1}^T = 2^k \boldsymbol{\pi}^{(2)} \mathbf{B} \mathbf{1}^T \\ 3^k \boldsymbol{\pi}^{(0)} \displaystyle\sum_{l=3}^{\infty} \widehat{\mathbf{F}}^{(l)} \mathbf{1}^T + 3^k \boldsymbol{\pi}^{(1)} \sum_{l=2}^{\infty} \mathbf{F}^{(l)} \mathbf{1}^T + 3^k \boldsymbol{\pi}^{(2)} \sum_{l=1}^{\infty} \mathbf{F}^{(l)} \mathbf{1}^T = 3^k \boldsymbol{\pi}^{(3)} \mathbf{B} \mathbf{1}^T \\ \vdots \end{cases} \cdot \tag{26}$$

We introduce the following notation

$$\widehat{\mathbf{F}}_{[k,j]} = \sum_{l=j}^{\infty} l^k \cdot \widehat{\mathbf{F}}^{(l)}, \qquad \mathbf{F}_{[k,j]} = \sum_{l=j}^{\infty} l^k \cdot \mathbf{F}^{(l)}, \qquad j \geq 1. \tag{27}$$

We then sum all lines in Eq.(26) and obtain:

$$\underbrace{\boldsymbol{\pi}^{(0)} \sum_{j=2}^{\infty} j^k \widehat{\mathbf{F}}_{[0,j]} \mathbf{1}^T}_{\stackrel{\text{def}}{=}\ \hat{f}_c} + \underbrace{\boldsymbol{\pi}^{(1)} \sum_{j=1}^{\infty} (j+1)^k \mathbf{F}_{[0,j]} \mathbf{1}^T}_{\stackrel{\text{def}}{=}\ f_c} + \sum_{h=2}^{\infty} \boldsymbol{\pi}^{(h)} \sum_{j=1}^{\infty} (j+h)^k \mathbf{F}_{[0,j]} \mathbf{1}^T = \underbrace{\sum_{j=2}^{\infty} j^k \boldsymbol{\pi}^{(j)}}_{=\ \mathbf{r}^{[k]}} \mathbf{B} \mathbf{1}^T,$$

which, with steps analogous to those just performed to obtain Eq.(25), can be written as

$$\mathbf{r}^{[k]}(\mathbf{F}_{[1,1]} - \mathbf{B})\mathbf{1}^T = c^{[k]}, \tag{28}$$

where $c^{[k]}$ is defined as:

$$c^{[k]} = -(\hat{f}_c + f_c + \sum_{l=1}^{k} \begin{pmatrix} k \\ l \end{pmatrix} \mathbf{r}^{[k-l]} \sum_{j=1}^{\infty} j^l \mathbf{F}_{[0,j]} \cdot \mathbf{1}^T). \tag{29}$$

Note that the $n \times n$ matrix

$$[(\mathbf{B} + \mathbf{L} + \mathbf{F}_{[0,1]})^\diamond | (\mathbf{F}_{[1,1]} - \mathbf{B})\mathbf{1}^T] \tag{30}$$

has full rank. This is true because $(\mathbf{B}+\mathbf{L}+\mathbf{F}_{[0,1]})$ is an infinitesimal generator with rank $n-1$, thus has a unique stationary probability vector $\boldsymbol{\gamma}$ satisfying $\boldsymbol{\gamma}(\mathbf{B}+\mathbf{L}+\mathbf{F}_{[0,1]}) = \mathbf{0}$. However, this same vector must satisfy $\boldsymbol{\gamma}\mathbf{B}\mathbf{1}^T > \boldsymbol{\gamma}\mathbf{F}_{[1,1]}\mathbf{1}^T$ to ensure that the process has a positive drift toward $\mathcal{S}^{(0)}$, thus is ergodic, hence $\boldsymbol{\gamma}(\mathbf{F}_{[1,1]} - \mathbf{B})\mathbf{1}^T < 0$, which shows that $(\mathbf{F}_{[1,1]} - \mathbf{B})\mathbf{1}^T$ cannot be possibly obtained as linear combination of columns in $(\mathbf{B} + \mathbf{L} + \mathbf{F}_{[0,1]})$, therefore the $n \times n$ matrix defined in Eq.(30) has full rank.

Hence, we can compute $\mathbf{r}^{[k]}$ using Eqs. (25) and (28), i.e., solving a linear system in n unknowns (of course, we must do so first for $l = 1, \ldots, k-1$).

As an example, we consider $\mathbf{r}^{[1]}$, which is used to compute measures such as the first moment of the queue length. In this case,

$$\mathbf{b}^{[1]} = -\left(\boldsymbol{\pi}^{(0)} \sum_{j=1}^{\infty}(j+1) \cdot \widehat{\mathbf{F}}^{(j)} + \boldsymbol{\pi}^{(1)}(2\mathbf{L} + \sum_{j=1}^{\infty}(j+2) \cdot \mathbf{F}^{(j)}) + \boldsymbol{\pi}^{(*)}(\mathbf{L} + \sum_{j=1}^{\infty}(j+1) \cdot \mathbf{F}^{(j)}\right),$$

and

$$c^{[1]} = -\left(\boldsymbol{\pi}^{(0)} \sum_{j=2}^{\infty} j\widehat{\mathbf{F}}_{[0,j]} + \boldsymbol{\pi}^{(1)} \sum_{j=1}^{\infty}(j+1)\mathbf{F}_{[0,j]} + \boldsymbol{\pi}^{(*)} \sum_{j=1}^{\infty} j\mathbf{F}_{[0,j]}) \cdot \mathbf{1}^T\right).$$

In the general case that was considered here some measures might be infinite. For example, if the sequences are summable but decrease only like $1/j^h$ for some $h > 1$, then the moments of order $h-1$ or higher for the queue length do not exist (are infinite). From the practical point of view, we always store a finite set of matrices from the sequences $\{\widehat{\mathbf{F}}^{(j)} : j \geq 1\}$ and $\{\mathbf{F}^{(j)} : j \geq 1\}$, so the sums of type $\widehat{\mathbf{F}}_{[k,j]}$ and $\mathbf{F}_{[k,j]}$ for $j \geq 1, k \geq 0$ are always finite.

We conclude by observing that, when the sequences $\{\widehat{\mathbf{F}}^{(j)} : j \geq 1\}$ and $\{\mathbf{F}^{(j)} : j \geq 1\}$ do have a nicer relation, like a geometric one, the treatment in this section can be modified appropriately to simplify the different sums introduced here, and give closed form formulas.

3.2. Time and Storage Complexity

In this section, we present a detailed comparison of ETAQA for M/G/1-type processes with the Matrix-analytic method using the Fast-FFT implementation of Ramaswami's recursive formula as outlined in Subsection 2.1. The complexity analysis is within the accuracy of O-notation. In our analysis, $O^L(x)$ denotes the time complexity of solving a linear system described by x nonzero entries and $\eta\{\mathbf{A}\}$ denotes the number of nonzero entries in matrix $\mathbf{A}$. In the general case, $\eta\{\widehat{\mathbf{F}}\}$ and $\eta\{\mathbf{F}\}$ should be taken to mean $\eta\{\cup_{j=1}^{p}\widehat{\mathbf{F}}^{(j)}\}$ and $\eta\{\cup_{j=1}^{p}\mathbf{F}^{(j)}\}$, respectively.

Since practically, we cannot store an infinite number of matrices, we store up to p matrices of type $\widehat{\mathbf{F}}^{(j)}$ and $\mathbf{F}^{(j)}$, $j \geq 1$. Furthermore, for the matrix analytic method to reach the necessary accuracy, it is necessary to compute up to s block vectors $\boldsymbol{\pi}^{(i)}$ of the stationary probability vector $\boldsymbol{\pi}$.

We outline the required steps for each method and analyze the computation and storage complexity of each step up to the computation of the expected queue length of the process. In our analysis, we do not include the cost to compute the matrix $\mathbf{G}$ since *both* methodologies require the computation of $\mathbf{G}$ as a first step. Note that $\mathbf{G}$ should be computed with an efficient method like the cyclic-reduction algorithm of Bini et al. (2000). Furthermore, we do not consider the cost of computing $\widehat{\mathbf{S}}^{(i)}$ and $\mathbf{S}^{(i)}$ for $i \geq 0$ since they are required in both methodologies.

Analysis of ETAQA for M/G/1 processes:

- Computation of the aggregate stationary probability vector $\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(*)}$
  - $O(p \cdot (m \cdot \eta\{\widehat{\mathbf{F}}, \mathbf{G}\} + n \cdot \eta\{\mathbf{F}, \mathbf{G}\}))$ to compute sums of the form $\sum_{i=k}^{\infty} \widehat{\mathbf{S}}^{(i)} \cdot \mathbf{G}$ and $\sum_{i=k}^{\infty} \widehat{\mathbf{S}}^{(i)} \cdot \mathbf{G}$ for $i \geq 1$, and $k = 1, 2, 3$, whose sparsity depends directly on the sparsity of $\mathbf{G}$, $\widehat{\mathbf{F}}^{(i)}$ and $\mathbf{F}^{(i)}$ for $i \geq 1$.
  - $O(p \cdot (\eta\{\widehat{\mathbf{F}}\} + \eta\{\mathbf{F}\}))$ to compute sums of the form $\sum_{j=1}^{\infty} \mathbf{F}^{(j)}$, and $\sum_{j=2}^{\infty} \widehat{\mathbf{F}}^{(j)}$.
  - $O^L(\eta\{\widehat{\mathbf{B}}, \widehat{\mathbf{L}}, \mathbf{L}, \widehat{\mathbf{F}}, \mathbf{F}, \mathbf{G}\})$ for the solution of the system of $m + 2n$ linear equations.
- Storage requirements for computation of $\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(*)}$
  - $O(m \cdot n + n^2)$ to store the sums $\sum_{i=1}^{\infty} \widehat{\mathbf{S}}^{(i)}$ and $\sum_{i=1}^{\infty} \mathbf{S}^{(i)}$.
  - $m + 2n$ to store the probability vectors $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ and $\boldsymbol{\pi}^{(*)}$.

- Computation of the expected queue length
  - $O(p \cdot (\eta\{\widehat{\mathbf{F}}\} + \eta\{\mathbf{F}\}))$ to compute sums of the form $\sum_{j=1}^{\infty} j^k \cdot \mathbf{F}^{(j)}$, and $\sum_{j=2}^{\infty} j^k \cdot \widehat{\mathbf{F}}^{(j)}$ where k is a constant.
  - $O^L(\eta\{\mathbf{F}, \mathbf{L}, \mathbf{B}\})$ for the solution of the sparse system of n linear equations.
- No additional storage requirements for computation of the expected queue length.

Analysis of M/G/1 matrix-analytic methodology:

- Computation of the stationary probability vector $\boldsymbol{\pi}$
  - $O(p \cdot (m \cdot \eta\{\widehat{\mathbf{F}}, \mathbf{G}\} + n \cdot \eta\{\mathbf{F}, \mathbf{G}\}))$ to compute the sums of the form $\widehat{\mathbf{S}}^{(i)}$ for $i \geq 1$, and $\mathbf{S}^{(i)}$ for $i \geq 0$.
  - $O(n^3 + m \cdot \eta\{\widehat{\mathbf{F}}, \mathbf{G}\} + n \cdot \eta\{\widehat{\mathbf{B}}\})$ for the computation of the inverses of $\mathbf{S}^{(0)}$ and $\sum_{j=0}^{\infty} \mathbf{S}^{(j)}$ and additional multiplications of full matrices.
  - $O^L(m^2)$ for the solution of the system of m linear equations.
  - $O(pn^3 + sn^2 + p \log p)$ (Meini (1997b)), since the fast FFT-based version of Ramaswami's recursive formula is used to compute the s vectors of the stationary probability vector.
- Storage requirements for computation of $\boldsymbol{\pi}$
  - $O(p \cdot (m \cdot n + n^2))$ to store all sums of form $\widehat{\mathbf{S}}^{(i)}$ for $i \geq 1$, and $\mathbf{S}^{(i)}$ for $i \geq 0$.
  - m to store $\boldsymbol{\pi}^{(0)}$.
  - $s \cdot n$ to store vectors $\boldsymbol{\pi}^{(i)}$ for $i \geq 1$.
- Computation of the expected queue length
  - $O(s \cdot n)$ to compute the queue length.
- No additional storage requirements for the computation of the expected queue length.

Tables 1 and 2 summarize the discussion in this section.

Concluding our analysis, we point out that the ETAQA solution is a more efficient approach, both computation- and storage-wise. In comparison to the Matrix-analytic solution, it entails only a few steps and is thus much easier to implement. Since we do not need to

Table 1: Computational Complexities of ETAQA-M/G/1 and Matrix-analytic

Computation of $\boldsymbol{\pi}$ (Matrix-analytic) and $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, $\boldsymbol{\pi}^{(*)}$ (ETAQA-M/G/1)	
ETAQA-M/G/1	$O^L(\eta\{\widehat{\mathbf{B}}, \widehat{\mathbf{L}}, \mathbf{L}, \widehat{\mathbf{F}}, \mathbf{F}, \mathbf{G}\}) + O(p \cdot (m \cdot \eta\{\widehat{\mathbf{F}}, \mathbf{G}\} + n \cdot \eta\{\mathbf{F}, \mathbf{G}\}))$
Matrix-analytic	$O^L(m^2) + O(p \cdot (m \cdot \eta\{\widehat{\mathbf{F}}, \mathbf{G}\} + n \cdot \eta\{\mathbf{F}, \mathbf{G}, \widehat{\mathbf{B}}\}) + pn^3 + sn^2 + p \log p)$
First moment measures	
ETAQA-M/G/1	$O^L(\eta\{\mathbf{B}, \mathbf{L}, \mathbf{F}\}) + O(p \cdot \eta(\widehat{\mathbf{F}}) + p \cdot \eta(\mathbf{F}))$
Matrix-analytic	$O(s \cdot n)$

Table 2: Storage Complexities of ETAQA-MG1 and Matrix-analytic

	Additional storage	**Storage of the probabilities**
Computation of $\boldsymbol{\pi}^{(0)}$ (Matrix-analytic) or $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, $\boldsymbol{\pi}^{(*)}$ (ETAQA-M/G/1)		
ETAQA-M/G/1	$O(m \cdot n + n^2)$	$m + 2n$
Matrix-analytic	$O(p \cdot (m \cdot n + n^2))$	$m + s \cdot n$
First moment measures		
ETAQA-M/G/1	none	none
Matrix-analytic	none	none

generate the whole stationary probability vector, in our complexity analysis the term s does not appear for ETAQA-M/G/1 which, in comparison with the value of p or n, is several times higher.

Furthermore, since the ETAQA solution does not introduce any matrix inversion or matrix multiplication, the sparsity of the original process is preserved resulting in significant savings with respect to both computation and storage. We emphasize the fact that the sparsity of $\mathbf{G}$ is key for preserving the sparsity of the original process, in both methods. There are special cases where $\mathbf{G}$ is very sparse (e.g., $\mathbf{G}$ is a single column matrix if $\mathbf{B}$ is a single column matrix). In these cases, the sums of the form $\widehat{\mathbf{S}}^{(i)}$ for $i \geq 1$, and $\mathbf{S}^{(i)}$ for $i \geq 0$ almost preserve the sparsity of the original process and reduce the computation and storage cost.

4. ETAQA Solution for GI/M/1-type Processes

We apply the same aggregation technique, that we first introduced in Section 3, to obtain the exact aggregate solution of GI/M/1-type processes. Using the same block partitioning of the stationary probability vector $\boldsymbol{\pi}$ allows us to rewrite the matrix equality $\boldsymbol{\pi} \cdot \mathbf{Q}_{GI/M/1} = \mathbf{0}$

as:

$$\left\{\begin{array}{lllllll} & & \boldsymbol{\pi}^{(0)}\ \widehat{\mathbf{L}} & + & \sum_{i=1}^{\infty}\boldsymbol{\pi}^{(i)}\ \widehat{\mathbf{B}}^{(i)} & = & \mathbf{0}\\ \boldsymbol{\pi}^{(0)}\ \widehat{\mathbf{F}} & + & \boldsymbol{\pi}^{(1)}\ \mathbf{L} & + & \sum_{i=2}^{\infty}\boldsymbol{\pi}^{(i)}\ \mathbf{B}^{(i-1)} & = & \mathbf{0}\\ \boldsymbol{\pi}^{(1)}\ \mathbf{F} & + & \boldsymbol{\pi}^{(2)}\ \mathbf{L} & + & \sum_{i=3}^{\infty}\boldsymbol{\pi}^{(i)}\ \mathbf{B}^{(i-2)} & = & \mathbf{0}\\ \boldsymbol{\pi}^{(2)}\ \mathbf{F} & + & \boldsymbol{\pi}^{(3)}\ \mathbf{L} & + & \sum_{i=4}^{\infty}\boldsymbol{\pi}^{(i)}\ \mathbf{B}^{(i-3)} & = & \mathbf{0}\\ & & & \vdots & & & \end{array}\right. . \tag{31}$$

Assuming that matrix $\mathbf{R}$ is available, we apply the same steps as for the case of M/G/1-type processes and formulate the following theorem:

Theorem 2 *Given an ergodic CTMC with infinitesimal generator $\mathbf{Q}_{GI/M/1}$ having the structure shown in Eq.(7), with stationary probability vector $\boldsymbol{\pi} = [\boldsymbol{\pi}^{(0)},\ \boldsymbol{\pi}^{(1)},\ \boldsymbol{\pi}^{(2)},\ ...]$, the system of linear equations*

$$\mathbf{x}\cdot\mathbf{X} = [1, \mathbf{0}] \tag{32}$$

where $\mathbf{X} \in \mathbb{R}^{(m+2n)\times(m+2n)}$ is defined as follows

$$\mathbf{X} = \left[\begin{array}{c|c|c|c} \mathbf{1}^T & \widehat{\mathbf{L}} & \widehat{\mathbf{F}} & \mathbf{0}^{\diamond}\\ \mathbf{1}^T & \widehat{\mathbf{B}}^{(1)} & \mathbf{L} & \mathbf{F}^{\diamond}\\ \mathbf{1}^T & \sum_{i=2}^{\infty}\mathbf{R}^{i-2}\cdot(\mathbf{I}-\mathbf{R})\cdot\widehat{\mathbf{B}}^{(i)} & \sum_{i=1}^{\infty}\mathbf{R}^{i-1}\cdot(\mathbf{I}-\mathbf{R})\cdot\mathbf{B}^{(i)} & (\mathbf{F}+\mathbf{L}+\sum_{i=1}^{\infty}\mathbf{R}^{i}\cdot\mathbf{B}^{(i)})^{\diamond} \end{array}\right], \tag{33}$$

admits a unique solution $\mathbf{x} = [\boldsymbol{\pi}^{(0)},\ \boldsymbol{\pi}^{(1)},\ \boldsymbol{\pi}^{(*)}]$, where $\boldsymbol{\pi}^{(*)} = \sum_{i=2}^{\infty}\boldsymbol{\pi}^{(i)}$.

Proof. The steps to derive Eq.(33) are outlined as follows.

(i) The first equation is the normalization constraint:

$$\boldsymbol{\pi}^{(0)}\cdot\mathbf{1}^T + \boldsymbol{\pi}^{(1)}\cdot\mathbf{1}^T + \boldsymbol{\pi}^{(*)}\cdot\mathbf{1}^T = 1. \tag{34}$$

(ii) From the first line in Eq.(31) we have:

$$\boldsymbol{\pi}^{(0)}\cdot\widehat{\mathbf{L}} + \boldsymbol{\pi}^{(1)}\cdot\widehat{\mathbf{B}}^{(1)} + \sum_{i=2}^{\infty}\boldsymbol{\pi}^{(i)}\cdot\widehat{\mathbf{B}}^{(i)} = \mathbf{0}.$$

The sum $\sum_{i=2}^{\infty}\boldsymbol{\pi}^{(i)}\cdot\widehat{\mathbf{B}}^{(i)}$ can be expressed as:

$$\sum_{i=2}^{\infty}\boldsymbol{\pi}^{(i)}\cdot\widehat{\mathbf{B}}^{(i)} = \sum_{i=2}^{\infty}(\sum_{j=i}^{\infty}\boldsymbol{\pi}^{(j)} - \sum_{j=i+1}^{\infty}\boldsymbol{\pi}^{(j)})\cdot\widehat{\mathbf{B}}^{(i)},$$

and after simple derivations that exploit the geometric relation of the stationary probability vectors $\boldsymbol{\pi}^{(j)}$, for $j \geq 2$, we obtain m equations:

$$\boldsymbol{\pi}^{(0)}\cdot\widehat{\mathbf{L}} + \boldsymbol{\pi}^{(1)}\cdot\widehat{\mathbf{B}}^{(1)} + \boldsymbol{\pi}^{(*)}\sum_{i=2}^{\infty}\mathbf{R}^{i-2}\cdot(\mathbf{I}-\mathbf{R})\cdot\widehat{\mathbf{B}}^{(i)} = \mathbf{0}.$$

(iii) From the second line of Eq.(31) and using similar derivations as in step (ii), we get the third set of n equations:

$$\boldsymbol{\pi}^{(0)} \cdot \widehat{\mathbf{F}} + \boldsymbol{\pi}^{(1)} \cdot \mathbf{L} + \boldsymbol{\pi}^{(*)} \sum_{i=1}^{\infty} \mathbf{R}^{i-1} \cdot (\mathbf{I} - \mathbf{R}) \cdot \mathbf{B}^{(i)} = \mathbf{0}.$$

(iv) Another set of n equations is obtained by summing all the remaining lines in Eq.(31):

$$\boldsymbol{\pi}^{(1)} \cdot \mathbf{F} + \boldsymbol{\pi}^{(*)} \cdot (\mathbf{L} + \mathbf{F}) + \sum_{i=3}^{\infty} \sum_{j=i}^{\infty} \boldsymbol{\pi}^{(j)} \mathbf{B}^{(i-2)} = \mathbf{0},$$

and by expressing the sum $\sum_{i=3}^{\infty} \sum_{j=i}^{\infty} \boldsymbol{\pi}^{(j)} \mathbf{B}^{(i-2)}$ as a function of $\boldsymbol{\pi}^{(*)}$, we obtain an additional set of n equations:

$$\boldsymbol{\pi}^{(1)} \cdot \mathbf{F} + \boldsymbol{\pi}^{(*)} \left(\mathbf{L} + \mathbf{F} + \sum_{i=1}^{\infty} \mathbf{R}^{i} \cdot \mathbf{B}^{(i)} \right) = \mathbf{0}.$$

The matrix $\mathbf{X}$ has full rank. This follows from the fact that the infinitesimal generator $\mathbf{Q}_{GI/M/1}$ has a defect of one. We obtained the second and the third block columns in $\mathbf{X}$ by keeping their respective first two upper blocks in the first block column of $\mathbf{Q}_{GI/M/1}$ and substituting the remaining lower blocks with one block that results as a linear combination of the remaining lower blocks within the same block column of $\mathbf{Q}_{GI/M/1}$. We obtained the fourth block column in $\mathbf{X}$ by keeping the first two upper blocks from the third block column of $\mathbf{Q}_{GI/M/1}$ and substituting the rest with one block that results as a linear combination of the remaining lower blocks of the third block column in $\mathbf{Q}_{GI/M/1}$ plus all remaining blocks in $\mathbf{Q}_{GI/M/1}$ (i.e., from the fourth block column of $\mathbf{Q}_{GI/M/1}$ onwards). Substituting one (any) of these columns with a column of 1s, we obtain the rank of $m + 2n$. □

4.1. Computing Measures of Interest for GI/M/1-type Processes

For the GI/M/1-type processes as for the M/G/1-type processes, ETAQA allows the computation of the reward rate of state $s_i^{(j)}$, for $j \geq 2$ and $i = 1, \ldots, n$, if it is a polynomial of degree k in j with arbitrary coefficients $\mathbf{a}_i^{[0]}, \mathbf{a}_i^{[1]}, \ldots, \mathbf{a}_i^{[k]}$:

$$\forall j \geq 2,\ \forall i \in \{1, 2, \ldots, n\}, \qquad \boldsymbol{\rho}_i^{(j)} = \mathbf{a}_i^{[0]} + \mathbf{a}_i^{[1]} j + \cdots + \mathbf{a}_i^{[k]} j^k.$$

We follow the exact same steps as those presented in Section 3.1. $\mathbf{r}^{[k]}$ is obtained by solving the system of linear equations

$$\mathbf{r}^{[k]}[\mathbf{F}+\mathbf{L}+\sum_{i=1}^{\infty}\mathbf{R}^{i-1}\mathbf{B}^{(i)})^{\diamond} \mid ((\mathbf{I}-\mathbf{R})\sum_{j=2}^{\infty}\sum_{i=j}^{\infty}\mathbf{R}^{i-2}\cdot\widehat{\mathbf{B}}^{(i)}+\sum_{j=1}^{\infty}\sum_{i=j}^{\infty}\mathbf{R}^{i-1}\mathbf{B}^{(i)}-\mathbf{F})\cdot\mathbf{1}^T] = [(\mathbf{b}^{[k]})^{\diamond} \mid c^{[k]}], \tag{35}$$

where

$$\mathbf{b}^{[k]} = -\left(\boldsymbol{\pi}^{(0)} \cdot 2^k\widehat{\mathbf{F}} + \boldsymbol{\pi}^{(1)} \cdot (2^k\mathbf{L} + 3^k\mathbf{F}) + \sum_{l=1}^{k} \binom{k}{l} \mathbf{r}^{[k-l]} \cdot (\mathbf{L} + 2^l\mathbf{F})\right)$$

and

$$c^{[k]} = -\left(2^k\boldsymbol{\pi}^{(1)}\mathbf{F} + \sum_{l=1}^{k} \binom{k}{l} \mathbf{r}^{[k-l]}(\sum_{j=2}^{\infty}\sum_{i=j}^{\infty}((i-2)^l\mathbf{I} - (i-1)^l\mathbf{R})\mathbf{R}^{i-2}\widehat{\mathbf{B}}^{(i)} + \sum_{j=0}^{\infty} j^l \sum_{i=j}^{\infty}\mathbf{R}^i\mathbf{B}^{(i+1)} - \mathbf{F})\right)\mathbf{1}^T.$$

The $n \times n$ matrix used in Eq.(35) has full rank. The proof follows the same steps as those used for the proof of Theorem (2) and is omitted here for the sake of brevity.

4.2. Time and Storage Complexity

In this section, we present a detailed comparison of our ETAQA-GI/M/1 solution for GI/M/1-type processes with the matrix geometric solution outlined in Section 2.2. The complexity analysis is within the accuracy of O-notation. We assume that up to p of the $\widehat{\mathbf{B}}^{(j)}$, and $\mathbf{B}^{(j)}$, $j \geq 1$ matrices are stored. The notation in this section follows the one defined in Section 3.2.

We outline the required steps for each method and analyze the computation and storage complexity of each step up to the computation of the expected queue length. Since both methods require $\mathbf{R}$, we do not include this cost in our analysis and assume that is computed using an efficient method.

Analysis of ETAQA-GI/M/1 solution:

- Computation of the aggregate stationary probability vectors $\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(*)}$
  - $O(p \cdot (m \cdot \eta\{\widehat{\mathbf{B}}, \mathbf{R}\} + n \cdot \eta\{\mathbf{B}, \mathbf{R}\}))$ for the computation of sums of the form $\sum_{i=2}^{\infty} \mathbf{R}^{i-j} \cdot (\mathbf{I} - \mathbf{R}) \cdot \widehat{\mathbf{B}}^{(i)}$ for $j = 1, 2$, and $\sum_{i=1}^{\infty} \mathbf{R}^{i-j} \cdot (\mathbf{I} - \mathbf{R}) \cdot \mathbf{B}^{(i)}$ for $j = 0, 1$.
  - $O^L(\eta\{\widehat{\mathbf{L}}, \widehat{\mathbf{F}}, \widehat{\mathbf{B}}, \mathbf{L}, \mathbf{B}, \mathbf{R}\})$ for the solution of a system of $m + 2n$ linear equations.
- Storage requirements for computation of $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ and $\boldsymbol{\pi}^{(*)}$
  - $O(m \cdot n + n^2)$ to store sums of form $\sum_{i=2}^{\infty} \mathbf{R}^{i-j} \cdot (\mathbf{I} - \mathbf{R}) \cdot \widehat{\mathbf{B}}^{(i)}$ for $j = 1, 2$ and $\sum_{i=1}^{\infty} \mathbf{R}^{i-j} \cdot (\mathbf{I} - \mathbf{R}) \cdot \mathbf{B}^{(i)}$ for $j = 0, 1$.
  - n^2 to store matrix $\mathbf{R}$.
  - $m + 2n$ to store $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ and $\boldsymbol{\pi}^{(*)}$.
- Computation of the queue length

- $O^L(\eta\{\mathbf{F}, \mathbf{L}, \mathbf{B}, \mathbf{R}\})$ to solve a system of n linear equations.
- $O(p^2(m \cdot \eta\{\widehat{\mathbf{B}}, \mathbf{R}\} + n \cdot \eta\{\mathbf{B}, \mathbf{R}\})$ for the sums required to construct the matrices of the system of linear equations.

- Storage requirements for computation of queue length
  - No additional requirements.

Analysis of matrix-geometric solution:

- Computation of the boundary stationary probability vectors $\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(1)}$
  - $O(p \cdot (m \cdot \eta\{\widehat{\mathbf{B}}, \mathbf{R}\} + n \cdot \eta\{\mathbf{B}, \mathbf{R}\}))$ to compute sums of the form $\sum_{i=2}^{\infty} \mathbf{R}^{i-j}\widehat{\mathbf{B}}^{(i)}$ for $j = 1, 2$ and $\sum_{i=1}^{\infty} \mathbf{R}^{i-j}\mathbf{B}^{(i)}$ for $j = 0, 1$.
  - $O(n^3)$ to compute of $(\mathbf{I} - \mathbf{R})^{-1}$.
  - $O^L(\eta\{\widehat{\mathbf{L}}, \widehat{\mathbf{F}}, \widehat{\mathbf{B}}, \mathbf{L}, \mathbf{F}, \mathbf{B}, \mathbf{R}\})$ for the solution of a system of $m + n$ linear equations.

- Storage requirements for computation of $\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(1)}$
  - $O(m \cdot n + n^2)$ to store sums of the form $\sum_{i=2}^{\infty} \mathbf{R}^{i-j}\widehat{\mathbf{B}}^{(i)}$ for $j = 1, 2$ and $\sum_{i=1}^{\infty} \mathbf{R}^{i-j}\mathbf{B}^{(i)}$ for $j = 0, 1$.
  - $O(n^2)$ to store $\mathbf{R}$ and $(\mathbf{I} - \mathbf{R})^{-1}$.
  - $m + n$ to store $\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(1)}$.

- Computation of queue length
  - $O(n^2)$ to compute the closed-form formula for queue length: $\boldsymbol{\pi}^{(1)} \cdot \mathbf{R} \cdot (\mathbf{I} - \mathbf{R})^{-2} \cdot \mathbf{1}^T$.

- Storage requirements for computation of queue length
  - No additional requirements.

Tables 3 and 4 summarize the computational and storage costs of the two methods.

We conclude this section that noting that the appeal of the classic matrix geometric method is its simplicity. The geometric relation between the vectors of the stationary probability distribution allows for simple closed form formulas for the computation of measures of interest such as the system queue length. The ETAQA-GI/M/1 method performs better when we are only interested in the computation of the probability vectors, depending on the

Table 3: Computational Complexities of ETAQA-GI/M/1 and Matrix Geometric

Computation of $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ (matrix geometric) or $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, $\boldsymbol{\pi}^{(*)}$ (ETAQA-GI/M/1)	
ETAQA-GI/M/1	$O^L(\eta\{\widehat{\mathbf{L}}, \widehat{\mathbf{F}}, \widehat{\mathbf{B}}, \mathbf{L}, \mathbf{F}, \mathbf{B}, \mathbf{R}\}) + O(p \cdot (m \cdot \eta\{\widehat{\mathbf{B}}, \mathbf{R}\} + n \cdot \eta\{\mathbf{B}, \mathbf{R}\}))$
%hline Matrix-geometric	$O^L(\eta\{\widehat{\mathbf{L}}, \widehat{\mathbf{F}}, \widehat{\mathbf{B}}, \mathbf{L}, \mathbf{B}, \mathbf{R}\}) + O(p \cdot (m \cdot \eta\{\widehat{\mathbf{B}}, \mathbf{R}\} + n \cdot \eta\{\mathbf{B}, \mathbf{R}\}) + n^3)$
First moment measures	
ETAQA-GI/M/1	$O^L(\eta\{\mathbf{F}, \mathbf{L}, \mathbf{B}, \mathbf{R}\}) + O(p^2(m \cdot \eta\{\widehat{\mathbf{B}}, \mathbf{R}\} + n \cdot \eta\{\mathbf{B}, \mathbf{R}\})$
Matrix-geometric	$O(n^2)$

Table 4: Storage Complexities of ETAQA-GI/M/1 and Matrix Geometric

	Additional storage	**Storage of the probabilities**
Computation of $\boldsymbol{\pi}^{(0)}$ (matrix-geometric) or $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, $\boldsymbol{\pi}^{(*)}$ (ETAQA-GI/M/1)		
ETAQA-GI/M/1	$O(m \cdot n + n^2)$	$m + 2n$
Matrix-geometric	$O(m \cdot n + n^2)$	$m + n$
First moment measures		
ETAQA-GI/M/1	none	*none*
Matrix-geometric	none	*none*

system sparsity, the size of matrices, and the number of stored matrices that capture the behavior of the whole process, but not when we are interested in computing measures of interest.

5. ETAQA for QBD Processes

Quasi-birth-death (QBD) processes are essentially a subcase of both M/G/1-type and GI/M/1-type processes and can be therefore solved with either the matrix analytic method outlined in Section 2.1 or the matrix geometric method outlined in Section 2.2. Of the two methods, the method of choice for the solution of QBD processes is matrix geometric because of its simplicity and its ability to provide closed form formulas for measures of interest such as the expected queue length. In contrary to the matrix analytic methods that solve QBDs using matrix geometric solution, we choose to solve QBDs using the ETAQA-M/G/1 because from the complexity analysis presented in subsections 3.2 and 4.2 we have concluded that ETAQA-M/G/1 is more efficient.

Assuming the knowledge of matrix $\mathbf{G}$ for a QBD process with the infinitesimal generator as shown in Eq.(11), the proposed aggregate solution for the QBD process is stated in the following theorem:

Theorem 3 *Given an ergodic CTMC with infinitesimal generator* $\mathbf{Q}_{QBD}$ *having the structure shown in Eq.(11), with stationary probability vector* $\boldsymbol{\pi} = [\boldsymbol{\pi}^{(0)}, \ \boldsymbol{\pi}^{(1)}, \ \boldsymbol{\pi}^{(2)}, \ ...]$ *the system of linear equations*

$$\mathbf{x} \cdot \mathbf{X} = [1, \mathbf{0}], \tag{36}$$

where $\mathbf{X} \in \mathbb{R}^{(m+2n)\times(m+2n)}$ *is defined as follows*

$$\mathbf{X} = \left[\begin{array}{c|c|c|c} \mathbf{1}^T & \widehat{\mathbf{L}} & \widehat{\mathbf{F}} & \mathbf{0}^{\diamond} \\ \mathbf{1}^T & \widehat{\mathbf{B}} & \mathbf{L} & \mathbf{F}^{\diamond} \\ \mathbf{1}^T & \mathbf{0} & \mathbf{B} - \mathbf{F}\cdot\mathbf{G} & (\mathbf{L} + \mathbf{F} + \mathbf{F}\cdot\mathbf{G})^{\diamond} \end{array} \right], \tag{37}$$

admits a unique solution $\mathbf{x} = [\boldsymbol{\pi}^{(0)}, \ \boldsymbol{\pi}^{(1)}, \ \boldsymbol{\pi}^{(*)}]$, *where* $\boldsymbol{\pi}^{(*)} = \sum_{i=2}^{\infty} \boldsymbol{\pi}^{(i)}$.

Proof. The steps in the proof are identical to the steps in the Proof of Theorem 1 since QBDs are special case of M/G/1-type processes. □

5.1. Computing Measures of Interest for QBD Processes

Similarly to the M/G/1 case, ETAQA allows the computation of the reward rate of state $s_i^{(j)}$, for $j \geq 2$ and $i = 1, \ldots, n$, if it is a polynomial of degree k in j with arbitrary coefficients $\mathbf{a}_i^{[0]}, \mathbf{a}_i^{[1]}, \ldots, \mathbf{a}_i^{[k]}$:

$$\forall j \geq 2, \ \forall i \in \{1, 2, \ldots, n\}, \qquad \boldsymbol{\rho}_i^{(j)} = \mathbf{a}_i^{[0]} + \mathbf{a}_i^{[1]} j + \cdots + \mathbf{a}_i^{[k]} j^k.$$

Here, we follow the exact same steps as in Section 3.1, albeit significantly simplified. Observe that that $\mathbf{r}^{[0]}$ is simply $\boldsymbol{\pi}^{(*)}$ while, for $k > 0$, $\mathbf{r}^{[k]}$ can be computed after having obtained $\mathbf{r}^{[l]}$ for $0 \leq l < k$, by solving the system of n linear equations:

$$\begin{cases} \mathbf{r}^{[k]}(\mathbf{B} + \mathbf{L} + \mathbf{F})^{\diamond} & = & \mathbf{b}^{[k]\diamond} \\ \mathbf{r}^{[k]}(\mathbf{F} - \mathbf{B})\mathbf{1}^T & = & c^{[k]} \end{cases}, \tag{38}$$

where

$$\begin{aligned} \mathbf{b}^{[k]} &= -\left(2^k \boldsymbol{\pi}^{(0)} \cdot \widehat{\mathbf{F}} + 2^k \boldsymbol{\pi}^{(1)} \cdot \mathbf{L} + 3^k \boldsymbol{\pi}^{(1)} \cdot \mathbf{F} + \sum_{l=1}^{k} \binom{k}{l} \left(2^l \mathbf{r}^{[k-l]} \cdot \mathbf{F} + \mathbf{r}^{[k-l]} \cdot \mathbf{L}\right)\right), \\ c^{[k]} &= -2^k \boldsymbol{\pi}^{(1)} \mathbf{F} \mathbf{1}^T - \sum_{l=1}^{k} \binom{k}{l} \mathbf{r}^{[k-l]} \cdot \mathbf{F} \cdot \mathbf{1}^T. \end{aligned}$$

The rank of the system of linear equations depicted in Eq.(38) is n, since QBDs are a special case of M/G/1-type processes.

We conclude by reiterating that in order to compute the k^{th} moment of the queue length we must solve k linear systems in n unknowns each and, in particular, the expected queue length is obtained by solving just one linear system in n unknowns.

5.2. Time and Storage Complexity

In this section, we present a detailed comparison of our aggregate solution for QBD processes with the matrix geometric method for QBDs outlined in Section 2.3. The notation in this section follows the one defined in section 3.2.

We outline the required steps for each method and analyze the computation and storage complexity of each step up to the computation of the expected queue length. We assume that the algorithm of choice for computation of $\mathbf{R}$ in the matrix geometric solution for QBDs is logarithmic reduction as the most efficient one. Therefore in our analysis we do not include the cost to compute matrix $\mathbf{G}$, which is the first matrix to be computed by the logarithmic reduction algorithm of Latouche and Ramaswami (1999).

Analysis of ETAQA-QBD:

- Computation of the aggregate stationary probability vector $[\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(*)}]$
  - $O(n \cdot \eta\{\mathbf{F}, \mathbf{G}\})$ to compute $\mathbf{FG}$.
  - $O^L(\widehat{\mathbf{L}}, \widehat{\mathbf{F}}, \widehat{\mathbf{B}}, \mathbf{B}, \mathbf{L}, \mathbf{F}, \mathbf{G})$ to solve the system of $m + 2n$ linear equations.
- Storage requirements for computation of $[\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(*)}]$
  - $O(n^2)$ for matrix $\mathbf{FG}$.
  - $m + 2n$ for the vector $[\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(*)}]$.
- Computation of the queue length
  - $O(\eta\{\mathbf{F}, \mathbf{L}, \mathbf{B}\})$ to compute $\mathbf{F} + \mathbf{L} + \mathbf{B}$ and $\mathbf{F} - \mathbf{B}$.
  - $O^L(\eta\{\mathbf{F}, \mathbf{L}, \mathbf{B}\})$ to solve a system of n linear equations.
- Storage requirements for the computation of the queue length
  - No additional storage.

Analysis of matrix geometric for QBDs:

- Computation of the boundary stationary probability vector $[\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}]$
  - $O(n^3)$ to compute $\mathbf{R}$ from $\mathbf{G}$ (last step of the logarithmic reduction algorithm) using the relation $\mathbf{R} = -\mathbf{F}(\mathbf{L} + \mathbf{FG})^{-1}$ (see the Online Supplement).

- $O(n^3)$ to compute $(\mathbf{I}-\mathbf{R})^{-1}$.
- $O(n \cdot \eta\{\mathbf{R},\mathbf{B}\}$ to compute $\mathbf{RB}$.
- $O^L(\widehat{\mathbf{L}},\widehat{\mathbf{F}},\widehat{\mathbf{B}},\mathbf{L},\mathbf{B},\mathbf{R})$ for the solution of the system of $m+n$ linear equations to obtain $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$. The required storage for the probability vectors $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ is exactly $m+n$.

- Storage requirements to compute $[\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}]$
  - $O(n^2)$ for matrix $\mathbf{R}$ and $(\mathbf{I}-\mathbf{R})^{-1}$.
  - $m+n$ to store $\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(1)}$.
- Computation of the queue length
  - $O(n^2)$ to compute queue length from $\boldsymbol{\pi}^{(1)} \cdot \mathbf{R} \cdot (\mathbf{I}-\mathbf{R})^{-2} \cdot \mathbf{1}^T$.
- Storage requirements to compute queue length
  - No additional storage.

Tables 5 and 6 summarize the discussion in this section.

Table 5: Computational Complexities of ETAQA-QBD and Matrix Geometric

Computation of $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ (matrix geometric) or $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, $\boldsymbol{\pi}^{(*)}$ (ETAQA-QBD)	
ETAQA-QBD	$O^L(\eta\{\widehat{\mathbf{L}},\widehat{\mathbf{B}},\widehat{\mathbf{F}},\mathbf{L},\mathbf{F},\mathbf{B},\mathbf{G}\}) + O(n \cdot \eta\{\mathbf{F},\mathbf{G}\}))$
Matrix geometric	$O^L(\eta\{\widehat{\mathbf{L}},\widehat{\mathbf{B}},\widehat{\mathbf{F}},\mathbf{L},\mathbf{B},\mathbf{R}\}) + O(n^3) + O(n \cdot \eta\{\mathbf{R},\mathbf{B}\}$
First moment measures	
ETAQA-QBD	$O^L(\eta\{\mathbf{B},\mathbf{L},\mathbf{F}\}) + O(\eta(\mathbf{B},\mathbf{L},\mathbf{F}))$
Matrix geometric	$O(n^2)$

Table 6: Storage Complexities of ETAQA-QBD and Matrix Geometric

	Additional storage	**Storage of the probabilities**
Computation of $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$ (matrix geometric) or $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, $\boldsymbol{\pi}^{(*)}$ (ETAQA-QBD)		
ETAQA-QBD	$O(n^2)$	$m+2n$
Matrix geometric	$O(n^2)$	$m+n$
First moment measures		
ETAQA-QBD	none	*none*
Matrix geometric	none	*none*

We emphasize the fact that the sparsity of $\mathbf{G}$ is key to preserving the sparsity of the original process in the ETAQA-QBD method, while the $\mathbf{R}$ that is required in matrix-geometric is

Figure 3: Execution Times in Seconds

usually full. Concluding our analysis, we note that the ETAQA-QBD solution is as efficient as the matrix geometric method. We note that storage-wise we do gain (although this gain is not obvious using O-notation) because the aggregate solution requires only *temporal* storage of the matrix $\mathbf{F} \cdot \mathbf{G}$, while the matrix geometric method needs *persistent* storage of $\mathbf{R}$ and $(\mathbf{I} - \mathbf{R})^{-1}$.

6. Computational efficiency

In the previous section, we argue using O-notation about the the computational and storage efficiency of ETAQA-M/G/1. Here, we present further numerical evidence that ETAQA-M/G/1 is more efficient than other methods. For our comparisons, we use the classic Ramaswami's formula and Meini (1997b)'s fast FFT implementation of Ramaswami's formula, the most efficient known algorithm for solving M/G/1-type processes. We used Meini's implementation available at `http://www.dm.unipi.it/~meini/ric.html`. for the cyclic reduction for the computation of $\mathbf{G}$ that is required in all three solution algorithms. We also used Meini's code for the fast FFT implementation of Ramaswami's formula that was made available to us via a personal communication (Meini (1997a)). We implemented the ETAQA-M/G/1 method and the classic Ramaswami's formula in C. All experiments were conducted on a 450 MHz Sun Enterprise 420R server with 4 GB memory.

The chain we selected for our experiments is a general BMAP/M/1 queue. Recall that in practice, it is not possible to store an infinite number of $\widehat{\mathbf{F}}^{(i)}$ and $\mathbf{F}^{(i)}$ matrices, $1 < i < \infty$. One should stop storing when all entries of $\widehat{\mathbf{F}}^{(i)}$ and $\mathbf{F}^{(i)}$ for $i > p$ are below a sufficient threshold (i.e., *very close* to zero in a practical implementation), as suggested in Latouche and Ramaswami (1999). As illustrated in the previous section, computation time does depend on the size (i.e., parameters m and n) and the number (of stored) matrices (i.e., parameter p) that define the infinitesimal generator $\mathbf{Q}$. Finally, one last parameter that affects computation time is the number s of vector probabilities that should be computed so as the normalization condition $\sum_{i=1}^{s} \boldsymbol{\pi}^{(i)} = 1.0$ is reached (again, within a sufficient numerical threshold).

In our experiments, we vary the parameters n, p, and s (for the case of BMAP/M/1 queue $m = n$) and provide timing results for the computation of the stationary vector $\boldsymbol{\pi}$ using the classic Ramaswami implementation and the fast FFT implementation, and the computation of $(\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(*)})$ using ETAQA-M/G/1. We also provide timings for the computation of the queue length for both methods. Results are presented in Figure 3.

The first experiment, considers a BMAP/M/1 queue with $n = 16$ and $p = 32$, a relatively small case. The timings of the three algorithms, which do not take into consideration the computation of $\mathbf{G}$, are shown as a function of s. Figure 3(a) depicts the computation cost of the probability vector and Figure 3(b) illustrates the computation cost the queue length. Observe that the y-axis is in log-scale. Note that the value of s does affect the execution time of both Matrix-analytic approaches, but does not have any affect on ETAQA-M/G/1. As expected, for the computation of the stationary vector, the FFT implementation is superior to the classic Ramaswami formula, behavior that persists when we increase p and n (see Figures 3(c) and 3(e)). ETAQA-M/G/1 consistently outperforms the other two methods, plus its performance is insensitive to s (see Figures 3(a), 3(c) and 3(e)).

Figures 3(b), 3(d) and 3(f) illustrate the computation cost of the queue length for the three algorithms for various values of n, p, and s. Note that the two implementations of Ramaswami's formula have the same cost, since the same classic formula is used for the computation of queue length: first weight appropriately and then sum the probability vector which is already computed. The figures further confirm that the cost of solving a small system of linear equations that ETAQA-M/G/1 requires for the computations of queue length is in many cases preferable to the classic methods. If this linear system increases and s is also small, then the classic methods may offer better performance.

7. Concluding Remarks

In this paper, we presented ETAQA, an aggregate approach for the solution of M/G/1-type, GI/M/1-type, and QBD processes. Our exposition focuses on computing efficiently the *exact* probabilities of the boundary states of the process and the aggregate probability distribution of the states in each of the equivalence classes corresponding to a specific partitioning of the remaining infinite portion of the state space. Although the method does not compute the probability distribution of all states, it still provides enough information for the "mathematically exact" computation of a rich set of Markov reward functions such as the expected queue length or any of its higher moments.

We presented detailed analysis of the computation and storage complexity of our method.

We conclude that for the case of M/G/1-type processes ETAQA requires a few simple steps that provide significant savings with respect to both computation and storage when compared with the traditional matrix analytic and matrix geometric solutions, respectively. These gains are a direct outcome of the fact that ETAQA computes only the aggregate stationary probability vector instead of the entire stationary probability vector computed by the matrix-analytic methods. Additionally, ETAQA closely preserves the structure (thus the sparsity) of the original process, thus facilitating computational gains, in contrast to the classic methods that instead introduce structures that destroy the sparsity of the original matrices.

For the case of GI/M/1-type and QBD processes, ETAQA has the same complexity as the classic matrix geometric method for the computation of the stationary probability vector, albeit the classic method results in less expensive and more intuitively appealing formulas for the computation of measures of interest such as the expected queue length.

An issue that often arises in the area of numerical solutions of Markov chains is the method's numerical stability. The numerical stability of algorithms for the solution of processes that focus on in this paper has hardly been investigated, if at all, as stated in Latouche and Ramaswami (1999). The methods that offer a recursive computation of the probability vector via a formula that entails additions and multiplications, are considered numerically stable. Ramaswami's recursive formula for M/G/1-type processes is a classical case of a stable algorithm. Once subtractions are involved the possibility of numerical instability increases because of the loss of significance (as discussed in Neuts (1989, page 165)). Our construction of the matrix $\mathbf{X}$ in Eqs.(16) does introduce subtractions. Yet, we have strong experimental indications that ETAQA is stable. Examining theoretically the numerical stability of our methodology is subject of future work.

Acknowledgments

We thank Beatrice Meini for providing us with her implementation of the FFT algorithm for Ramaswami's recursive formula. We also thank Mark Squillante for insightful discussions and Guy Latouche for his valuable feedback in earlier versions of this work. This work was partially supported by the National Science Foundation under grants ITR-0428330, CCR-0098278, and ACI-0090221.

References

Bini, D. A., B. Meini. 1998. Using displacement structure for solving non-skip-free M/G/1 type Markov chains. A. S. Alfa, S. R. Chakravarthy, eds., *Advances in Matrix Analytic Methods for Stochastic Models*. Notable Publications Inc, NJ, 17–37.

Bini, D. A., B. Meini, V. Ramaswami. 2000. Analyzing M/G/1 paradigms through QBDs: the role of the block structure in computing the matrix G. G. Latouche, P. Taylor, eds., *Advances in Matrix Analytic Methods for Stochastic Models*. Notable Publications Inc, NJ, 73–86.

Ciardo, G., W. Mao, A. Riska, E. Smirni. 2004. ETAQA-MG1: An efficient technique for the analysis of M/G/1-type processes by aggregation. *Performance Evaluation Journal* **57** 235–260.

Ciardo, G., E. Smirni. 1999. ETAQA: An efficient technique for the analysis of QBD-processes by aggregation. *Performance Evaluation* **36-37** 71–93.

Grassman, W. K., D. A. Stanford. 2000. Matrix analytic methods. W. K. Grassman, ed., *Computational Probability*. Kluwer Academic Publishers, Boston, MA, 153–204.

Kleinrock, L. 1975. *Queueing Systems, Volume I: Theory*. Wiley.

Latouche, G. 1993. Algorithms for infinite Markov chains with repeating columns. C. Meyer, R. J. Plemmons, eds., *Linear Algebra, Markov Chains, and Queueing Models*, vol. 48. IMA Volumes in Mathematics and its Applications, Springer Verlag, 231–265.

Latouche, G., V. Ramaswami. 1999. *Introduction to Matrix Analytic Methods in Stochastic Modeling*. SIAM, Philadelphia PA. ASA-SIAM Series on Statistics and Applied Probability.

Latouche, G., G. W. Stewart. 1995. Numerical methods for M/G/1-type queues. W. J. Stewart, ed., *Computations with Markov Chains*. Kluwer Academic Publishers, Boston, MA, 571–581.

Lucantoni, D. M. 1983. *An algorithmic analysis of a communication model with retransmission of flawed messages*. Pitman, Boston.

Meini, B. 1997a. *Implementation of FFT-based version of Ramaswami's formula*. University of Pisa, Italy.

Meini, B. 1997b. An improved FFT-based version of Ramaswami's formula. *Comm. Statist. Stochastic Models* **13** 223–238.

Meini, B. 1998. Solving M/G/1-type Markov chains: Recent advances and applications. *Comm. Statist. Stochastic Models* **14** 479–496.

Nelson, R. 1995. *Probability, Stochastic Processes, and Queueing Theory*. Springer-Verlag.

Neuts, M. F. 1981. *Matrix-geometric Solutions in Stochastic Models*. Johns Hopkins University Press, Baltimore, MD.

Neuts, M. F. 1989. *Structured Stochastic Matrices of M/G/1-type and their Applications*. Marcel Dekker, New York, NY.

Ramaswami, V. 1988. A stable recursion for the steady state vector in Markov chains of M/G/1-type. *Commun. Statist. Stochastic Models* **4** 183–189.

Ramaswami, V., G. Latouche. 1986. A general class of Markov processes with explicit matrix-geometric solutions. *OR Spektrum* **8** 209–218.

Ramaswami, V., J. L. Wang. 1996. A hybrid analysis/simulation for ATM performance with application to quality-of-service of CBR traffic. *Telecommunication Systems* **5** 25–48.

Riska, A., E. Smirni. 2002a. Exact aggregate solutions for M/G/1-type Markov processes. *Proceedings of ACM SIGMETRICS Conference*. Marina del Rey, CA, 86–96.

Riska, A., E. Smirni. 2002b. M/G/1-type Markov processes: A tutorial. M. C. Calzarossa, S. Tucci, eds., *Performance Evaluation of Complex Computer Systems: Techniques and Tools, LNCS 2459*. Springer-Verlag, 36–63.

Squillante, M. S. 1998. Matrix-analytic methods in stochastic parallel-server scheduling models. S. R. Chakravarthy, A. S. Alfa, eds., *Advances in Matrix-Analytic Methods for Stochastic Models: Lecture Notes in Pure and Applied Mathematics*. Notable Publications, NJ.

Squillante, M. S. 2000. Matrix-analytic methods: Applications, results and software tools. G. Latouche, P. Taylor, eds., *Advances in Matrix-Analytic Methods for Stochastic Models*. Notable Publications, NJ.

Stathopoulos, A., A. Riska, Z. Hua, E. Smirni. 2005. Bridging ETAQA and Ramaswami's formula for the solution of M/G/1 processes. *Performance Evaluation Journal (to appear)* .