An aggregation-based solution method for M/G/1-type processes

Gianfranco Ciardo Alma Riska Evgenia Smirni*
Department of Computer Science
College of William and Mary
Williamsburg, VA 23187-8795
{ciardo,riska,esmirni}@cs.wm.edu

Abstract

We extend the ETAQA approach, initially proposed for the efficient numerical solution of a class of quasi birth-death processes, to the more complex case of M/G/1-type Markov processes. The proposed technique can be used for the exact solution of a class of M/G/1-type models by simply computing the solution of a finite linear system. We further demonstrate the utility of the method by describing the exact computation of an extensive set of Markov reward functions such as the expected queue length or its higher moments. We illustrate the method, discuss its complexity, present comparisons with other traditional techniques, and illustrate its applicability in the area of computer system modeling.

1 Introduction

In this paper, we consider Markov chains on an infinite state space having a M/G/1-type structure. Such processes often serve as the modeling tool of choice for modern computer and communication systems [7]. As a consequence, considerable effort has been placed into the development of exact analysis techniques for M/G/1-type processes. The infinitesimal generator of such processes (for the case of continuous time Markov chains) is upper block Hessenberg and follows a repetitive structure. Matrix-analytic methods have been proposed for their solution with most prominent the one developed by Neuts [9]. The key in the matrix-analytic solution is the computation of an auxiliary matrix called **G**. Similarly, for Markov chains the of GI/M/1-type, which have a lower block Hessenberg form, matrix-geometric solutions have been proposed [8]. Again, the key in the matrix-geometric solution is the computation of an auxiliary matrix, called **R**. Traditionally, iterative procedures are used for the determination of both matrices. Alternative algorithms for the computation of **G** (and **R**) have been proposed. We note here the work of Latouche [4] for efficient algorithms for determining the matrix **R**. The same algorithms can be used to compute the matrix **G** [4]. Other methods for the computation of **G** (and **R**) using a recursive descent method have been proposed [14] and compared with the traditional methods [5].

The primary distinction between our research and the above works is that we restrict our attention to a family of M/G/1-type processes with a specific form, for which "returns" from a

*This research has been supported by a William and Mary Summer Research Grant.

higher level of states to the immediate lower level are always directed toward a single state (for such a subclass, the computation of the matrix $\mathbf{G}$ is trivial, but the remaining solution steps using standard methods are still expensive). We instead recast the problem to the one of solving a finite linear system of $m+2n$ equations 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 are able to obtain "exact" results.

Our approach is an extension of the ETAQA method that was initially proposed for the efficient solution of quasi-birth-death processes with matrix-geometric form [2]. The proposed methodology uses basic well-known results for Markov chains. We exploit the structure of the repetitive portion of the chain and instead of evaluating the probability distribution of *all* states in the chain, we calculate the *aggregate* probability distribution of n equivalence classes, appropriately defined. The extended ETAQA approach is both efficient and exact, allowing us to compute the probabilities of the boundary states as well as the aggregate probability of the n equivalence classes; it also provides the ability to efficiently compute reward rates of interests such as the k^{th} moment of the queue length.

This paper is organized as follows. Section 2 presents initial terminology and summarizes the utility of using ETAQA for the solution of quasi-birth-death processes. In Section 3 we present the basic theorem that extends ETAQA to M/G/1-type processes. We demonstrate how the methodology can be used for the computation of Markov reward functions in Section 4. We continue by showing the applicability of extended ETAQA to bounded bulk arrivals in Section 5. In Section 6 we compare the computation and storage complexity of the extended ETAQA with Neuts's algorithm [9] for the analysis of M/G/1 processes. Section 7 presents two applications from the computer systems area that can be analyzed using the extended ETAQA. Finally, Section 8 summarizes our findings and outlines future work.

2 Background: using ETAQA for the solution of QBD processes

In this section, we briefly review the basic terminology used to describe the class of processes we consider, as well as our previous results on ETAQA. In our exposition, we restrict ourselves to the case of continuous-time Markov chains (hence we refer to the infinitesimal generator matrix $\mathbf{Q}$), but the theory can just as well be applied to the discrete case.

2.1 Markov chains with repetitive structure

Neuts [8] defines various classes of infinite-state Markov chains with a repetitive structure. In all cases, the state space 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 $j \geq 1$:

- GI/M/1-type Markov chains, whose infinitesimal generator can be block-partitioned as:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{L}^{(0)} & \hat{\mathbf{F}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ \hat{\mathbf{B}}^{(1)} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ \hat{\mathbf{B}}^{(2)} & \mathbf{B}^{(1)} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \mathbf{0} & \cdots \\ \hat{\mathbf{B}}^{(3)} & \mathbf{B}^{(2)} & \mathbf{B}^{(1)} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \cdots \\ \hat{\mathbf{B}}^{(4)} & \mathbf{B}^{(3)} & \mathbf{B}^{(2)} & \mathbf{B}^{(1)} & \mathbf{L} & \mathbf{F} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

- M/G/1-type Markov chains, whose infinitesimal generator can be block-partitioned as:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{L}^{(0)} & \hat{\mathbf{F}}^{(1)} & \hat{\mathbf{F}}^{(2)} & \hat{\mathbf{F}}^{(3)} & \hat{\mathbf{F}}^{(4)} & \hat{\mathbf{F}}^{(5)} & \cdots \\ \hat{\mathbf{B}} & \mathbf{L} & \mathbf{F}^{(1)} & \mathbf{F}^{(2)} & \mathbf{F}^{(3)} & \mathbf{F}^{(4)} & \cdots \\ \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F}^{(1)} & \mathbf{F}^{(2)} & \mathbf{F}^{(3)} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F}^{(1)} & \mathbf{F}^{(2)} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F}^{(1)} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

- and quasi-birth-death (QBD) Markov chains, essentially the intersection of the two previous cases, whose infinitesimal generator can be block-partitioned as:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{L}^{(0)} & \hat{\mathbf{F}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ \hat{\mathbf{B}} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

(we use the letter "L", "F", and "B" according to whether the matrices describe "local", 'forward", and "backward" transition rates, respectively, and we use a "^" for matrices related to $\mathcal{S}^{(0)}$).

The *matrix-geometric* approach [8] can be used to study GI/M/1-type processes. If $\boldsymbol{\pi}^{(j)}$ is the stationary probability vector for states in $\mathcal{S}^{(j)}$, $j \geq 0$, we can write

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

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

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

Iterative numerical algorithms can be used to compute $\mathbf{R}$. Then, we can write

$$[\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}] \cdot \begin{bmatrix} \mathbf{L}^{(0)} & \hat{\mathbf{F}}^{(1)} \\ \sum_{k=1}^{\infty} \mathbf{R}^{k-1} \cdot \hat{\mathbf{B}}^{(k)} & \mathbf{L} + \sum_{k=1}^{\infty} \mathbf{R}^{k} \cdot \mathbf{B}^{(k)} \end{bmatrix} = \mathbf{0}$$

which, together with the normalization condition

$$\boldsymbol{\pi}^{(0)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(1)} \cdot \sum_{j=1}^{\infty} \mathbf{R}^{j-1} \cdot \mathbf{1}^T = 1 \qquad \text{that is} \qquad \boldsymbol{\pi}^{(0)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(1)} \cdot (\mathbf{I} - \mathbf{R})^{-1} \cdot \mathbf{1}^T = 1,$$

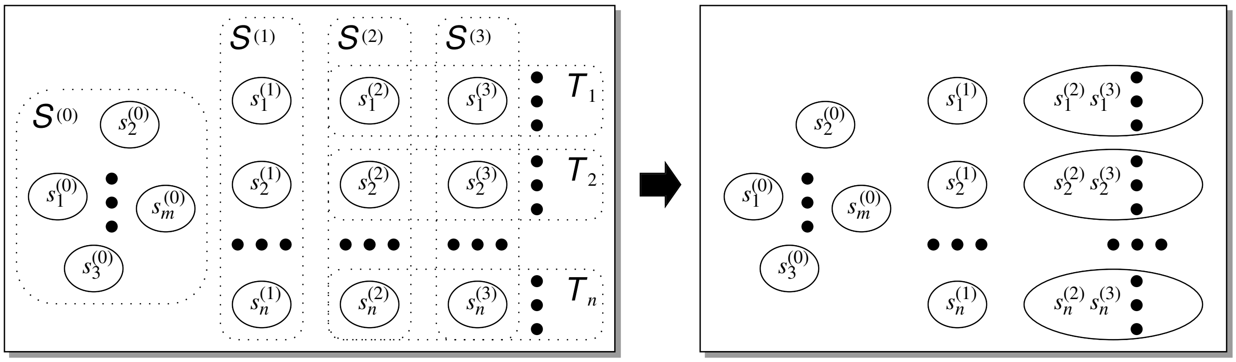

Figure 1: Aggregation of an infinite $\mathcal{S}$ into a finite number of states.

yields a unique solution for $\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(1)}$. For $j \geq 2$, $\boldsymbol{\pi}^{(j)}$ can be obtained numerically from (1). More importantly, though, many useful performance metrics, such as expected system utilization, throughput, or queue length, can be computed exactly in explicit form from $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and $\mathbf{R}$ alone.

2.2 ETAQA

The matrix-geometric approach is directly applicable to QBD processes as well, since QBD processes are a special case of GI/M/1-type processes. In [2] we presented ETAQA, an alternative approach that can be used to solve a subclass of QBD processes more efficiently than the classic matrix geometric technique.

To apply ETAQA, $\mathbf{B}$ must contain nonzero entries in only one of its columns (which by convention we number last, n). This effectively means that all transitions from $\mathcal{S}^{(j)}$ to $\mathcal{S}^{(j-1)}$ are restricted to go to $s_n^{(j-1)}$. When this condition holds, it is possible to derive a system of $m + 2n$ independent linear equations in $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and $\boldsymbol{\pi}^{(*)}$, a new vector of n unknowns representing the stationary probability of being in the sets of states $\mathcal{T}_i = \{s_i^{(j)} : j \geq 2\}$, for $i = 1, \ldots, n$, i.e., $\boldsymbol{\pi}^{(*)} = \sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)}$. Fig. 1 illustrates how this approach can be thought of as aggregating the states of $\mathcal{T}_i$ into a single macro-state.

Any stationary measure expressed as the expected reward rate can then be computed, as long as the reward rate of state $s_i^{(j)}$ is a polynomial of finite degree k in j, with coefficients that can depend on i. Then, the computation of the expected reward rate requires the solution of k linear systems in n unknowns. In practice, simple measures such as average queue length (or its variance) require only one (or two) linear system solutions, in addition to the solution of the initial linear system in $m + 2n$ unknowns. The matrix $\mathbf{Q}$ is highly sparse in practical problems, and so are the matrices describing the linear systems solved by ETAQA, with beneficial implications to the overall computational and storage complexity.

We stress that the special structure of $\mathbf{B}$ required by ETAQA does not enforce any special structure in $\mathbf{R}$ itself: $\mathbf{R}$ can still be completely full and, since $\mathbf{L}$ and $\mathbf{F}$ are completely general, there is no simple relation between its entries. The fact that $\mathbf{B}$ contains only a single nonzero column does reduce the computational requirements of the matrix-geometric method from $O(I \cdot n^3)$ [4] to $O(n^3 + I \cdot n^2)$ [2], where I is the number of iterations needed to achieve convergence to a

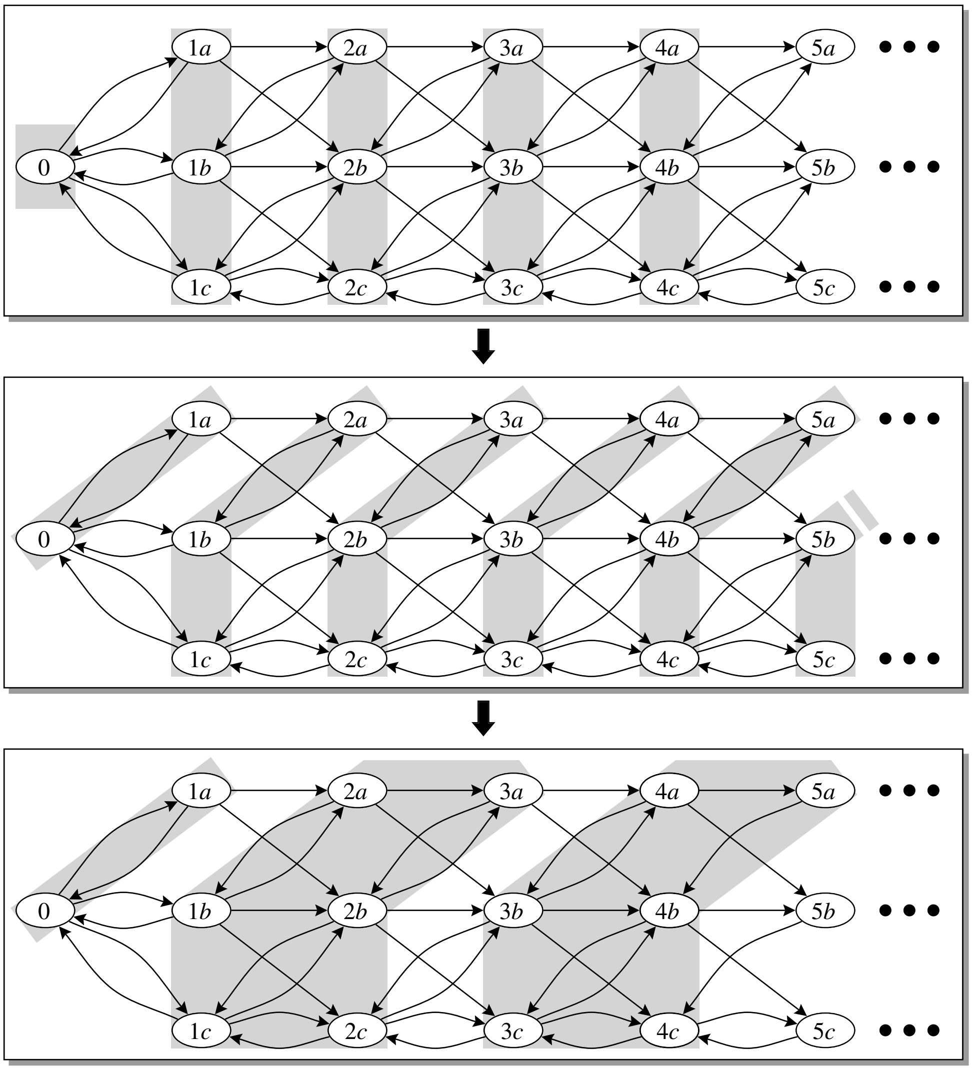

Figure 2: State repartitioning and merging.

given tolerance for $\mathbf{R}$. However, for large values of n, the computational cost is still high and, even more importantly, the storage requirements of the matrix-geometric method are still $O(n^2)$.

2.3 Limitations of ETAQA

The approach we introduced in [2] is very efficient, but its applicability is limited by the requirement that $\mathbf{B}$ has only one nonzero column. There are models where this condition on $\mathbf{B}$ is not immediately satisfied yet it can be achieved through an appropriate *repartitioning* of the states. However, a repartitioning might in turn destroy the QBD structure.

For example, Fig. 2 (top) shows a QBD process where the transitions from $\mathcal{S}^{(j)}$ to $\mathcal{S}^{(j-1)}$ can have two destinations, $s_b^{(j-1)}$ and $s_c^{(j-1)}$. This would appear to prevent us from applying ETAQA. Fig. 2 (middle) shows instead that, by redefining the sets $\mathcal{S}^{(j)}$ in such a way that $\mathcal{S}^{(1)} = \{s_a^{(2)}, s_b^{(1)}, s_c^{(1)}\}$, $\mathcal{S}^{(2)} = \{s_a^{(3)}, s_b^{(2)}, s_c^{(2)}\}$, etc., the transitions from $\mathcal{S}^{(j)}$ to $\mathcal{S}^{(j-1)}$ now go to a single state, $s_c^{(j-1)}$. The condition on $\mathbf{B}$ is then satisfied, but, in this case, the resulting a process is not QBD anymore, and therefore we cannot apply ETAQA as defined in [2], since transitions from $s_a^{(j)}$ to $s_b^{(j+1)}$ now span two levels (i.e., from $\mathcal{S}^{(j-1)}$ to $\mathcal{S}^{(j+1)}$).

However, if a process has a repeated structure with forward jumps from $\mathcal{S}^{(j)}$ up to $\mathcal{S}^{(j+p)}$ for a finite $p > 1$, it can still be treated as a QBD process provided that we merge p levels at a time into a new larger single level. For example, in Fig. 2 (bottom), we merge the old $\mathcal{S}^{(1)}$ and $\mathcal{S}^{(2)}$ into a new $\mathcal{S}^{(1)}$, the old $\mathcal{S}^{(3)}$ and $\mathcal{S}^{(4)}$ into a new $\mathcal{S}^{(2)}$, and so on. Then, again, transitions are only between adjacent levels. Note that this merging technique can be applied even if the multi-level forward jumps are not created by a state repartitioning but are already present in the initial model, as in the case of a queue with *bulk arrivals* of maximum size p.

The price paid for these state space manipulations has two components. First, the cost of repartitioning the classes to ensure that $\mathbf{B}$ contains a single column: at the moment, we perform this step "by hand", so this is clearly a topic for future work. Second, merging p classes into one has the effect of increasing the complexity of ETAQA by a factor p: a linear system in $m + 2pn$ variables has to be solved, and the k linear systems that must be solved to compute the expected reward rates have now pn variables.

In this paper, we extend ETAQA so that it applies to M/G/1-type processes, still subject to the restriction that $\mathbf{B}$ is a matrix with a single nonzero column. This allows us to study systems with unbounded bulk arrivals. In addition, we also extend ETAQA to bounded bulk arrivals, without having to merge classes of states; this effectively eliminates the additional complexity factor p just discussed.

3 Extending ETAQA to M/G/1-type processes

The processes we consider have the following structure in the infinitesimal generator matrix $\mathbf{Q}$:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{L}^{(0)} & \hat{\mathbf{F}}^{(1)} & \hat{\mathbf{F}}^{(2)} & \hat{\mathbf{F}}^{(3)} & \hat{\mathbf{F}}^{(4)} & \hat{\mathbf{F}}^{(5)} & \cdots \\ \hat{\mathbf{B}} & \mathbf{L}^{(1)} & \mathbf{F}^{(1)} & \mathbf{F}^{(2)} & \mathbf{F}^{(3)} & \mathbf{F}^{(4)} & \cdots \\ \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F}^{(1)} & \mathbf{F}^{(2)} & \mathbf{F}^{(3)} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F}^{(1)} & \mathbf{F}^{(2)} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{B} & \mathbf{L} & \mathbf{F}^{(1)} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \tag{2}$$

where $\mathbf{L}^{(0)} \in \mathbb{R}^{m\times m}$, $\mathbf{L}^{(1)} \in \mathbb{R}^{n\times n}$, and $\mathbf{L} \in \mathbb{R}^{n\times n}$ represent the local transition rates between states in $\mathcal{S}^{(0)}$, $\mathcal{S}^{(1)}$, and $\mathcal{S}^{(j)}$ for $j \geq 2$, respectively; $\hat{\mathbf{F}}^{(j)} \in \mathbb{R}^{m\times n}$ and $\mathbf{F}^{(j)} \in \mathbb{R}^{n\times n}$ represent the forward transition rates from states in $\mathcal{S}^{(0)}$ to states in $\mathcal{S}^{(j)}$ and from states in $\mathcal{S}^{(k)}$ to states in $\mathcal{S}^{(k+j)}$ for $j \geq 1$ and $k \geq 1$, respectively; $\hat{\mathbf{B}} \in \mathbb{R}^{n\times m}$ and $\mathbf{B} \in \mathbb{R}^{n\times n}$ represent the backward transition rates from states in $\mathcal{S}^{(1)}$ to states in $\mathcal{S}^{(0)}$ and from states in $\mathcal{S}^{(j)}$ to states in $\mathcal{S}^{(j-1)}$ for $j \geq 2$, respectively; and $\mathbf{0}$ is a zero matrix of the appropriate dimension.

For $\mathbf{Q}$ to be an infinitesimal generator, the infinite sets of matrices $\{\hat{\mathbf{F}}^{(j)} : j \geq 1\}$ and $\{\mathbf{F}^{j} : j \geq 1\}$ must be summable. In practice, we must also be able to describe them using a finite representation, so we assume that they obey the following geometric expression:

$$\forall j \geq 1, \quad \hat{\mathbf{F}}^{(j)} = \hat{\mathbf{A}}^{j-1} \cdot \hat{\mathbf{F}} \qquad \text{and} \qquad \mathbf{F}^{(j)} = \mathbf{A}^{j-1} \cdot \mathbf{F},$$

where $\hat{\mathbf{A}}$ and $\mathbf{A}$ are nonnegative diagonal matrices with elements strictly less than one:

$$\hat{\mathbf{A}} = \begin{bmatrix} \hat{\alpha}_1 & 0 & 0 & 0 \\ 0 & \hat{\alpha}_2 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \hat{\alpha}_m \end{bmatrix} \quad \text{and} \quad \mathbf{A} = \begin{bmatrix} \alpha_1 & 0 & 0 & 0 \\ 0 & \alpha_2 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_n \end{bmatrix}.$$

This ensures that the infinite sums $\sum_{j=0}^{\infty} \hat{\mathbf{A}}^j = (\mathbf{I} - \hat{\mathbf{A}})^{-1}$ and $\sum_{j=0}^{\infty} \mathbf{A}^j = (\mathbf{I} - \mathbf{A})^{-1}$ exist. However, our methodology is not bound to the special diagonal structure of $\hat{\mathbf{A}}$ and $\mathbf{A}$ but rather to the requirement that these sums exist and are efficiently computable.

If we also partition the stationary probability vector satisfying $\boldsymbol{\pi} \cdot \mathbf{Q} = \mathbf{0}$ as $\boldsymbol{\pi} = \left[\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(2)}, \ldots\right]$ with $\boldsymbol{\pi}^{(0)} \in \mathbb{R}^m$ and $\boldsymbol{\pi}^{(j)} \in \mathbb{R}^n$ for $j \geq 1$, we can then write $\boldsymbol{\pi} \cdot \mathbf{Q} = \mathbf{0}$ as:

$$\begin{cases} \boldsymbol{\pi}^{(0)} \cdot \mathbf{L}^{(0)} + \boldsymbol{\pi}^{(1)} \cdot \hat{\mathbf{B}} = \mathbf{0} \\ \boldsymbol{\pi}^{(0)} \cdot \hat{\mathbf{F}}^{(1)} + \boldsymbol{\pi}^{(1)} \cdot \mathbf{L}^{(1)} + \boldsymbol{\pi}^{(2)} \cdot \mathbf{B} = \mathbf{0} \\ \boldsymbol{\pi}^{(0)} \cdot \hat{\mathbf{F}}^{(2)} + \boldsymbol{\pi}^{(1)} \cdot \mathbf{F}^{(1)} + \boldsymbol{\pi}^{(2)} \cdot \mathbf{L} + \boldsymbol{\pi}^{(3)} \cdot \mathbf{B} = \mathbf{0} \\ \boldsymbol{\pi}^{(0)} \cdot \hat{\mathbf{F}}^{(3)} + \boldsymbol{\pi}^{(1)} \cdot \mathbf{F}^{(2)} + \boldsymbol{\pi}^{(2)} \cdot \mathbf{F}^{(1)} + \boldsymbol{\pi}^{(3)} \cdot \mathbf{L} + \boldsymbol{\pi}^{(4)} \cdot \mathbf{B} = \mathbf{0} \\ \cdots \end{cases} \quad (3)$$

3.1 Conditions for stability

We briefly review the conditions that enable us to assert that the CTMC described by the infinitesimal generator $\mathbf{Q}$ in (2) is stable, that is, admits a probability vector satisfying $\boldsymbol{\pi} \cdot \mathbf{Q} = \mathbf{0}$ and $\boldsymbol{\pi} \cdot \mathbf{1}^T = 1$.

First, observe that the matrix $\tilde{\mathbf{Q}} = \mathbf{B} + \mathbf{L} + \sum_{j=1}^{\infty} \mathbf{F}^{(j)}$ is an infinitesimal generator, since it has zero row sums and non-negative off-diagonal entries. If $\tilde{\mathbf{Q}}$ is irreducible, any state $s_i^{(j)}$ in the original process can reach any state $s_{i'}^{(j')}$, for $2 \leq j \leq j'$ and $1 \leq i, i' \leq n$, without having to go through states in $\mathcal{S}^{(l)}$, $l < j$. In this case, for "large values of j", the conditional probability of being in $s_i^{(j)}$ given that we are in $\mathcal{S}^{(j)}$ tends to $\tilde{\boldsymbol{\pi}}_i$, where $\tilde{\boldsymbol{\pi}}$ is the unique solution of $\tilde{\boldsymbol{\pi}} \cdot \tilde{\mathbf{Q}} = \mathbf{0}$, subject to $\tilde{\boldsymbol{\pi}} \cdot \mathbf{1}^T = 1$, that is, the stationary solution of the ergodic CTMC having $\tilde{\mathbf{Q}}$ as the infinitesimal generator.

Then, the M/G/1-type process is stable as long as, for large values of j, the forward drift from $\mathcal{S}^{(j)}$ is less than the backward drift from it:

$$\tilde{\boldsymbol{\pi}} \cdot \left(\sum_{l=1}^{\infty} l \cdot \mathbf{F}^{(l)} \right) \cdot \mathbf{1}^T < \tilde{\boldsymbol{\pi}} \cdot \mathbf{B} \cdot \mathbf{1}^T.$$

This condition can be verified numerically, and it is easy to see that it is equivalent to the one given by Neuts in [9], $\tilde{\boldsymbol{\pi}} \cdot \boldsymbol{\beta} < 1$, where, in our terminology, the column vector $\boldsymbol{\beta}$ is given by $\boldsymbol{\beta} = \left(\mathbf{L} + \sum_{l=1}^{\infty} l \cdot \mathbf{F}^{(l)} \right) \cdot \mathbf{1}^T$. As in the scalar case, the condition where $\tilde{\boldsymbol{\pi}} \cdot \boldsymbol{\beta}$ is exactly equal to 1 results in a null-recurrent CTMC. If $\tilde{\mathbf{Q}}$ is instead reducible, the same condition for stability must be applied to each subset of $\{1, \ldots, n\}$ corresponding to a recurrent class in the CTMC described by $\tilde{\mathbf{Q}}$.

3.2 Main theorem

As done in [2], we must require that $\mathbf{B}$ contains nonzero entries only in its last column. Then, we derive $m+2n$ equations in $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and the new vector of n unknowns $\boldsymbol{\pi}^{(*)}=\sum_{i=2}^{\infty}\boldsymbol{\pi}^{(i)}$. Under these assumptions we can formulate the following theorem.

Theorem. Given an ergodic CTMC with infinitesimal generator $\mathbf{Q}$ having the structure shown in (2) such that the first $n-1$ columns of $\mathbf{B}$ are null, $\mathbf{B}_{1:n,1:n-1}=\mathbf{0}$, and with stationary probability vector $\boldsymbol{\pi}=\left[\boldsymbol{\pi}^{(0)},\boldsymbol{\pi}^{(1)},\boldsymbol{\pi}^{(2)},\ldots\right]$, the system of linear equations

$$\mathbf{x}\cdot\mathbf{M}=[1,\mathbf{0}] \tag{4}$$

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

$$\mathbf{M}=\left[\begin{array}{c|c|c|c|c}\mathbf{1}^T & \mathbf{L}^{(0)} & \hat{\mathbf{F}}^{(1)}_{1:m,1:n-1} & (\hat{\mathbf{A}}\cdot(\mathbf{I}-\hat{\mathbf{A}})^{-1}\cdot\hat{\mathbf{F}})_{1:n,1:n-1} & \hat{\mathbf{A}}\cdot(\mathbf{I}-\hat{\mathbf{A}})^{-2}\cdot\hat{\mathbf{F}}\cdot\mathbf{1}^T \\ \mathbf{1}^T & \hat{\mathbf{B}} & \mathbf{L}^{(1)}_{1:n,1:n-1} & ((\mathbf{I}-\mathbf{A})^{-1}\cdot\mathbf{F})_{1:n,1:n-1} & (\mathbf{I}-\mathbf{A})^{-2}\cdot\mathbf{F}\cdot 1^T \\ \mathbf{1}^T & \mathbf{0} & \mathbf{0} & ((\mathbf{I}-\mathbf{A})^{-1}\cdot\mathbf{F}+\mathbf{L})_{1:n,1:n-1} & ((\mathbf{I}-\mathbf{A})^{-2}\mathbf{F}-\mathbf{B})\cdot\mathbf{1}^T\end{array}\right] \tag{5}$$

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 $\left[\boldsymbol{\pi}^{(0)},\boldsymbol{\pi}^{(1)},\boldsymbol{\pi}^{(*)}\right]$ is a solution of (4) by verifying that it satisfies five matrix equations corresponding to the the five sets of columns we used to define $\mathbf{M}$.

(i) The first equation is the normalization constraint:

$$\boldsymbol{\pi}^{(0)}\cdot\mathbf{1}^T+\boldsymbol{\pi}^{(1)}\cdot\mathbf{1}^T+\boldsymbol{\pi}^{(*)}\cdot\mathbf{1}^T=1. \tag{6}$$

(ii) The second set of m equations is the first line in (3):

$$\boldsymbol{\pi}^{(0)}\cdot\mathbf{L}^{(0)}+\boldsymbol{\pi}^{(1)}\cdot\hat{\mathbf{B}}=\mathbf{0}. \tag{7}$$

(iii) The third set of $n-1$ equations is from the second line in (3), which defines n equations, fortunately only the last one actually containing a contribution from $\boldsymbol{\pi}^{(2)}$, due to the structure of $\mathbf{B}$. This is of fundamental importance, since $\boldsymbol{\pi}^{(2)}$ is not one of our variables. Hence, we consider only the first $n-1$ equations and write:

$$\boldsymbol{\pi}^{(0)}\cdot\hat{\mathbf{F}}^{(1)}_{1:m,1:n-1}+\boldsymbol{\pi}^{(1)}\cdot\mathbf{L}^{(1)}_{1:m,1:n-1}=\mathbf{0} \tag{8}$$

(iv) Another set of $n-1$ equations is obtained as follows. First, the sum of the remaining lines in (3) gives

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

which, since $\sum_{j=2}^{\infty}\hat{\mathbf{F}}^{(j)}=\hat{\mathbf{A}}\cdot(\mathbf{I}-\hat{\mathbf{A}})^{-1}\cdot\hat{\mathbf{F}}$ and $\sum_{j=1}^{\infty}\mathbf{F}^{(j)}=(\mathbf{I}-\mathbf{A})^{-1}\cdot\mathbf{F}$, can be written as

$$\boldsymbol{\pi}^{(0)}\hat{\mathbf{A}}\cdot(\mathbf{I}-\hat{\mathbf{A}})^{-1}\cdot\hat{\mathbf{F}}+\boldsymbol{\pi}^{(1)}(\mathbf{I}-\mathbf{A})^{-1}\cdot\mathbf{F}+\boldsymbol{\pi}^{(*)}\left((\mathbf{I}-\mathbf{A})^{-1}\cdot\mathbf{F}+\mathbf{L}\right)+(\boldsymbol{\pi}^{(*)}-\boldsymbol{\pi}^{(2)})\cdot\mathbf{B}=\mathbf{0}.$$

Again, only the last equation contains a contribution from $\boldsymbol{\pi}^{(2)}$, thus we consider only the first $n-1$ equations:

$$\boldsymbol{\pi}^{(0)}\left(\hat{\mathbf{A}}\cdot(\mathbf{I}-\hat{\mathbf{A}})^{-1}\cdot\hat{\mathbf{F}}\right)_{1:n,1:n-1}+\boldsymbol{\pi}^{(1)}\left((\mathbf{I}-\mathbf{A})^{-1}\cdot\mathbf{F}\right)_{1:n,1:n-1}+\boldsymbol{\pi}^{(*)}\left((\mathbf{I}-\mathbf{A})^{-1}\mathbf{F}\cdot+\mathbf{L}\right)_{1:n,1:n-1}=\mathbf{0} \tag{9}$$

(v) For the last equation, consider the flow balance equations between $\cup_{l=0}^{j-1}\mathcal{S}^{(l)}$ and $\cup_{l=j}^{\infty}\mathcal{S}^{(l)}$, for $j \geq 2$,

$$\begin{cases} \boldsymbol{\pi}^{(0)} \cdot \sum_{i=2}^{\infty} \hat{\mathbf{F}}^{(i)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(1)} \cdot \sum_{i=1}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T + & = \boldsymbol{\pi}^{(2)} \cdot \mathbf{B} \cdot \mathbf{1}^T \\ \boldsymbol{\pi}^{(0)} \cdot \sum_{i=3}^{\infty} \hat{\mathbf{F}}^{(i)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(1)} \cdot \sum_{i=2}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(2)} \cdot \sum_{i=1}^{\infty} \cdot \mathbf{F}^{(i)} \cdot \mathbf{1}^T & = \boldsymbol{\pi}^{(3)} \cdot \mathbf{B} \cdot \mathbf{1}^T \\ \cdots & \\ \boldsymbol{\pi}^{(0)} \cdot \sum_{i=k+1}^{\infty} \hat{\mathbf{F}}^{(i)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(1)} \cdot \sum_{i=k}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(2)} \cdot \sum_{i=k-1}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T + \cdots + \boldsymbol{\pi}^{(k)} \cdot \sum_{i=1}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T & = \boldsymbol{\pi}^{(k+1)} \cdot \mathbf{B} \cdot \mathbf{1}^T \\ \cdots & \end{cases} \tag{10}$$

and sum these equations by parts:

$$\boldsymbol{\pi}^{(0)} \cdot \sum_{k=2}^{\infty} \sum_{i=k}^{\infty} \hat{\mathbf{F}}^{(i)} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(1)} \cdot \sum_{k=1}^{\infty} \sum_{i=k}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T + \sum_{j=2}^{\infty} \boldsymbol{\pi}^{(j)} \cdot \sum_{k=1}^{\infty} \sum_{i=k}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T = \sum_{k=2}^{\infty} \boldsymbol{\pi}^{(k)} \cdot \mathbf{B} \cdot \mathbf{1}^T,$$

since $\sum_{k=2}^{\infty} \sum_{i=k}^{\infty} \hat{\mathbf{F}}^{(i)} = \hat{\mathbf{A}} \cdot (\mathbf{I} - \hat{\mathbf{A}})^{-2} \cdot \hat{\mathbf{F}}$ and $\sum_{k=1}^{\infty} \sum_{i=k}^{\infty} \mathbf{F}^{(i)} = (\mathbf{I} - \mathbf{A})^{-2} \cdot \mathbf{F}$, we finally obtain

$$\boldsymbol{\pi}^{(0)} \cdot \hat{\mathbf{A}} \cdot (\mathbf{I} - \hat{\mathbf{A}})^{-2} \cdot \hat{\mathbf{F}} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(1)} \cdot (\mathbf{I} - \mathbf{A})^{-2} \cdot \mathbf{F} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(*)} \cdot ((\mathbf{I} - \mathbf{A})^{-2} \cdot \mathbf{F} - \mathbf{B}) \cdot \mathbf{1}^T = 0 \tag{11}$$

The vector $[\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(*)}]$ satisfies (6), (7), (8), (9), and (11), hence it is a solution of (4). Now we have to show that this solution is unique. For this, it is enough to prove that the rank of $\mathbf{M}$ is $m + 2n$, by showing that its $m + 2n$ rows are linearly independent.

Since $\mathbf{Q}$ is ergodic, we know that the vector $\mathbf{1}^T$ and the set of vectors corresponding to all the columns of $\mathbf{Q}$ except one (any one of them) are linearly independent. In particular, we choose to remove from $\mathbf{Q}$ the n^{th} column of the second block of columns in (2), corresponding to the transitions into state s_n^1. The result is then the countably infinite set of linearly independent column vectors of $\mathbb{R}^{\mathbb{N}}$ $\{\mathbf{v}^{[1]}, \mathbf{v}^{[2]}, \ldots\}$ shown in Fig. 3. We then define n new vectors of $\mathbb{R}^{\mathbb{N}}$, $\{\mathbf{z}^{[1]}, \ldots, \mathbf{z}^{[n]}\}$ as follows (see Fig. 3):

- For $i = 1, \ldots, n-1$, let $\mathbf{z}^{[i]} = \sum_{j=1}^{\infty} \mathbf{v}^{[m+jn+i]}$, that is we sum the i^{th} column for each block of columns, starting from the block corresponding to transitions into level $\mathcal{S}^{(2)}$. $\mathbf{B}$ doesn't appear in the expression of these vectors because $\mathbf{B}_{1:n,1:n-1} = \mathbf{0}$.

- Let $\mathbf{z}^{[n]} = \sum_{j=1}^{\infty} \sum_{i=1}^{n} \sum_{k=j}^{\infty} \mathbf{v}^{[m+kn+i]}$. To justify the expression for $\mathbf{z}^{[n]}$, it is convenient to consider its derivation in steps: $\mathbf{z}^{[n]} = \sum_{j=1}^{\infty} \mathbf{y}^{[j]}$, where $\mathbf{y}^{[j]} = \sum_{i=1}^{n} \mathbf{x}^{[(j-1)n+i]}$, and $\mathbf{x}^{[(j-1)n+i]} = \sum_{k=j}^{\infty} \mathbf{v}^{[m+kn+i]}$. Note that $\mathbf{Q}$ being an infinitesimal generator implies that $(\mathbf{B} + \mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}) \cdot \mathbf{1}^T = \mathbf{0}$, which explains the $\mathbf{0}$ components in the vectors $\mathbf{y}^{[j]}$, or, equivalently, that $(\mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}) \cdot \mathbf{1}^T = -\mathbf{B} \cdot \mathbf{1}^T$, which explains the presence of $\mathbf{B}$ in $\mathbf{z}^{[n]}$.

The $m + 2n$ vectors $\{\mathbf{v}^{[1]}, \ldots, \mathbf{v}^{[m+n]}, \mathbf{z}^{[1]}, \ldots, \mathbf{z}^{[n]}\}$ are linearly independent because:

- The original set $\{\mathbf{v}^{[1]}, \mathbf{v}^{[2]}, \ldots\}$ is linearly independent.

- The vectors $\{\mathbf{z}^{[1]}, \ldots, \mathbf{z}^{[n]}\}$ are obtained as linear combinations of different subsets of vectors from $\{\mathbf{v}^{[m+n+1]}, \mathbf{v}^{[m+n+2]}, \ldots\}$.

$\mathbf{v}^{[1]}$	$\mathbf{v}^{[2]}$ through $\mathbf{v}^{[m+1]}$	$\mathbf{v}^{[m+2]}$ through $\mathbf{v}^{[m+n]}$	$\mathbf{v}^{[m+n+1]}$ through $\mathbf{v}^{[m+2n]}$	$\mathbf{v}^{[m+2n+1]}$ through $\mathbf{v}^{[m+3n]}$	$\mathbf{v}^{[m+3n+1]}$ through $\mathbf{v}^{[m+4n]}$	$\cdots$
$\mathbf{1}^T$	$\mathbf{L}^{(0)}$	$\hat{\mathbf{F}}^{(1)}_{1:m,1:n-1}$	$\hat{\mathbf{F}}^{(2)}$	$\hat{\mathbf{F}}^{(3)}$	$\hat{\mathbf{F}}^{(4)}$	$\cdots$
$\mathbf{1}^T$	$\hat{\mathbf{B}}$	$\mathbf{L}^{(1)}_{1:n,1:n-1}$	$\mathbf{F}^{(1)}$	$\mathbf{F}^{(2)}$	$\mathbf{F}^{(3)}$	$\cdots$
$\mathbf{1}^T$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{L}$	$\mathbf{F}^{(1)}$	$\mathbf{F}^{(2)}$	$\cdots$
$\mathbf{1}^T$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{B}$	$\mathbf{L}$	$\mathbf{F}^{(1)}$	$\cdots$
$\mathbf{1}^T$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{B}$	$\mathbf{L}$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$

$\mathbf{z}^{[1]}$ through $\mathbf{z}^{[n-1]}$
$(\hat{\mathbf{A}} \cdot (\mathbf{I} - \hat{\mathbf{A}})^{-1} \cdot \hat{\mathbf{F}})_{1:m,1:n-1}$
$((\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F})_{1:n,1:n-1}$
$((\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F} + \mathbf{L})_{1:n,1:n-1}$
$((\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F} + \mathbf{L})_{1:n,1:n-1}$
$((\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F} + \mathbf{L})_{1:n,1:n-1}$
$\vdots$

$\mathbf{x}^{[1]}$ through $\mathbf{x}^{[n]}$	$\mathbf{x}^{[n+1]}$ through $\mathbf{x}^{[2n]}$	$\cdots$
$\hat{\mathbf{A}} \cdot (\mathbf{I} - \hat{\mathbf{A}})^{-1} \cdot \hat{\mathbf{F}}$	$\hat{\mathbf{A}}^2 \cdot (\mathbf{I} - \hat{\mathbf{A}})^{-1} \cdot \hat{\mathbf{F}}$	$\cdots$
$(\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}$	$\mathbf{A} \cdot (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}$	$\cdots$
$\mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}$	$(\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}$	$\cdots$
$\mathbf{B} + \mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}$	$\mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}$	$\cdots$
$\mathbf{B} + \mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}$	$\mathbf{B} + \mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}$	$\cdots$
$\vdots$	$\vdots$	$\cdots$

$\mathbf{y}^{[1]}$	$\mathbf{y}^{[2]}$	$\cdots$
$\hat{\mathbf{A}} \cdot (\mathbf{I} - \hat{\mathbf{A}})^{-1} \cdot \hat{\mathbf{F}} \cdot \mathbf{1}^T$	$\hat{\mathbf{A}}^2 \cdot (\mathbf{I} - \hat{\mathbf{A}})^{-1} \cdot \hat{\mathbf{F}} \cdot \mathbf{1}^T$	$\cdots$
$(\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F} \cdot \mathbf{1}^T$	$\mathbf{A} \cdot (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F} \cdot \mathbf{1}^T$	$\cdots$
$(\mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}) \cdot \mathbf{1}^T$	$(\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F} \cdot \mathbf{1}^T$	$\cdots$
$\mathbf{0}$	$(\mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}) \cdot \mathbf{1}^T$	$\cdots$
$\mathbf{0}$	$\mathbf{0}$	$\cdots$
$\vdots$	$\vdots$	$\cdots$

$\mathbf{z}^{[n]}$
$\hat{\mathbf{A}} \cdot (\mathbf{I} - \hat{\mathbf{A}})^{-2} \cdot \hat{\mathbf{F}} \cdot \mathbf{1}^T$
$(\mathbf{I} - \mathbf{A})^{-2} \cdot \mathbf{F} \cdot \mathbf{1}^T$
$((\mathbf{I} - \mathbf{A})^{-2} \cdot \mathbf{F} - \mathbf{B}) \cdot \mathbf{1}^T$
$((\mathbf{I} - \mathbf{A})^{-2} \cdot \mathbf{F} - \mathbf{B}) \cdot \mathbf{1}^T$
$((\mathbf{I} - \mathbf{A})^{-2} \cdot \mathbf{F} - \mathbf{B}) \cdot \mathbf{1}^T$
$\vdots$

Figure 3: The column vectors used to prove linear independence.

- Disjoint subsets of vectors are used to build $\{\mathbf{z}^{[1]}, \ldots, \mathbf{z}^{[n-1]}\}$.
- $\mathbf{z}^{[n]}$ is built using vectors already used for $\{\mathbf{z}^{[1]}, \ldots, \mathbf{z}^{[n-1]}\}$, but also vectors of the form $\mathbf{v}^{[m+jn]}$, which are not used to build any of the vectors in $\{\mathbf{z}^{[1]}, \ldots, \mathbf{z}^{[n-1]}\}$.

Hence, the matrix having as column the vectors $\{\mathbf{v}^{[1]}, \ldots, \mathbf{v}^{[m+n]}, \mathbf{z}^{[1]}, \ldots, \mathbf{z}^{[n]}\}$ has rank $m+2n$, which implies that we must be able to find $m+2n$ linearly independent rows in it. Since the row $m+jn+i$ is identical to the row $m+n+i$ for $j > 1$ and $i = 1, \ldots, n$, the first $m+2n$ rows must be linearly independent. These are also the rows of our matrix $\mathbf{M}$, and so the proof is complete. $\square$

4 Computing the measures of interest

We now consider the problem of obtaining stationary measures of interest once $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and $\boldsymbol{\pi}^{(*)}$ have been computed. We consider measures that can be expressed as the expected reward rate

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

where $\boldsymbol{\rho}_i^{(j)}$ is the *reward rate* of state $s_i^{(j)}$. For example, if we want to compute the expected queue length in steady state for a model where $\mathcal{S}^{(j)}$ contains the system states with j customers in the queue, we let $\boldsymbol{\rho}_i^{(j)} = j$, while, 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 $\boldsymbol{\pi}^{(*)}$, we rewrite r as

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

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 having the values of $\boldsymbol{\pi}^{(j)}$ for $j \geq 2$ explicitly available. We do so for the case where 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 $\mathbf{a}_i^{[0]}, \mathbf{a}_i^{[1]}, \ldots, \mathbf{a}_i^{[k]}$:

$$\forall j \geq 2,\ \forall i \in \{1, 2, \ldots, n\}, \qquad \boldsymbol{\rho}_i^{(j)} = \mathbf{a}_i^{[0]} + \mathbf{a}_i^{[1]} j + \cdots + \mathbf{a}_i^{[k]} j^k. \tag{12}$$

In this case, then,

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

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 (3) from the second line on by the appropriate factor j^k results in

$$\left\{ \begin{array}{lllllllllll}
2^k\boldsymbol{\pi}^{(0)}\cdot\hat{\mathbf{F}}^{(1)} & + & 2^k\boldsymbol{\pi}^{(1)}\cdot\mathbf{L}^{(1)} & + & 2^k\boldsymbol{\pi}^{(2)}\cdot\mathbf{B} & & & & & = & \mathbf{0} \\
3^k\boldsymbol{\pi}^{(0)}\cdot\hat{\mathbf{F}}^{(2)} & + & 3^k\boldsymbol{\pi}^{(1)}\cdot\mathbf{F}^{(1)} & + & 3^k\boldsymbol{\pi}^{(2)}\cdot\mathbf{L} & + & 3^k\boldsymbol{\pi}^{(3)}\cdot\mathbf{B} & & & = & \mathbf{0} \\
4^k\boldsymbol{\pi}^{(0)}\cdot\hat{\mathbf{F}}^{(3)} & + & 4^k\boldsymbol{\pi}^{(1)}\cdot\mathbf{F}^{(2)} & + & 4^k\boldsymbol{\pi}^{(2)}\cdot\mathbf{F}^{(1)} & + & 4^k\boldsymbol{\pi}^{(3)}\cdot\mathbf{L} & + & 4^k\boldsymbol{\pi}^{(4)}\cdot\mathbf{B} & = & \mathbf{0} \\
 & & \cdots & & & & & & & &
\end{array} \right. .$$

Summing these equations by parts we obtain

$$\underbrace{\boldsymbol{\pi}^{(0)} \sum_{j=0}^{\infty} (j+2)^k \cdot \hat{\mathbf{A}}^j \cdot \hat{\mathbf{F}}}_{\stackrel{\text{def}}{=}\ \mathbf{f}^{[k,0]}} + \underbrace{\boldsymbol{\pi}^{(1)} \left(2^k \mathbf{L}^{(1)} + \sum_{j=0}^{\infty} (j+3)^k \cdot \mathbf{A}^j \cdot \mathbf{F}\right)}_{\stackrel{\text{def}}{=}\ \mathbf{f}^{[k,1]}} +$$

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

which can then be rewritten as

$$\sum_{i=2}^{\infty} \boldsymbol{\pi}^{(i)} \cdot \left[\left(\sum_{j=0}^{\infty} \sum_{l=0}^{k} \binom{k}{l} (j+2)^l i^{k-l} \cdot \mathbf{A}^j \cdot \mathbf{F} \right) + \left(\sum_{l=0}^{k} \binom{k}{l} 1^l i^{k-l} \cdot \mathbf{L} \right) \right] + \mathbf{r}^{[k]} \cdot \mathbf{B} = -\mathbf{f}^{[k,0]} - \mathbf{f}^{[k,1]}.$$

Exchanging the order of summations we obtain

$$\sum_{l=0}^{k} \binom{k}{l} \underbrace{\sum_{i=2}^{\infty} \boldsymbol{\pi}^{(i)} \cdot i^{k-l}}_{=\ \mathbf{r}^{[k-l]}} \cdot \left(\mathbf{L} + \sum_{j=0}^{\infty} (j+2)^l \cdot \mathbf{A}^j \cdot \mathbf{F} \right) + \mathbf{r}^{[k]} \cdot \mathbf{B} = -\mathbf{f}^{[k,0]} - \mathbf{f}^{[k,1]}.$$

Finally, separating the case $l = 0$ from the rest in the outermost summation we obtain

$$\mathbf{r}^{[k]} \cdot \left(\mathbf{L} + \sum_{j=0}^{\infty} \mathbf{A}^j \cdot \mathbf{F} \right) + \mathbf{r}^{[k]} \cdot \mathbf{B} = -\mathbf{f}^{[k,0]} - \mathbf{f}^{[k,1]} - \sum_{l=1}^{k} \binom{k}{l} \mathbf{r}^{[k-l]} \cdot \left(\mathbf{L} + \sum_{j=0}^{\infty} (j+2)^l \cdot \mathbf{A}^j \cdot \mathbf{F} \right),$$

which is a linear system of the form $\mathbf{r}^{[k]} \cdot (\mathbf{B} + \mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}) = \mathbf{b}^{[k]}$, 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 $(\mathbf{B} + \mathbf{L} + (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F})$ is $n-1$, so the above system is under-determined. We then remove the equation corresponding to the last column, resulting in

$$\mathbf{r}^{[k]} \cdot ((\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F} + \mathbf{L})_{1:n,1:n-1} = \mathbf{b}^{[k]}_{1:n-1}, \tag{13}$$

and obtain one additional equation for $\mathbf{r}^{[k]}$ from the equations in (10), again after multiplying them by the appropriate factor j^k,

$$\begin{cases} 2^k \boldsymbol{\pi}^{(0)} \cdot \sum_{i=2}^{\infty} \hat{\mathbf{F}}^{(i)} \cdot \mathbf{1}^T + 2^k \boldsymbol{\pi}^{(1)} \cdot \sum_{i=1}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T & = 2^k \boldsymbol{\pi}^{(2)} \cdot \mathbf{B} \cdot \mathbf{1}^T \\ 3^k \boldsymbol{\pi}^{(0)} \cdot \sum_{i=3}^{\infty} \hat{\mathbf{F}}^{(i)} \cdot \mathbf{1}^T + 3^k \boldsymbol{\pi}^{(1)} \cdot \sum_{i=2}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T + 3^k \boldsymbol{\pi}^{(2)} \cdot \sum_{i=1}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T & = 3^k \boldsymbol{\pi}^{(3)} \cdot \mathbf{B} \cdot \mathbf{1}^T \\ 4^k \boldsymbol{\pi}^{(0)} \cdot \sum_{i=4}^{\infty} \hat{\mathbf{F}}^{(i)} \cdot \mathbf{1}^T + 4^k \boldsymbol{\pi}^{(1)} \cdot \sum_{i=3}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T + 4^k \boldsymbol{\pi}^{(2)} \cdot \sum_{i=2}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T + 4^k \boldsymbol{\pi}^{(3)} \cdot \sum_{i=1}^{\infty} \mathbf{F}^{(i)} \cdot \mathbf{1}^T & = 4^k \boldsymbol{\pi}^{(4)} \cdot \mathbf{B} \cdot \mathbf{1}^T \\ \cdots & \end{cases}.$$

Noting that, for $j \geq 1$, $\sum_{i=j}^{\infty} \mathbf{F}^{(i)} = \mathbf{A}^{j-1} \cdot (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F}$, summing these equations by parts gives:

$$\boldsymbol{\pi}^{(0)} \sum_{j=0}^{\infty} (j+2)^k \cdot \hat{\mathbf{A}}^{j+1} \cdot (\mathbf{I} - \hat{\mathbf{A}})^{-1} \cdot \hat{\mathbf{F}} \cdot \mathbf{1}^T + \boldsymbol{\pi}^{(1)} \sum_{j=0}^{\infty} (j+2)^k \cdot \mathbf{A}^j \cdot (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F} \cdot \mathbf{1}^T +$$

$$\sum_{l=2}^{\infty} \boldsymbol{\pi}^{(l)} \cdot \sum_{j=0}^{\infty} (j+l+1)^k \cdot \mathbf{A}^j \cdot (\mathbf{I} - \mathbf{A})^{-1} \cdot \mathbf{F} \cdot \mathbf{1}^T = \sum_{j=2}^{\infty} j^k \cdot \boldsymbol{\pi}^{(j)} \cdot \mathbf{B} \cdot \mathbf{1}^T,$$

which, with steps analogous to those just performed to obtain (13), can be written as

$$\mathbf{r}^{[k]} \cdot ((\mathbf{I} - \mathbf{A})^{-2} \cdot \mathbf{F} - \mathbf{B}) \cdot \mathbf{1}^T = c^{[k]} \tag{14}$$

where $c^{[k]}$ is, again, an expression containing $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and the vectors $\mathbf{r}^{[0]}$ through $\mathbf{r}^{[k-1]}$.

Note that the $n \times n$ matrix $[((\mathbf{I}-\mathbf{A})^{-1} \cdot \mathbf{F} + \mathbf{L})_{1:n,1:n-1} | ((\mathbf{I}-\mathbf{A})^{-2} \cdot \mathbf{F} - \mathbf{B}) \cdot \mathbf{1}^T]$ has full rank. This follows from the fact that we already proved that the matrix $\mathbf{M}$ of our main theorem, (5), has full rank, hence its last n rows are linearly independent. But the first $m+n$ entries in each of those n rows are identical, hence their last n columns must be linearly independent; that is exactly the matrix we are considering here. Hence, we can compute $\mathbf{r}^{[k]}$ using (13) and (14), i.e., solving a linear system in n unknowns (of course, we must do so first for $l = 1, \ldots, k-1$).

It should be noted that the expressions for $\mathbf{b}^{[k]}$ and $c^{[k]}$ contain sums of the form

$$\sum_{j=0}^{\infty} (j+d)^l \cdot \mathbf{A}^j = \left(\sum_{j=1}^{\infty} j^l \cdot \mathbf{A}^j - \sum_{j=1}^{d-1} j^l \cdot \mathbf{A}^j \right) \cdot \mathbf{A}^{-d}, \qquad d = 2,3 \qquad l = 1, \ldots, k.$$

Thus, we need to compute infinite sum of the form $\sum_{j=1}^{\infty} j^l \cdot \mathbf{A}^j$, which obviously converge since the spectral radius of $\mathbf{A}$ is strictly less than one. It can be shown that

$$\sum_{j=1}^{\infty} j^l \cdot \mathbf{A}^j = \left(\sum_{i=1}^{l} A(l,i) \cdot \mathbf{A}^i \right) \cdot (\mathbf{I}-\mathbf{A})^{-(l+1)}$$

where $A(l,i)$ are the Eulerian numbers [10], which can be computed using the recursion

$$A(l,i) = i \cdot A(l-1,i) + (l-i+1) \cdot A(l-1,i-1),$$

with $A(1,1) = 1$ and $A(l,i) = 0$ for $i < 1$ or $i > l$.

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,

$$\mathbf{b}^{[1]} = -\boldsymbol{\pi}^{(0)} \cdot (2(\mathbf{I}-\hat{\mathbf{A}})^{-1} + \hat{\mathbf{A}} \cdot (\mathbf{I}-\hat{\mathbf{A}})^{-2}) \cdot \hat{\mathbf{F}} - \boldsymbol{\pi}^{(1)} \cdot \left(2^k \mathbf{L}^{(1)} + (3(\mathbf{I}-\mathbf{A})^{-1} + \mathbf{A} \cdot (\mathbf{I}-\mathbf{A})^{-2}) \cdot \mathbf{F} \right)$$
$$-\mathbf{r}^{[0]} \cdot \left(\mathbf{A} \cdot (\mathbf{I}-\mathbf{A})^{-2} \cdot \mathbf{F} + 2(\mathbf{I}-\mathbf{A})^{-1} \cdot \mathbf{F} + \mathbf{L} \right)$$

and

$$c^{[1]} = -\boldsymbol{\pi}^{(0)} \cdot ((\mathbf{I}-\hat{\mathbf{A}})^{-3} + (\mathbf{I}-\hat{\mathbf{A}})^{-2}) \cdot \hat{\mathbf{A}} \cdot \hat{\mathbf{F}} \cdot \mathbf{1}^T - \boldsymbol{\pi}^{(1)} \cdot (\mathbf{A} \cdot (\mathbf{I}-\mathbf{A})^{-3} + 2(\mathbf{I}-\mathbf{A})^{-2}) \cdot \mathbf{F} \cdot \mathbf{1}^T$$
$$-\mathbf{r}^{[0]} \cdot (\mathbf{A} \cdot (\mathbf{I}-\mathbf{A})^{-3} + (\mathbf{I}-\mathbf{A})^{-2}) \cdot \mathbf{F} \cdot \mathbf{1}^T.$$

We conclude by observing that, when the sequences $\{\hat{\mathbf{F}}^{(j)} : j \geq 1\}$ and $\{\mathbf{F}^{(j)} : j \geq 1\}$ do not have the geometric form we assume, the treatment in this section can be modified appropriately. However, some measures might be infinite. For example, if the sequences are summable but decrease only like $1/j^d$ for some $d > 1$, then the moments of order $d-1$ or higher for the queue length do not exist (are infinite).

5 The case of bounded bulk arrivals

If we restrict the process so that it is allowed to jump forward by at most p levels, its infinitesimal generator $\mathbf{Q}$ still has the structure of (2), except that $\hat{\mathbf{F}}^{(j)}$ and $\mathbf{F}^{(j)}$ are zero for $j > p$ (this occurs, for example, when modeling queues with bounded bulk arrivals). Since the number of matrices on any row is finite, there is no reason to require any particular relation between the matrices $\hat{\mathbf{F}}^{(j)}$ or $\mathbf{F}^{(j)}$. We can then formulate the following lemma.

Lemma. Given an ergodic CTMC with infinitesimal generator $\mathbf{Q}$, having the structure shown in (2) such that $\hat{\mathbf{F}}^{(j)} = \mathbf{0}$ and $\mathbf{F}^{(j)} = \mathbf{0}$ for $j > p$ and the first $n-1$ columns of $\mathbf{B}$ are null, $\mathbf{B}_{1:n,1:n-1} = \mathbf{0}$, and with stationary probability vector $\boldsymbol{\pi} = \left[\boldsymbol{\pi}^{(0)}, \boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(2)}, \ldots\right]$, the system of linear equations

$$\mathbf{x} \cdot \left[\begin{array}{c|c|c|c|c} \mathbf{1}^T & \mathbf{L}^{(0)} & \mathbf{F}_{1:m,1:n-1} & \left(\sum_{j=2}^{p} \hat{\mathbf{F}}^{(j)}\right)_{1:n,1:n-1} & \left(\sum_{j=2}^{p} (j-1) \cdot \hat{\mathbf{F}}^{(j)}\right) \cdot \mathbf{1}^T \\ \mathbf{1}^T & \hat{\mathbf{B}} & \mathbf{L}^{(1)}_{1:n,1:n-1} & \left(\sum_{j=1}^{p} \mathbf{F}^{(j)}\right)_{1:n,1:n-1} & \left(\sum_{j=1}^{p} j \cdot \mathbf{F}^{(j)}\right) \cdot \mathbf{1}^T \\ \mathbf{1}^T & \mathbf{0} & \mathbf{0} & \left(\sum_{j=1}^{p} \mathbf{F}^{(j)} + \mathbf{L}\right)_{1:n,1:n-1} & \left(\sum_{j=1}^{p} j \cdot \mathbf{F}^{(j)} - \mathbf{B}\right) \cdot \mathbf{1}^T \end{array}\right] = [1, \mathbf{0}]$$

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 of the proof are exactly the same as those of the theorem introduced in Sect. 3, hence they are omitted. □

The measure of interest r can also be derived as done in Sect. 4. Using the same definitions for r, $\boldsymbol{\rho}$, and $\mathbf{r}^{[l]}$, we need to show how to express the vectors $\mathbf{r}^{[l]}$, $l = 0, \ldots, k$. For brevity's sake, we omit the steps, and simply state that, again, $\mathbf{r}^{[k]}$ can be computed recursively by solving the equation

$$\mathbf{r}^{[k]} \cdot \left[\left(\sum_{j=1}^{p} \mathbf{F}^{(j)} + \mathbf{L}\right)_{1:n,1:n-1} \;\middle|\; \left(\sum_{j=1}^{p} j \cdot \mathbf{F}^{(j)} - \mathbf{B}\right) \cdot \mathbf{1}^T\right] = \mathbf{b}^{[k]},$$

where $\mathbf{b}^{[k]}$ is computed using $\boldsymbol{\pi}^{(0)}$, $\boldsymbol{\pi}^{(1)}$, and $\mathbf{r}^{[0]}$ through $\mathbf{r}^{[k-1]}$.

We stress that we consider the case of finite bulk arrivals explicitly because, while such type of process could be solved using the version of ETAQA presented in [2] or the matrix-geometric method [8], by merging every p levels into a larger single level, the results of this section show how to compute the solution without the corresponding overhead (a factor of p for both execution time and storage in ETAQA, a factor of p^3 for execution time and p^2 for storage in the matrix-geometric method).

6 ETAQA and alternative solution methods

In this section, we attempt to examine where the extended ETAQA approach we just introduced lies with respect to alternative solution approaches. We start by saying that M/G/1-type processes have generally received less attention than GI/M/1-type processes in the literature. This is no doubt due to their greater analytical complexity and to the lack of such a well-known and successful solution approach as the matrix-geometric technique for GI/M/1-type processes. The main reference we are aware of for the solution of M/G/1-type processes is the "other" book by Neuts, published in 1989, which presents a solution approach which we will call "Neuts's algorithm" [9, pages 158-167].

Neuts's algorithm is described in terms of a discrete-time Markov chain with a transition probability matrix $\mathbf{Q}$ having the same structure as in (2), with $\mathbf{L}^{(1)} = \mathbf{L}$ (we do not assume this condition in the continuous case because $\hat{\mathbf{B}} \neq \mathbf{B}$ implies that the diagonal of $\mathbf{L}^{(1)}$ might differ from that of $\mathbf{L}$). The algorithm is based on the matrix $\mathbf{G}$, whose entry in row i and column i' represents the probability that, if the DTMC is in state $s_i^{(j)}$, it will enter level $\mathcal{S}^{(j-1)}$ for the first time through state $s_{i'}^{(j-1)}$. $\mathbf{G}$ is stochastic iff the DTMC is recurrent, and the algorithm assumes that $\mathbf{G}$ is irreducible. Several steps are required to obtain the value of $\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(1)}$.

1. Let $\tilde{\mathbf{Q}} = (\mathbf{B} + \mathbf{L} + \sum_{j=1}^{\infty} \mathbf{F}^{(j)})$ be the transition probability matrix analogous to the one defined in Sect. 3.1 for the continuous case; compute the column vector $\boldsymbol{\beta} = (\mathbf{L} + \sum_{j=1}^{\infty} j \cdot \mathbf{F}^{(j)}) \cdot \mathbf{1}^T$, the probability vector $\tilde{\boldsymbol{\pi}}$ solution of $\tilde{\boldsymbol{\pi}} \cdot \tilde{\mathbf{Q}} = \tilde{\boldsymbol{\pi}}$, and $\rho = \tilde{\boldsymbol{\pi}} \cdot \boldsymbol{\beta}$.
2. Compute the matrix $\mathbf{G}$ minimal solution of $\mathbf{G} = \mathbf{B} + \mathbf{L} \cdot \mathbf{G} + \sum_{j=1}^{\infty} \mathbf{F}^{(j)} \cdot \mathbf{G}^{j+1}$ (see [5, 9] for algorithmic issues).
3. Compute the probability vector $\mathbf{g}$ satisfying $\mathbf{g} = \mathbf{g} \cdot \mathbf{G}$.
4. Compute the matrices $\mathbf{X}$ and $\mathbf{Y}$ solution of $\mathbf{X} = \hat{\mathbf{B}} + \left(\mathbf{L} + \sum_{j=1}^{\infty} \mathbf{F}^{(j)} \cdot \mathbf{G}^j\right) \cdot \mathbf{X}$ and $\mathbf{Y} = \sum_{j=1}^{\infty} \hat{\mathbf{F}}^{(j)} \cdot \mathbf{G}^{j-1} + \mathbf{L}^{(0)} \cdot \mathbf{Y}$.
5. Compute the stochastic matrices $\mathbf{K} = \mathbf{L}^{(0)} + \left(\sum_{j=1}^{\infty} \hat{\mathbf{F}}^{(j)} \cdot \mathbf{G}^{j-1}\right) \cdot \mathbf{X}$ and $\mathbf{H} = \hat{\mathbf{B}} \cdot \mathbf{Y} + \sum_{j=1}^{\infty} \mathbf{F}^{(j)} \cdot \mathbf{G}^j$.
6. Compute the probability vectors $\mathbf{k}$ and $\mathbf{h}$ satisfying $\mathbf{k} = \mathbf{k} \cdot \mathbf{K}$ and $\mathbf{h} = \mathbf{h} \cdot \mathbf{H}$.
7. Compute the column vectors

$$\boldsymbol{\phi}' = \left[\mathbf{I} - \mathbf{B} - \sum_{j=1}^{\infty} \mathbf{F}^{(j)} \cdot \mathbf{G}^j\right] \cdot \left[\mathbf{I} - \tilde{\mathbf{Q}} + (\mathbf{1}^T - \boldsymbol{\beta}) \cdot \mathbf{g}\right]^{-1} \cdot 1^T + (1-\rho)^{-1} \cdot \mathbf{B} \cdot \mathbf{1}^T$$

and

$$\boldsymbol{\phi}'' = \mathbf{1}^T + \left[\sum_{j=1}^{\infty} \hat{\mathbf{F}}^{(j)} - \sum_{j=1}^{\infty} \hat{\mathbf{F}}^{(j)} \cdot \mathbf{G}^{j-1}\right] \cdot \left[\mathbf{I} - \tilde{\mathbf{Q}} + (\mathbf{1}^T - \boldsymbol{\beta}) \cdot \mathbf{g}\right]^{-1} \cdot \mathbf{1}^T + (1-\rho)^{-1} \cdot \sum_{j=1}^{\infty} (j-1) \cdot \hat{\mathbf{F}}^{(j)} \cdot \mathbf{1}^T.$$

8. Compute the column vectors $\mathbf{k}' = \boldsymbol{\phi}'' + \sum_{j=1}^{\infty} \hat{\mathbf{F}}^{(j)} \cdot \mathbf{G}^{j-1} \cdot \left[\mathbf{I} - \mathbf{L} - \sum_{j=1}^{\infty} \mathbf{F}^{(j)} \cdot \mathbf{G}^j\right] \cdot \boldsymbol{\phi}'$ and $\mathbf{h}' = \boldsymbol{\phi}' + \hat{\mathbf{B}} \cdot (\mathbf{I} - \mathbf{L}^{(0)})^{-1} \cdot \boldsymbol{\phi}''$.
9. Finally, compute $\boldsymbol{\pi}^{(0)} = (\mathbf{k} \cdot \mathbf{k}')^{-1} \cdot \mathbf{k}$ and $\boldsymbol{\pi}^{(1)} = (\mathbf{h} \cdot \mathbf{h}')^{-1} \cdot \mathbf{h}$.

Once $\boldsymbol{\pi}^{(0)}$ and $\boldsymbol{\pi}^{(1)}$ are known, we can then iteratively compute $\boldsymbol{\pi}^{(j)}$ for $j = 2, 3$, etc., stopping when the accumulated probability mass is close to one. This can be a numerically intensive step when ρ is close to one since, in this case, the entries of $\boldsymbol{\pi}^{(j)}$ decrease slowly as j grows.

Comparing now ETAQA to Neuts's algorithm, it is only fair to point out immediately that, while in principle Neuts's algorithm can solve *all* M/G/1-type processes, our extended ETAQA approach can be applied only when the connectivity of the states in the CTMC satisfies certain conditions: we essentially require that $\mathbf{B}$ has a single nonzero column (after repartitioning the state space, if required). When ETAQA is applicable, however, it makes sense to investigate its advantages with respect to more general algorithms.

The condition we impose on $\mathbf{B}$ does simplify Neuts's algorithm considerably, as it implies that $\mathbf{G}$ can be obtained without any computation: $\mathbf{G}_{i,i'}$ is 1 if $i' = n$ and 0 otherwise. However, given the complete generality of the matrices $\mathbf{L}$ and $\mathbf{L}^{(0)}$, the computation of $\mathbf{X}$ and $\mathbf{Y}$ still requires considerable effort and, from that point on, the structure of $\mathbf{G}$ only reduces the computational complexity of Neuts's algorithm in the sense that all matrix products containing a factor $\mathbf{G}$ on the right can be performed efficiently and require only a vector of size n for their storage. However, the matrices $\mathbf{X}$, $\mathbf{Y}$, $\mathbf{H}$, and $\mathbf{K}$ must be computed and stored in their entirety (at least the last two; $\mathbf{X}$ and $\mathbf{Y}$ can be actually computed and stored one column at a time), and they are not sparse in general, hence the computational complexity of is $O(m^3 + n^3)$ and the storage complexity is

$O(m^2 + n^2)$. Furthermore, Neuts's algorithm computes a *finite* number of entries in $\boldsymbol{\pi}$, thus any stationary measure obtained from it is, in a sense, approximate because of this truncation.

Our extended ETAQA method, in addition to its appealing simplicity, allows us to exploit the sparsity of the blocks defining $\mathbf{Q}$, for both execution time and storage: the matrix $\mathbf{M}$ used by ETAQA (or the analogous one for the case of bounded bulk arrivals) has the same sparsity as the original blocks (the inverses $(\mathbf{I} - \hat{\mathbf{A}})^{-1}$ and $(\mathbf{I} - \mathbf{A})^{-1}$ require linear computation and storage, since the geometric scaling factors $\hat{\mathbf{A}}$ and $\mathbf{A}$ are diagonal matrices), and the iterative solution of the linear system in (4) requires time linear in the number of nonzero entries in $\mathbf{M}$. Furthermore, ETAQA allows instead to efficiently compute "mathematically exact" measures.

7 Applications

In this section, we describe two applications that can be solved by the extended ETAQA approach described in Sect. 3. We concentrate on presenting the CTMCs that models the application of interest and we focus on the form of the repeating matrix pattern that allows us to apply our technique.

7.1 Multiprocessor scheduling

Resource allocation in multiprocessor systems has been a favorite research topic for the performance and operating systems community in recent years. The number of users that attempt to use the system simultaneously, the parallelism of the applications and their respective computational and secondary storage needs, and the need to meet the execution deadlines are examples of issues that exacerbate the difficulty of the resource allocation problem.

A popular way to allocate processors among various competing applications is by space-sharing the processors: processors are partitioned in disjoint groups and each application executes in isolation on one of these groups. Space-sharing can be done in a static, adaptive, or dynamic way. If a job requires a fixed number of processors for execution, this requires a static space-sharing policy [12]. Adaptive space-sharing policies [1] have been proposed for jobs that can configure themselves to the number of processors allocated by the scheduler at the beginning of their execution. Dynamic space-sharing policies [6] have been proposed for jobs that are flexible enough to allow the scheduler to reduce or augment the number of processors allocated to each application in immediate response to environment changes. Because of their flexibility, dynamic policies can offer optimal performance but are the most expensive to implement because they reallocate resources while applications are executing.

Modeling the behavior of scheduling policies in parallel systems often results in CTMCs with matrix-geometric form [13]. Here, we present a CTMC that models the behavior of a restricted dynamic scheduling policy. A common solution to reducing the processor reconfiguration cost in dynamic policies is to limit the number of reallocations that can occur during the lifetime of a program [1]. Fig. 4 illustrates the CTMC of a dynamic scheduling policy that can only reduce the number of processor allocated to an application, and only immediately after a service completion. We restrict our attention to the system behavior under bursty arrival conditions modeled as bulks of finite size (see [11] for an analysis of adaptive space-sharing policies under the same assumptions for the arrival process).

The system state is described by the number of applications that are waiting for service in the queue and the number of applications that are in service. $NwMs_r$ denotes that there are N applications in the queue, waiting to be executed, and M applications in service, each having a fraction $\frac{1}{r}$ of the total number of processors allocated to it. Thus, the service rate of an application is μ_r, a function of r. For simplicity, the figure assumes that the maximum number of simultaneously executing applications is set to three and that the bulk size is at most four, but any other finite concurrency level and unbounded bulk sizes can be managed by our approach.

A deterministic set of transition rules specify the successor state for each state in the chain. The policy strives to guarantee that all executing applications are allocated an equal number of processors while minimizing the waiting queue length, but since reallocations occur only at departure times, there are states where there are fewer than three executing applications even when there are applications waiting in the queue. We further assume that the policy incurs no reallocation overhead to illustrate its ideal behavior.[1]

It is easy to observe that our lemma for the bounded bulk arrivals can be readily applied for the performance analysis of this policy. Since the behavior of this dynamic scheduling policy under different workloads is outside the scope of this paper, we do not present numerical results. We point out, however, that the average job response time can be easily calculated by first computing the average queue length as discussed in Sect. 3 and then applying Little's law.

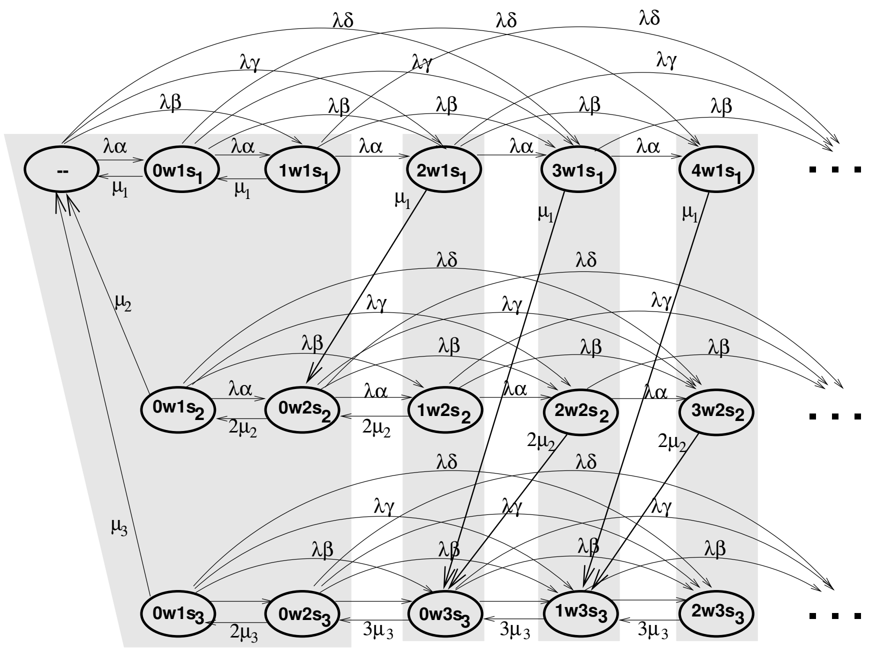

Figure 4: The CTMC that models a dynamic multiprocessor scheduling policy.

7.2 Self-monitoring and self-adjusting server

In many application domains it is common practice to dynamically add or remove servers so as to dynamically adjust server capacity to incoming workloads [3]. The motivation behind such

[1]This overhead can be accounted for by appropriately reducing the service rates when reallocation occurs.

self-adjusting systems is that it is desirable to operate using as few as possible servers when the load is low (and perhaps use the idle servers for other types of work), but at the same time being able to sustain heavier workload intensities by dynamically increasing the system capacity through the addition of more servers. Examples of such applications include protocols in communication networks, Internet information query services, and schedulers for concurrent bandwidth allocation to both multimedia and best-effort applications. Here, we concentrate on the behavior of a scheduler that serves applications in a time-sharing manner and adjusts its capacity as a function of the arrival rate intensity.

A generic CTMC that describes the behavior of such systems is illustrated in Fig. 5. The system state defines the number of applications in the system and the current service level. We assume that the server can operate at four levels, a, b, c, and d. Requests to the system may come from two sources. The first one is Poisson with parameter λ, while the second one is Poisson with parameter κ but may be of arbitrary bulk size. The bulk size is governed by a geometric distribution with parameter $1 - \rho$ (for the sake of clarity, Fig. 5 shows the bulk arrivals only for service level a).

The service station increases its capacity gradually, according to the current workload. The time to add a new server is exponentially distributed with parameter ν; this may occur during the service levels a, b, and c. We further assume that the reduction of server capacity is instantaneous and occurs only when the system empties completely (modeling an actual delay for taking servers offline can be easily accommodated, since it just increases the set of states $\mathcal{S}^{(0)}$).

It is easy to verify that our extended version of ETAQA can be readily applied after repartitioning the state space as shown by the grayed-out areas in Fig. 5.

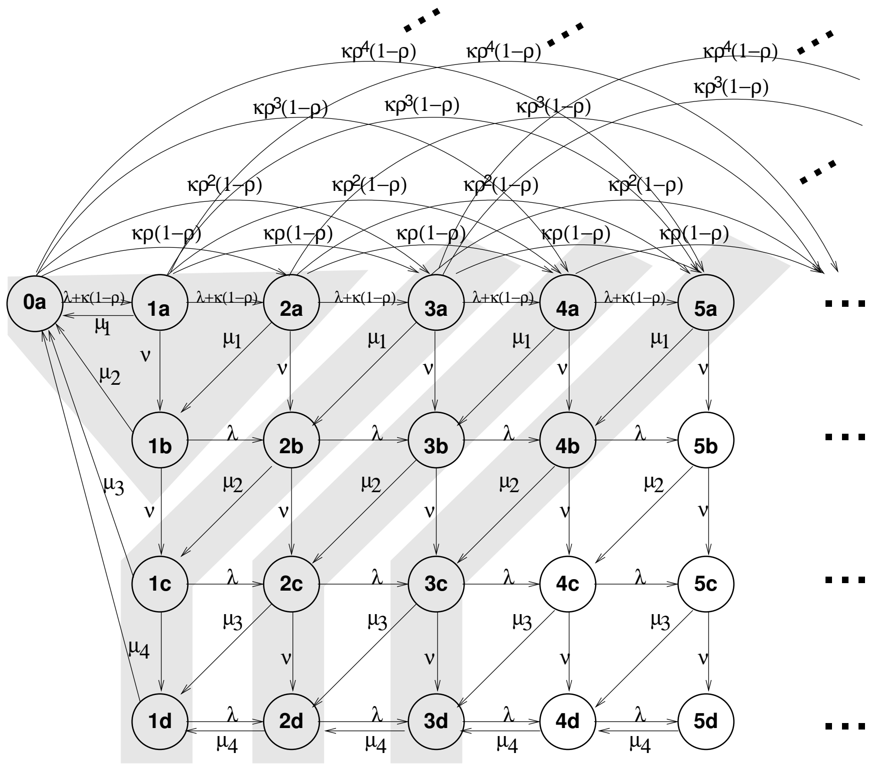

Figure 5: The CTMC that models a self-monitoring and self-adjusting server.

8 Conclusions and future work

In this paper we presented an extension of the ETAQA approach to M/G/1-type processes. Our exposition focuses on the description of the extended ETAQA methodology and its application to efficiently compute 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 wide variety of Markov reward functions.

We must emphasize that our treatment does not apply to all M/G/1-type processes. The necessary condition that must be satisfied is that all transitions from one level to the previous one are directed to a single state (but no restriction is placed on transitions within a level or toward higher levels). When this condition is satisfied, the solution is derived by solving a system of $m+2n$ linear equations and provides significant savings over standard algorithms for the solution of general M/G/1-type processes. Although the condition on $\mathbf{B}$ is indisputably restrictive, we present a set of applications from the computer systems field where extended ETAQA readily applies.

In the future, we expect to release a software tool that efficiently implements the extended ETAQA. Implementation of the tool is currently underway. We also intend to explore possibilities to extend the methodology to a wider class of M/G/1-type processes than those presented in this paper. To this end, the utility of approximation methods based on the extended ETAQA will be examined.

Acknowledgments

We would like to thank our colleague Prof. Paul Stockmeyer for furnishing us with the expression for the sum $\sum_{j=1}^{\infty} j^l \cdot \mathbf{A}^j$, and its relation to Euler numbers. There just isn't a substitute for having a good mathematician in the house!

References

[1] S.-H. Chiang, R. Mansharamani, and M. Vernon. Use of application characteristics and limited preemption for run-to-completion parallel processor scheduling policies. In *Proceedings of the 1994 Sigmetrics Conference on Measurement and Modeling of Computer Systems*, pages 33–44, May 1994.

[2] G. Ciardo and E. Smirni. ETAQA: An Efficient Technique for the Analysis of QBD-processes by Aggregation. Submitted for publication.

[3] L. Golubchik and J. C. Lui. Bounding of performance measures of a threashold-based queueing system with hysteresis. In *Proceedings of the 1997 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pages 147–156, June 1997.

[4] G. Latouche. Algorithms for infinite Markov chains with repeating columns. In C. Meyer and R. J. Plemmons, editors, *Linear Algebra, Markov Chains, and Queueing Models*, volume 48 of *IMA Volumes in Mathematics and its Applications*, pages 231–265. Springer-Verlag, 1993.

[5] G. Latouche and G. W. Stewart. Numerical methods for M/G/1 type queues. In W. J. Stewart, editor, *Computations with Markov Chains*, pages 571–581. Kluwer, Boston, MA, 1995.

[6] C. McCann, R. Vaswani, and J. Zahorjan. A dynamic processor allocation policy for multiprogrammed shared memory multiprocessors. *ACM Transactions on Computer Systems*, 11(2):146–178, 1993.

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

[8] M. F. Neuts. *Matrix-geometric solutions in stochastic models.* Johns Hopkins University Press, Baltimore, MD, 1981.

[9] M. F. Neuts. *Structured stochastic matrices of M/G/1 type and their applications.* Marcel Dekker, New York, NY, 1989.

[10] J. Riordan. *An Introduction to Combinatorial Analysis.* Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, 1958.

[11] E. Rosti, E. Smirni, G. Serazzi, L. Dowdy, and K. Sevcik. On processor saving scheduling policies for multiprocessor systems. *IEEE Trans. Comp.*, 47(2):178–189, Feb. 1998.

[12] E. Smirni, E. Rosti, L. Dowdy, and G. Serazzi. A methodology for the evaluation of multiprocessor non-preemptive allocation policies. *Journal of Systems Architecture*, 44:703–721, 1998.

[13] M. Squillante, F. Wang, and M. Papaefthymiou. Stochastic analysis of gang scheduling in parallel and distributed systems. *Perf. Eval.*, 27/28:273–296, 1996.

[14] G. W. Stewart. On the solution of block hessenberg systems. *Numerical Linear Algebra with Applications*, 2:287–296, 1995.