Queueing Models and
Service Management
Vol. 1, No. 2, page 059-078, 2018

QMSM

Physical Interpretations for Quasi-Birth-and-Death Process Algorithms

Nigel Bean[1,*] Guy Latouche[2] and Peter Taylor[3]

[1]School of Mathematical Sciences
University of Adelaide, SA 5005, Australia
[2]Département d'Informatique, Université libre de Bruxelles
CP 212, Boulevard du Triomphe, 1050 Bruxelles, Belgium
[3]School of Mathematics and Statistics
University of Melbourne, Vic 3010, Australia
(Received February 2018; accepted May 2018)

Abstract: The solution of polynomial matrix equations lies at the heart of the analysis of quasi-birth-and-death processes (QBDs), fluid queues and other random walks on a strip in the plane. Many algorithms have been proposed (and are still being proposed) to solve these equations. In order to improve upon one algorithm, or to understand the qualities which make one better than another, it often helps to use our physical understanding of the behaviour of the process. We illustrate this here by considering algorithms for the solution of the basic quadratic equation for QBDs, with a particular reference to Newton's Method.

Keywords: Matrix quadratic equations, matrix-analytic methods, quasi-birth-and-death processes, Newton's iterations.

1. Introduction

The theory of matrix-analytic methods emphasizes the importance of being able to perform computations, so that qualitative (or structural) analysis may be accompanied by quantitative (or numerical) evaluation. Matrix-analytic algorithms, which were first developed in the applied probability community, are not elementary. They have drawn the attention of numerical analysts, and there is now an interplay between the two fields, the thrust towards new progress sometimes coming from a reflection on the physical properties of a stochastic system, and sometimes from the application of techniques well grounded in numerical analysis.

Bean *et al.* [2] analysed various iterative procedures for solving an algebraic Riccati equation stemming from the analysis of fluid queues via matrix-analytic methods. They related the successive approximations in the various algorithms to different types of constrained behaviour of the stochastic system. One of the procedures applied in Bean *et al.* [2] was Newton's method. In particular, the authors gave an interpretation of Newton's method in terms of the dynamics of the model.

* Corresponding author
Email : nigel.bean@adelaide.edu.au

This analysis prompted one of the authors of the present paper (Peter Taylor) to assert that 'All matrix-analytic algorithms have a physical interpretation', to which one of the other authors (Guy Latouche) replied 'All of them?'. The purpose of this paper is to explore this issue in the context of the various algorithms that have been proposed for one of the simplest classes of system amenable to matrix-analytic methods, the quasi-birth-and-death processes (QBDs), and, in particular, with respect to Newton's method. Within the context of stochastic fluid models, Latouche and Nguyen [9, Section 9] have recently considered a similar question, relating the physical interpretation of one of the algorithms that has been proposed for fluid models to the behaviour of a stack.

In the next section, we shall set up our notation and recall some basic facts about QBDs, including the physical interpretations of the original linear algorithms that were proposed for analysing QBDs by Neuts [11] and Latouche [7], and the quadratically-convergent Logarithmic Reduction and the Cyclic Reduction algorithms of Latouche and Ramaswami [10] and Bini and Meini [4].

In Section 3, we shall describe Newton's method as applied to QBDs and give a physical interpretation for the iterates which has a surprisingly different character to the corresponding interpretation for the other algorithms. In Section 4, we shall compare the efficiency of Newton's method to that of the Logarithmic Reduction and the Cyclic Reduction algorithms. We shall conclude with a few recommendations in Section 5.

2. Background

Discrete-time QBDs are two-dimensional Markov chains $\{(X_k, \varphi_k), k \geq 0\}$ on the state space $\mathbb{N} \times \{1, \ldots, m\}$, where here we shall take m to be finite. The only possible transitions are

- from (k,i) to $(k+1, j)$, $k \geq 0$, $1 \leq i, j \leq m$, with transition probability $[A_1]_{ij}$,
- from (k,i) to (k, j), $k \geq 1$, $1 \leq i, j \leq m$, with transition probability $[A_0]_{ij}$,
- from (k,i) to $(k-1, j)$, $k \geq 1$, $1 \leq i, j \leq m$, with transition probability $[A_{-1}]_{ij}$,
- from $(0,i)$ to $(0, j)$, $1 \leq i, j \leq m$, with transition probability B_{ij}.

Thus, if we arrange the states in lexicographic order, the transition matrix of the QBD has the structure

$$P = \begin{bmatrix} B & A_1 & & & \\ A_{-1} & A_0 & A_1 & & \\ & A_{-1} & A_0 & A_1 & \\ & & A_{-1} & A_0 & \ddots \\ & & & \ddots & \ddots \end{bmatrix}. \qquad (1)$$

The component X of the state space is usually called the *level* and the component φ the *phase*.

Away from level 0, the matrix $A \equiv A_{-1} + A_0 + A_1$ describes transitions in the phase, independently of the level. For a discrete-time QBD, let $\mathbf{x}$ be the solution to

$$\mathbf{x}A = \mathbf{x}. \tag{2}$$

Then, with $\mathbf{e}'$ a column of ones, the chain is positive recurrent, null recurrent or transient according as

$$\mathbf{x}A_1\mathbf{e}' - \mathbf{x}A_{-1}\mathbf{e}'$$

is less than, equal to or greater than zero [11].

Write the stationary distribution of a positive recurrent QBD as $\boldsymbol{\pi} = (\boldsymbol{\pi}_0, \boldsymbol{\pi}_1, \ldots)$. The well-known matrix geometric property [11] states that there exists a matrix R such that

$$\boldsymbol{\pi}_k = \boldsymbol{\pi}_0 R^k \tag{3}$$

The vector $\boldsymbol{\pi}_0$ satisfies

$$\boldsymbol{\pi}_0 \left[B + RA_{-1} \right] = \boldsymbol{\pi}_0 . \tag{4}$$

The matrix R is the minimal nonnegative solution to the matrix equation

$$R = A_1 + RA_0 + R^2 A_{-1}. \tag{5}$$

For any level k, the (i, j)th entry of R can be interpreted as

- the expected number of visits to phase j of level $k+1$ before first return to level k conditional on the QBD starting in phase i of level k.

The matrix R has spectral radius which is less than or equal to one, and the QBD is positive recurrent if and only if the spectral radius of R is less than one.

The matrix G, which is the minimal nonnegative solution of

$$G = A_{-1} + A_0 G + A_1 G^2, \tag{6}$$

also has a physical interpretation. For any level $k > 0$, the (i, j)th entry of G is

- the probability that the QBD first visits level $k-1$ in phase j conditional on it starting in phase i of level k.

The matrix G also has spectral radius which is less than or equal to one, and the QBD is recurrent if and only if the spectral radius of G is equal to one. With the matrix G at hand we are in a position to solve various hitting probability and expected hitting time problems. Furthermore, for a QBD, the matrix R can be written in terms of the matrix G via the relation

$$R = A_1 \left[I - A_0 - A_1 G \right]^{-1}, \tag{7}$$

and so the stationary distribution can also be easily derived if we know G. Since it is a matrix of probabilities, rather than expected values, the matrix G is a 'nicer' object to work with than R. For this reason, we concentrate on methods for deriving G via equation (6). This equation has an analytic solution only in a few very special cases, and so we almost always have to resort to numerical solution.

We start with the simple procedure recommended by Neuts [11]. For an irreducible discrete-time QBD, $I - A_0$ is invertible. So, an obvious first approach to solving equation (6) is to transform it into the fixed-point equation

$$G = (I - A_0)^{-1}\left[A_{-1} + A_1 G^2\right] \tag{8}$$

and use the iterative procedure

$$\bar{G}_{n+1} = (I - A_0)^{-1}\left[A_{-1} + A_1 \bar{G}_n^2\right] \tag{9}$$

with $\bar{G}_0 = 0$.

Neuts showed that, with this iteration, $\bar{G}_n$ does converge to G. Furthermore, except when the QBD is null-recurrent, this convergence is *linear*. That is, there exists a constant $\nu \in (0,1)$ such that

$$\limsup_{n\to\infty} \| \bar{G}_n - G \|^{1/n} = \nu. \tag{10}$$

The type of question that we shall be interested in is *'Can we give a physical interpretation to the n-th iterate of procedures such as the one described above?'* For Neuts' original iteration (9), this question has not had a precise answer until recently. It can be understood in terms of iterations for tree-structured QBDs, see Bean *et al.* [1, Section 6].

In general, to understand physical interpretations of the type that we shall discuss here, we need to know about *censoring*. Consider an irreducible, finite-state, discrete-time Markov chain whose states are partitioned into two sets E_1 and E_2. This induces a partitioning of its transition matrix T so that

$$T = \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix}. \tag{11}$$

The irreducibility of T means that the spectral radius of T_{22} must be strictly less than one. We can then note that the stationary distribution $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2)$ that satisfies $\boldsymbol{\alpha} T = \boldsymbol{\alpha}$ also satisfies

$$\boldsymbol{\alpha}_1 = \boldsymbol{\alpha}_1\left[T_{11} + T_{12}\left(I - T_{22}\right)^{-1} T_{21}\right] \tag{12}$$

with

$$\boldsymbol{\alpha}_2 = \boldsymbol{\alpha}_1 T_{12}\left(I - T_{22}\right)^{-1} \tag{13}$$

Observe that

$$\left(I - T_{22}\right)^{-1} = \sum_{k=0}^{\infty} T_{22}^{k}$$

and so

$$\boldsymbol{\alpha}_1 \left[T_{11} + T_{12}\left(I - T_{22}\right)^{-1} T_{21} \right] = \boldsymbol{\alpha}_1 \left[T_{11} + T_{12} \left[\sum_{k=0}^{\infty} T_{22}^{k} \right] T_{21} \right]$$

and we can interpret $\boldsymbol{\alpha}_1 / (\boldsymbol{\alpha}_1 e')$ as the stationary distribution of the *censored* discrete-time Markov chain observed only when it is in E_1.

Similar comments can be made for a positive-recurrent, discrete-time Markov chain where E_2 is infinite. This follows because positive recurrence on E ensures that the expected total time spent during any sojourn in E_2 is finite, and so $\sum_{k=0}^{\infty} T_{22}^{k}$ converges elementwise.

Indeed, it is not only true that $\boldsymbol{\alpha}_1 / (\boldsymbol{\alpha}_1 e')$ is the stationary distribution of the censored discrete-time Markov chain, but

$$\left[T_{11} + T_{12} \left[\sum_{k=0}^{\infty} T_{22}^{k} \right] T_{21} \right] \tag{14}$$

is its transition matrix.

When E_2 is infinite, we can allow the chain with transition matrix T to be transient, in which case there is a positive probability that the Markov chain may leave E_1 and not return, which results in the matrix (14) being substochastic, and this interpretation will still hold. It follows that transient measures of the censored chain can be derived using standard methods applied to the transition matrix (14). We can also observe that the (i, j) th entry of

$$\left[I - T_{22}\right]^{-1} T_{21} = \left[\sum_{k=0}^{\infty} T_{22}^{k} \right] T_{21} \tag{15}$$

is the probability that the Markov chain first enters E_1 in state j given that it started in state i of E_2.

We claimed above that it was hard to give a physical interpretation of Neuts' algorithm (9). Neuts did, however, propose a second algorithm in Section 1.9 of Neuts [11] and Latouche [7] gave a simple physical interpretation for the iterates of this algorithm. Observe that

$$G = (I - A_0 - A_1 G)^{-1} A_{-1}, \tag{16}$$

and use the iteration

$$\underline{G}_{n+1} = (I - A_0 - A_1 \underline{G}_n)^{-1} A_{-1}, \tag{17}$$

with $\underline{G}_0 = 0$. We can show inductively that all the inverses exist.

The matrix $\underline{G}_1 = (I - A_0)^{-1} A_{-1} = \left[\sum_{\ell=0}^{\infty} A_0^{\ell} \right] A_{-1}$. It follows that the (i, j) th entry of $\underline{G}_1$ is

- the probability that the QBD never reaches level $k+1$ and first visits level $k-1$ in phase j, conditional on it starting in phase i of level k.

We can use induction to show that the (i,j)th entry of $\underline{G}_n$ is

- the probability that the QBD never reaches level $k+n$ and first visits level $k-1$ in phase j, conditional on it starting in phase i of level k.

Thus, the successive iterates of this algorithm have the same physical interpretation as that of the matrix G, but with a *taboo level* that increases linearly. Like the original algorithm of Neuts, this algorithm is linearly convergent except when the Markov chain is null-recurrent.

The two algorithms that are currently used as benchmarks for analysing QBDs (and, indeed, more general matrix-analytic models) are the Logarithmic Reduction algorithm of Latouche and Ramaswami [10] and the Cyclic Reduction algorithm of Bini and Meini [4]. Precise descriptions of these algorithms are given in Appendices 6.1 and 6.2 respectively, and a good explanation of the thinking behind them can be found in Bini *et al.* [3]. If the reader wants to implement them, we recommend using the Matlab packages available from Van Houdt's web-site at Bini *et al.* [5]. A number of speed-up features, such as transforming the matrices to move eigenvalues away from the unit circle and using Fast Fourier Transforms, are included in these implementations.

For the Logarithmic Reduction algorithm, we can write

$$G=\sum_{\ell=0}^{\infty}K^{(\ell)},$$

where, for any $k>0$,

$$K_{ij}^{(\ell)}=\mathrm{P}[\gamma(k+2^{\ell}-1)<\gamma(k-1)<\gamma(k+2^{\ell+1}-1),\varphi_{\gamma(k-1)}=j\,|\,X_0=k,\varphi_0=i]$$

and $\gamma(\ell)$ is the time of first visit to level ℓ. If, in the execution of Algorithm 6.1, Step 6.1 is performed n times, then one obtains the approximation $\widehat{G}^{(n)}=\sum_{0\le\ell\le n}K^{(\ell)}$, meaning that the physical interpretation of the (i,j)th entry of $\widehat{G}^{(n)}$ is as

- the probability, conditional on it starting in phase i of level k, that the QBD first visits level $k-1$ in phase j and never reaches level $k+2^{n+1}-1$.

Cyclic reduction is related to a slightly different sequence, based on the fact that

$$G=\lim_{\ell\to\infty}J^{(\ell)},$$

where, for any $k>0$,

$$J_{ij}^{(\ell)}=\mathrm{P}[\gamma(k-1)<\gamma(k+2^{\ell}),\varphi_{\gamma(k-1)}=j\,|\,X_0=k,\varphi_0=i].$$

If, in the execution of Algorithm 6.2, the sequence of matrices $J^{(\ell)}$ is truncated after n

iterations, then one obtains the approximation $\widetilde{G}^{(n)} = J^{(n)}$, whose (i, j) th entry has an interpretation as

- the probability, conditional on it starting in phase i of level k, that the QBD first visits level $k-1$ in phase j and never reaches level $k+2^n$.

We see that the two algorithms have very similar performance in that they compute nearly the same sequence of approximations. One would expect that, for a given precision, Algorithm 6.1 might require one iteration less than 6.2, but, on the other hand, each iteration of Algorithm 6.1 requires two more matrix multiplications than an iteration of 6.2 and so it takes more time. One might also expect that the convergence of both algorithms would be *quadratic* in the sense that there is a $\nu \in (0,1)$ such that

$$\limsup_{n\to\infty} \| \widehat{G}_n - G \|^{1/2^n} = \nu, \tag{18}$$

with a similar statement also true for $\widetilde{G}_n$. This is, in fact, the case unless the QBD is null-recurrent Bini *et al.* [3].

3. Newton's Method

We start with the equation for G, written in the form (16). If we apply Newton's method to the solution of this equation, we obtain the sequence

$$G_{\mathrm{N}}^{(n+1)} - W^{(n)} A_1 G_{\mathrm{N}}^{(n+1)} W^{(n)} A_{-1} = W^{(n)} A_{-1} - W^{(n)} A_1 G_{\mathrm{N}}^{(n)} W^{(n)} A_{-1} \tag{19}$$

where

$$W^{(n)} = (I - A_0 - A_1 G_{\mathrm{N}}^{(n)})^{-1} \tag{20}$$

and $G_{\mathrm{N}}^{(0)} = 0$. It was shown in Latouche [8] that for any initial matrix $G_{\mathrm{N}}^{(0)}$ with $0 \le G_{\mathrm{N}}^{(0)} \le G$, the sequence (19) converges monotonically and quadratically to G. A description of the algorithm is given in Appendix 6.3.

Also in [8], Latouche provided an algorithm for evaluating the sequence of matrices $\{G_{\mathrm{N}}^{(n)}\}$. The difficult part of this is solving equation (19), which is a special case of a Sylvester equation, for $G_{\mathrm{N}}^{(n+1)}$ in terms of $G_{\mathrm{N}}^{(n)}$. Latouche provided an algorithm in which (19) is transformed into a standard linear system by concatenating the columns of $G_{\mathrm{N}}^{(n+1)}$ and writing the coefficient matrix as a direct sum involving $W^{(n)} A_1$ and $W^{(n)} A_{-1}$. Using this transformation, he showed that each iteration of the algorithm has a complexity of order m^6.

Latouche went on to test Newton's algorithm against the best known algorithm of the time (the sequence (17)) on a suite of examples and found that, while Newton's algorithm required up to an order of magnitude fewer iterations, it could take up to an order of magnitude longer in terms of computer time to calculate G to within a given tolerance.

Since Latouche [8], it would be fair to say that the conventional wisdom in the matrix-

analytic community is that, even though it is a quadratically convergent algorithm, the complexity of each iteration of Newton's method makes it uncompetitive with other algorithms that have been proposed for solving (6). This attitude has only been reinforced by the later discovery of the quadratically-convergent Logarithmic Reduction and Cyclic Reduction algorithms.

However, there is actually an m^3 algorithm for solving the Sylvester equation (19) (see Gardiner *et al.* [6]). This motivated us to revisit the question of how useful Newton's method is for solving (6). At the same time, we considered whether we could give a physical description of the sample paths whose probability is recorded in the n th iteration $G_{\mathrm{N}}^{(n+1)}$. A physical description is given in this section, while we report in Section 4 a comparison of our numerical experience with the analysis of QBDs using Newton's method implemented according to Gardiner *et al.* [6], and via the Logarithmic and Cyclic Reduction algorithms.

As with the iterations of the Logarithmic Reduction and Cyclic Reduction algorithms, for $k > 0$, the matrices $G_{\mathrm{N}}^{(n)}$ contain the probabilities of certain sets of sample paths that start in level k and end in level $k-1$; we shall denote the set associated with $G_{\mathrm{N}}^{(n)}$ for $k = 1$ by $\mathcal{G}_{\mathrm{N}}^{(n)}$ and so

$$(G_{\mathrm{N}}^{(n)})_{ij} = \Pr[\{(X_t, \varphi_t)\} \in \mathcal{G}_{\mathrm{N}}^{(n)}, \varphi_{\gamma(0)} = j \mid X_0 = 1, \varphi_0 = i].$$

For $n = 0$, equation (19) can be written as

$$G_{\mathrm{N}}^{(1)} = W^{(0)} A_{-1} + W^{(0)} A_1 G_{\mathrm{N}}^{(1)} W^{(0)} A_{-1} \tag{21}$$

$$= \sum_{\ell=1}^{\infty} \left(W^{(0)} A_1\right)^{\ell-1} \left(W^{(0)} A_{-1}\right)^{\ell} \tag{22}$$

where the second equation follows by repeatedly replacing $G_{\mathrm{N}}^{(1)}$ in the right hand side by the right hand side itself. To understand the physical meaning of (22), it is necessary to associate sets of trajectories with the matrices $U_0 \equiv W^{(0)} A_1$ and $D_0 \equiv W^{(0)} A_{-1}$. As

$$W^{(0)} A_1 = \sum_{\nu \geq 0} A_0^{\nu} A_1,$$

we see that $(U_0)_{ij}$ is the probability that, starting from $(1,i)$, the QBD moves to $(2,j)$ before moving to level 0. Similarly, $(D_0)_{ij}$ is the probability that the QBD moves from $(1,i)$ to $(0,j)$ before reaching level 2. Formally,

$$(U_0)_{ij} = P[\gamma(2) < \gamma(0), \varphi_{\gamma(2)} = j \mid X_0 = 1, \varphi_0 = i]$$

and

$$(D_0)_{ij} = P[\gamma(0) < \gamma(2), \varphi_{\gamma(0)} = j \mid X_0 = 1, \varphi_0 = i].$$

We see from (22) that $\mathcal{G}_{\mathrm{N}}^{(1)}$ may be interpreted as follows: starting from level 1, the QBD progressively moves to levels 2, 3, ..., ℓ for some ℓ, without going down, then moves

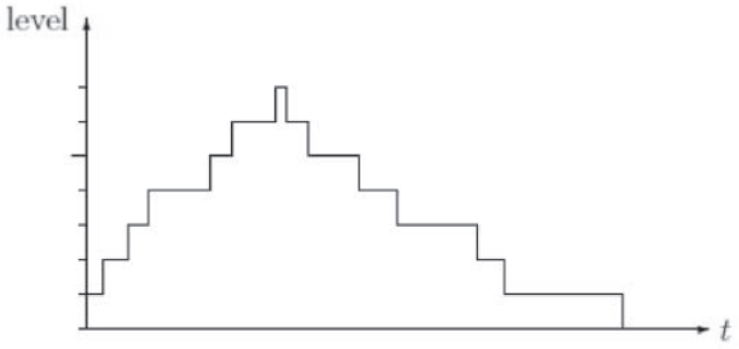

Figure 1. A sample path that contributes to $\mathcal{G}_{\mathrm{N}}^{(1)}$.

down to levels $\ell-1$, $\ell-2$, ..., 0 without going up again. Loosely stated, the sample paths in $\mathcal{G}_{\mathrm{N}}^{(1)}$ are those that have a "single peak", no matter how high. An example with $\ell=7$ is given in Figure 1.

To give a description of $\mathcal{G}_{\mathrm{N}}^{(n)}$ for general n, we introduce some set notation and two set operators. First observe that the spatial homogeneity of the transition matrix (1) ensures that

$$(U_0)_{ij} = P[\gamma(k+1) < \gamma(k-1),\ \varphi_{\gamma(k+1)} = j \mid X_0 = k,\ \varphi_0 = i]$$

and

$$(D_0)_{ij} = P[\gamma(k-1) < \gamma(k+1),\ \varphi_{\gamma(k)} = j \mid X_0 = k,\ \varphi_0 = i].$$

independently of k for any $k \geq 1$. Now define

- $\mathcal{U}_0$ to be the set of trajectories associated with U_0: the QBD stays at its initial level for a while and eventually jumps one level up, and so

  $$(U_0)_{ij} = \Pr[\{(X_t, \varphi_t)\} \in \mathcal{U}_0,\ \varphi_{\gamma(k+1)} = j \mid X_0 = k,\ \varphi_0 = i],$$

  independently of k;
- $\mathcal{D}_0$ is the set of trajectories associated with D_0: the QBD stays at its initial level before jumping one level down, and so

  $$(D_0)_{ij} = \Pr[\{(X_t, \varphi_t)\} \in \mathcal{D}_0,\ \varphi_{\gamma(k-1)} = j \mid X_0 = k,\ \varphi_0 = i],$$

  independently of k.

Furthermore, we define the concatenation operator "|" to join trajectories from two sets. For example, in $\mathcal{U}_0 \mid \mathcal{U}_0$, the QBD starts at some level k, moves to level $k+1$ without visiting level $k-1$, and then moves to level $k+2$ without having returned to level k.

We shall also use the operator "^{*n}" to denote concatenation of n trajectories from identical sets. For example, $\mathcal{U}_0^{*3} = \mathcal{U}_0 \mid \mathcal{U}_0 \mid \mathcal{U}_0$.

With these, we may associate with (21, 22) the set equations

$$\mathcal{G}_{\mathrm{N}}^{(1)} = \mathcal{D}_0 \cup \mathcal{U}_0 \,|\, \mathcal{G}_{\mathrm{N}}^{(1)} \,|\, \mathcal{D}_0 = \bigcup_{\ell \geq 0} \mathcal{U}_0^{*\ell} \,|\, \mathcal{D}_0 \,|\, \mathcal{D}_0^{*\ell}. \tag{23}$$

For general n, we define

$$U_n = W^{(n)} A_1 \qquad \textit{and} \qquad D_n = W^{(n)} A_{-1}. \tag{24}$$

Lemma 3.1. *The sets $\mathcal{U}_n$ and $\mathcal{D}_n$ of trajectories associated with U_n and D_n are such that*

$$\mathcal{D}_n = \bigcup_{m \geq 0} (\mathcal{U}_0 \,|\, \mathcal{G}_{\mathrm{N}}^{(n)})^{*m} \,|\, \mathcal{D}_0, \tag{25}$$

$$\mathcal{U}_n = \bigcup_{m \geq 0} (\mathcal{U}_0 \,|\, \mathcal{G}_{\mathrm{N}}^{(n)})^{*m} \,|\, \mathcal{U}_0, \tag{26}$$

Proof. By (24)

$$\begin{aligned} D_n &= (I - A_0 - A_1 G_{\mathrm{N}}^{(n)})^{-1} A_{-1} \\ &= (I - (I - A_0)^{-1} A_1 G_{\mathrm{N}}^{(n)})^{-1} (I - A_0)^{-1} A_{-1} \\ &= (I - U_0 G_{\mathrm{N}}^{(n)})^{-1} D_0 \\ &= \sum_{m \geq 0} (U_0 G_{\mathrm{N}}^{(n)})^m D_0 \end{aligned} \tag{27}$$

The interpretation of $U_0 G_{\mathrm{N}}^{(n)}$ is that, starting from some level k, the QBD eventually jumps to level $k+1$ and later returns to level k by following the constraints that characterise $G_{\mathrm{N}}^{(n)}$. Corresponding to the mth term in the series (27) this is repeated exactly m times before the QBD eventually moves to level $k-1$. This justifies (25), the proof of (26) is similar.

Theorem 3.2. *For $n \geq 0$, the set of trajectories associated with the nth iteration of Newton's method is given by the set*

$$\mathcal{G}_{\mathrm{N}}^{(n+1)} = \bigcup_{\ell \geq 0} \mathcal{U}_n^{*\ell} \,|\, \mathcal{D}_n \,|\, \mathcal{D}_n^{*\ell}. \tag{28}$$

Before giving the proof, which requires several steps, a few remarks are in order.

The first is that (28) is very similar to (23): at each iteration, the QBD moves step by step through a series of 'plateaus' from level 1 to some level $\ell+1$, then reverses itself and moves down to level 0, again through a series of plateaus. During a plateau at level $\nu \leq \ell+1$, the QBD is allowed to move above level ν and back but it must do so via a sample path adhering to the constraints in the previous iteration.

In Figure 2, we give a sample trajectory in $\mathcal{G}_{\mathrm{N}}^{(2)}$, with times τ_1, τ_2, τ_3 and τ_4 the initiation times for the plateaus on the way up to level 4 and the times θ_3, θ_2 and θ_1 the initiation times for the plateaus on the way down to level 0. Specifically,

- on the way up to level $\ell = 4$, Plateau 1 is from time τ_1 to τ_2, Plateau 2 from τ_2 to τ_3, Plateau 3 from τ_3 to τ_4 and Plateau 4 from τ_4 to θ_3;
- on the way down, we have Plateau 3 from θ_3 to θ_2, Plateau 2 from θ_2 to θ_1 and Plateau 1 from θ_1 to θ_0, at which time the QBD is at level 0.

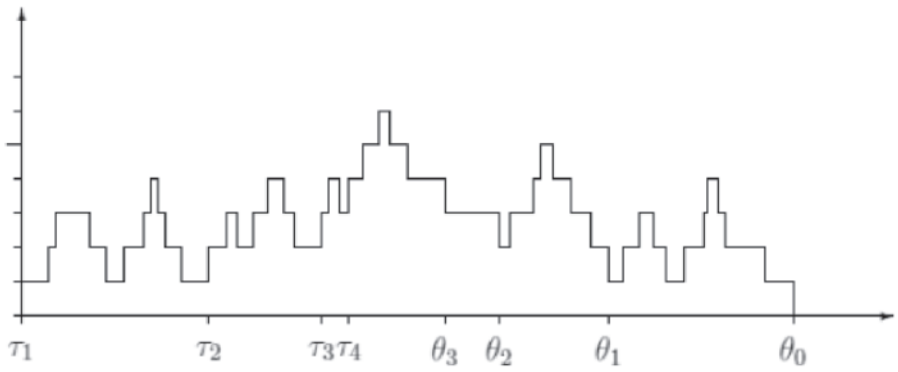

Figure 2. One possible trajectory for the second iteration of Newton's algorithm.

- During the interval (τ_1, τ_2) there are two excursions to higher levels with profiles in $\mathcal{G}_{\mathrm{N}}^{(1)}$ and, similarly, two excursions in (τ_2, τ_3) and (θ_1, θ_0), one in each of (τ_3, τ_4), (τ_4, θ_3) and (θ_2, θ_1) and none during (θ_3, θ_2).

A major difficulty with the analysis to follow results from the fact that the sets $\mathcal{S}_n^{(\ell)} \equiv \mathcal{U}_n^{*\ell} \,|\, \mathcal{D}_n \,|\, \mathcal{D}_n^{*\ell}$ are not disjoint over different values of ℓ. Indeed, we have shown above that the sample path in Figure 2 belongs to $\mathcal{S}_2^{(4)}$ but it also belongs to $\mathcal{S}_2^{(3)}$ if one forgets about τ_4 and θ_3, and takes the whole interval (τ_3, θ_2) as a plateau at level 3, with two excursions in $\mathcal{G}_{\mathrm{N}}^{(1)}$. Furthermore, the sample path also belongs to $\mathcal{S}_2^{(5)}$: to see that, we need to add epochs τ_5 for the first jump to level 5 after τ_4, and θ_4 for the first jump to level 4 after τ_5.

A major step in understanding the physical interpretation of Newton's method is to write $\mathcal{G}_{\mathrm{N}}^{(n+1)}$ as a union of disjoint sets.

To this end, rewrite (19) as

$$G_{\mathrm{N}}^{(n+1)} = D_n + U_n (G_{\mathrm{N}}^{(n+1)} - G_{\mathrm{N}}^{(n)}) D_n, \tag{29}$$

so that

$$\begin{aligned} T_{n+1} &\equiv G_{\mathrm{N}}^{(n+1)} - G_{\mathrm{N}}^{(n)} = D_n - G_{\mathrm{N}}^{(n)} + U_n (G_{\mathrm{N}}^{(n+1)} - G_{\mathrm{N}}^{(n)}) D_n \\ &= \sum_{\ell \geq 0} U_n^{\ell} (D_n - G_{\mathrm{N}}^{(n)}) D_n^{\ell} \end{aligned} \tag{30}$$

and finally

$$G_{\mathrm{N}}^{(n+1)} = D_n + \sum_{\ell \geq 1} U_n^{\ell} (D_n - G_{\mathrm{N}}^{(n)}) D_n^{\ell}. \tag{31}$$

Lemma 3.3. *For $n \geq 1$, the difference $D_n - G_{\mathrm{N}}^{(n)}$ is given by*

$$D_n - G_{\mathrm{N}}^{(n)} = (U_{n-1} T_n)^2 D_n. \tag{32}$$

Proof. By (29),

$$G_{\mathrm{N}}^{(n+1)} = D_n + U_n T_{n+1} D_n,$$

so that

$$D_n - G_{\mathrm{N}}^{(n)} = D_n - D_{n-1} - U_{n-1}T_nD_{n-1}. \tag{33}$$

Furthermore,

$$\begin{aligned} D_n &= (I - U_0G_{\mathrm{N}}^{(n)})^{-1}D_0 \\ &= (I - U_0(G_{\mathrm{N}}^{(n-1)} + T_n))^{-1}D_0 \\ &= (I - (I - U_0G_{\mathrm{N}}^{(n-1)})^{-1}U_0T_n)^{-1}(I - U_0G_{\mathrm{N}}^{(n-1)})^{-1}D_0 \\ &= (I - U_{n-1}T_n)^{-1}D_{n-1}. \end{aligned} \tag{34}$$

This, together with (33), shows that

$$\begin{aligned} D_n - G_{\mathrm{N}}^{(n)} &= \left[(I - U_{n-1}T_n)^{-1} - I - U_{n-1}T_n\right]D_{n-1} \\ &= \sum_{\ell \geq 2}(U_{n-1}T_n)^{\ell}D_{n-1} \end{aligned}$$

which, on account of (34), we may rewrite as (32).

Denote by $\mathcal{R}_n$ the set of trajectories associated with $D_n - G_{\mathrm{N}}^{(n)}$. From (32) we conclude that

$$\mathcal{D}_n = \mathcal{G}_{\mathrm{N}}^{(n)} \bigcup \mathcal{R}_n \tag{35}$$

where the two sets on the right are disjoint. Now, using the facts that

$$U_{n-1} = (I - U_0G_{\mathrm{N}}^{(n-1)})^{-1}U_0 = \sum_{m=0}^{\infty}(U_0G_{\mathrm{N}}^{(n-1)})^mU_0$$

and

$$D_n = (I - U_0G_{\mathrm{N}}^{(n)})^{-1}D_0 = \sum_{m=0}^{\infty}(U_0G_{\mathrm{N}}^{(n)})^mD_0,$$

we can write the right-hand side of (32) in expanded form as

$$\begin{aligned} (U_{n-1}T_n)^2D_n = \sum_{m_1,m_2,m_3 \geq 0} &(U_0G_{\mathrm{N}}^{(n-1)})^{m_1}U_0(G_{\mathrm{N}}^{(n)} - G_{\mathrm{N}}^{(n-1)}) \\ &(U_0G_{\mathrm{N}}^{(n-1)})^{m_2}U_0(G_{\mathrm{N}}^{(n)} - G_{\mathrm{N}}^{(n-1)})(U_0G_{\mathrm{N}}^{(n)})^{m_3}D_0, \end{aligned}$$

and we characterise $\mathcal{R}_n$ as follows: starting from some arbitrary level k, a trajectory consists of

- two jumps from level k to level $k+1$ which are followed by a return to level k through trajectories in $\mathcal{G}_{\mathrm{N}}^{(n)}$ but not in $\mathcal{G}_{\mathrm{N}}^{(n-1)}$,
- before each of these two jumps, there may be an arbitrary number of jumps from k to $k+1$ followed by a trajectory in $\mathcal{G}_{\mathrm{N}}^{(n-1)}$,
- after these mandatory two jumps, excursions to higher levels are constrained by $\mathcal{G}_{\mathrm{N}}^{(n)}$ only.

For $n = 1$, as $\mathcal{G}_{\mathrm{N}}^{(0)}$ is empty, this simplifies to the fact that $\mathcal{R}_1$ is formed of trajectories with at least two excursions to higher levels.

Using our set notation, we write (32) as

$$\mathcal{R}_n = \mathcal{U}_{n-1} \,|\, (\mathcal{G}_{\mathrm{N}}^{(n)} \setminus \mathcal{G}_{\mathrm{N}}^{(n-1)}) \,|\, \mathcal{U}_{n-1} \,|\, (\mathcal{G}_{\mathrm{N}}^{(n)} \setminus \mathcal{G}_{\mathrm{N}}^{(n-1)}) \,|\, \mathcal{D}_n.$$

Lemma 3.4. *We can express the set* $\mathcal{G}_{\mathrm{N}}^{(n+1)}$ *as a union of disjoint sets of sample paths via the expression*

$$\mathcal{G}_{\mathrm{N}}^{(n+1)} = \mathcal{D}_n \bigcup_{\ell \geq 1} \mathcal{U}_n^{*\ell} \,|\, \mathcal{R}_n \,|\, \mathcal{D}_n^{*\ell}. \tag{36}$$

Proof. We focus on the set $\mathcal{U}_n^{*\nu} \,|\, \mathcal{D}_n \,|\, \mathcal{D}_n^{*\nu}$ for a fixed value of ν. By (35), it is the union

$$\mathcal{U}_n^{*\nu} \,|\, \mathcal{D}_n \,|\, \mathcal{D}_n^{*\nu} = \mathcal{U}_n^{*\nu} \,|\, \mathcal{G}_{\mathrm{N}}^{(n)} \,|\, \mathcal{D}_n^{*\nu} \cup \mathcal{U}_n^{*\nu} \,|\, \mathcal{R}_n \,|\, \mathcal{D}_n^{*\nu}$$

of two distinct sets of trajectories. Furthermore,

$$\begin{aligned}
\mathcal{U}_n^{*\nu} \,|\, \mathcal{G}_{\mathrm{N}}^{(n)} \,|\, \mathcal{D}_n^{*\nu} &= \mathcal{U}_n^{*(\nu-1)} \,|\, \mathcal{U}_n \,|\, \mathcal{G}_{\mathrm{N}}^{(n)} \,|\, \mathcal{D}_n \,|\, \mathcal{D}_n^{*(\nu-1)} \\
&= \bigcup_{m1, m_2 \geq 0} \mathcal{U}_n^{*(\nu-1)} \,|\, (\mathcal{U}_0 \,|\, \mathcal{G}_{\mathrm{N}}^{(n)})^{*m_1} \,|\, \mathcal{U}_0 \,|\, \mathcal{G}_{\mathrm{N}}^{(n)} \,|\, (\mathcal{U}_0 \,|\, \mathcal{G}_{\mathrm{N}}^{(n)})^{*m_2} \,|\, \mathcal{D}_0 \,|\, \mathcal{D}_n^{*(\nu-1)} \\
&= \bigcup_{m \geq 1} \mathcal{U}_n^{*(\nu-1)} \,|\, (\mathcal{U}_0 \,|\, \mathcal{G}_{\mathrm{N}}^{(n)})^{*m} \,|\, \mathcal{D}_0 \,|\, \mathcal{D}_n^{*(\nu-1)} \\
&\subset \mathcal{U}_n^{*(\nu-1)} \,|\, \mathcal{D}_n \,|\, \mathcal{D}_n^{*(\nu-1)}.
\end{aligned}$$

As a consequence,

$$\begin{aligned}
\mathcal{D}_n \bigcup_{1 \leq \ell \leq \nu} \mathcal{U}_n^{*\ell} \,|\, \mathcal{D}_n \,|\, \mathcal{D}_n^{*\ell} &= \mathcal{D}_n \cup \mathcal{U}_n^{*\nu} \,|\, \mathcal{R}_n \,|\, \mathcal{D}_n^{*\nu} \cup \left(\bigcup_{1 \leq \ell \leq \nu-1} \mathcal{U}_n^{*\ell} \,|\, \mathcal{D}_n \,|\, \mathcal{D}_n^{*\ell} \right) \\
&= \mathcal{D}_n \bigcup_{1 \leq \ell \leq \nu} \mathcal{U}_n^{*\ell} \,|\, \mathcal{R}_n \,|\, \mathcal{D}_n^{*\ell}.
\end{aligned}$$

In view of equation (28), since ν is arbitrary, this proves the lemma.

This, together with Lemma 3.3 and Equation (31), concludes the proof of Theorem 3.2. As an illustration, we give two trajectories of $\mathcal{G}_{\mathrm{N}}^{(2)}$ in Figure 3. The only difference lies between the epochs τ_4 and θ_3. The path on top belongs to $\mathcal{U}_2^{*3} \,|\, \mathcal{R}_2 \,|\, \mathcal{D}_2^{*3}$ and the one underneath belongs to $\mathcal{U}_2^{*4} \,|\, \mathcal{R}_2 \,|\, \mathcal{D}_2^{*4}$.

4. Some Numerical Experience

We implemented in Matlab the Logarithmic-Reduction algorithm, the Cyclic Reduction algorithm and Newton's method as described in the Appendix at 6.1, 6.2 and 6.3 respectively and applied it to some specific examples of QBDs. Our three examples were all initially defined in continuous-time. We uniformized them to derive discrete-time QBDs which are directly amenable to the analysis discussed above.

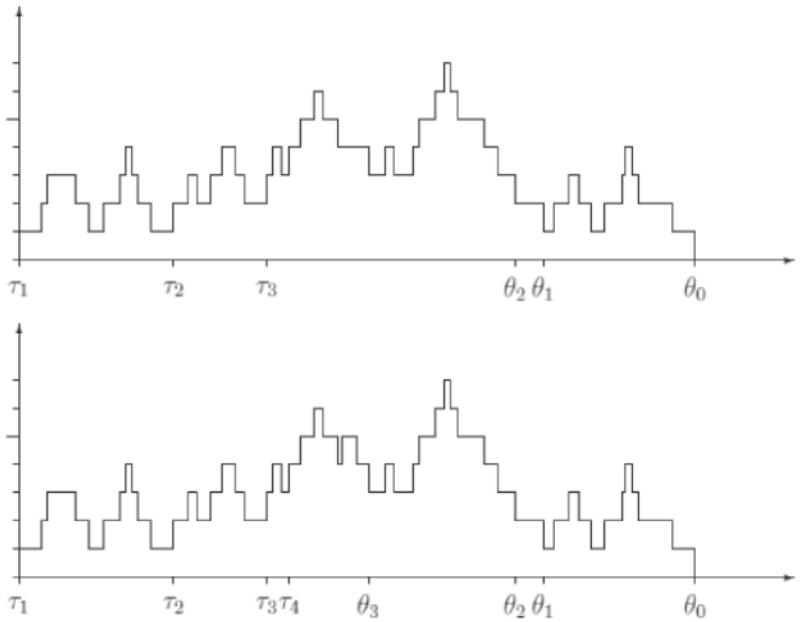

Figure 3. Two trajectories that belong to $\mathcal{G}_{\mathrm{N}}^{(2)}$.

4.1. Example 1

For our first example, we considered a small six-phase QBD with

$$A_1 = \begin{bmatrix} 0.05 & 1.0 & 0 & 0 & 0 & 0 \\ 0 & 0.05 & 1.0 & 0 & 0 & 0 \\ 0 & 0 & 0.05 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.05 & 1.0 & 0 \\ 0 & 0 & 0 & 0 & 0.05 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.05 \end{bmatrix},$$

$$A_0 = \begin{bmatrix} -1.21 & 0.1 & 0 & 0 & 0 & 0 \\ 0.1 & -1.31 & 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & -3.31 & 0.1 & 0 & 0 \\ 0 & 0 & 0.1 & -1.31 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 & -3.31 & 0.1 \\ 0 & 0 & 0 & 0 & 0.1 & -3.21 \end{bmatrix}$$

and

$$A_{-1} = \begin{bmatrix} 0.06 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.06 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.06 & 3.0 & 0 & 0 \\ 0 & 0 & 0 & 0.06 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.06 & 3.0 \\ 3.0 & 0 & 0 & 0 & 0 & 0.06 \end{bmatrix}.$$

We set the value of ε in 6.1, 6.2 and 6.3 to 10^{-10}. All three methods produced the matrix

$$G = \begin{bmatrix} 0.7831 & 0.0148 & 0.0015 & 0.1084 & 0.0015 & 0.0905 \\ 0.6538 & 0.0492 & 0.0029 & 0.1889 & 0.0018 & 0.1033 \\ 0.0532 & 0.0015 & 0.0180 & 0.9180 & 0.0001 & 0.0087 \\ 0.7426 & 0.0014 & 0.0015 & 0.1270 & 0.0022 & 0.1252 \\ 0.0650 & 0.0001 & 0.0000 & 0.0040 & 0.0182 & 0.9126 \\ 0.9489 & 0.0002 & 0.0000 & 0.0017 & 0.0006 & 0.0485 \end{bmatrix}$$

to this precision, even though it is displayed above with entries given only to four decimal places. The logarithmic reduction, cyclic reduction and Newton's method took 7, 8 and 8 iterations respectively to achieve this precision. Despite the number of iterations being consistent, the computer time used for each algorithm on a Dell Precision 5520 varied across different repetitions of the program: the logarithmic reduction algorithm ran for between 0.0048 and 0.0075 seconds, the cyclic reduction algorithm ran for between 0.0037 and 0.0051 seconds, while Newton's algorithm took between 0.0045 and 0.0080 seconds.

4.2. Example 2

Our second example was taken from Latouche and Ramaswami [10] (p. 208). It has

$$A_1 = 0.9\,\mathrm{diag}\begin{bmatrix} 0.2 & 0.2 & 0.2 & 0.2 & 13 & 1 & 1 & 0.2 \end{bmatrix} \tag{37}$$

$$A_{-1} = 2I \tag{38}$$

and

$$A_0 = S - A_1 - A_{-1} \tag{39}$$

where

$$S = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}.$$

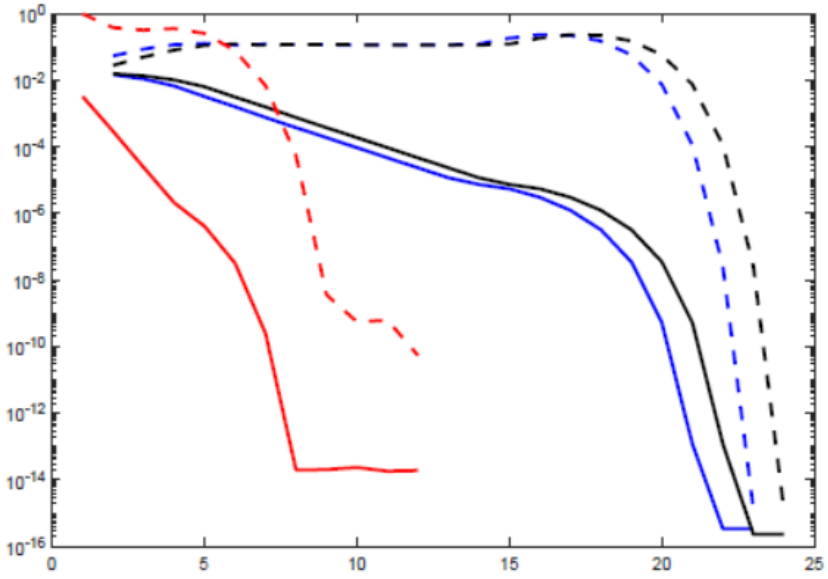

Figure 4. The precision of the three algorithms plotted against the number of iterations (logarithmic reduction in blue, cyclic reduction in black, Newton in red).

Figure 4 plots two indicators of the precision achieved by the three algorithms against the number of iterations used, the logarithmic reduction algorithm is depicted in blue, the cyclic reduction algorithm in black and Newton's method in red. For each algorithm, the continuous line depicts the residue $\|G_n - (A_{-1} + A_0 G_n + A_1 G_n^2)\|_\infty$ where G_n is the matrix obtained at the n th iteration of the respective algorithm. The dotted line is $\|G_n - G_{n-1}\|_\infty$. We see that Newton's method achieves better precision for the same number of iterations than the other two algorithms. However, the time taken by Newton's method to achieve a given precision was generally comparable.

4.3. Example 3

Our third example illustrates an interesting numerical phenomenon that we noticed when coding up the three algorithms. It involves a $2N$ phase QBD with N by N blocks given by

$$A_1 = \begin{bmatrix} \lambda I & 0 \\ 0 & \lambda/(2N) I \end{bmatrix},$$

and

$$A_{-1} = \begin{bmatrix} 0 & 0 \\ 0 & \mu I \end{bmatrix},$$

and where A_0 has entries of γ on the upper and lower diagonal, zeros in all the other off-diagonal entries and its diagonal entries adjusted so that the row sums of $A_1 + A_0 + A_{-1}$ are

zero. We took $N = 50$, $\lambda = 1.5$, $\mu = 2.5$ and $\gamma = 0.05$.

When the value of ε was set to 10^{-10}, the logarithmic reduction, cyclic reduction and Newton's algorithms took 23, 24 and 12 iterations respectively, and typical durations were around 0.03, 0.02 and 0.08 seconds respectively, but were also variable across different runs. When ε was reset to 10^{-11}, the logarithmic reduction and cyclic reduction algorithms worked in the same way, but Newton's algorithm took 21 iterations and its processing time increased by about 70%. With the tolerance set to 10^{-12}, the number of iterations used by the logarithmic reduction and cyclic reduction still remained the same, but the number of iterations used by Newton's method increased further to 25, with a corresponding increase in the processing time. We conclude that, at high precision, Newton's method seems to be affected by numerical factors more than the other two algorithms.

5. Conclusions

In this paper we have discussed physical interpretations of various numerical procedures for evaluating the matrix G in a quasi-birth-and-death process. While the physical interpretations of the linear, logarithmic reduction and cyclic reduction algorithms arise from restricting sample paths from rising too high, the restriction on the sample paths of Newton's algorithm is different: it involves restrictions on the complexity of the sample paths.

We expect similar results to hold for general $M/G/1$-type Markov chains, but the analysis is likely to be more complex still.

Appendix:

6. Algorithms

Algorithm 6.1. (Logarithmic reduction for positive recurrent QBDs)

Input: The positive integer m and the $m \times m$ matrices A_{-1}, A_0, A_1, defining a QBD; a real $\varepsilon > 0$.

Output: An approximation of the minimal nonnegative solution of (6).

Computation:

1. Set $V_1 = (I - A_0)^{-1} A_1$, $V_{-1} = (I - A_0)^{-1} A_{-1}$, $G_{\mathrm{LR}} = V_{-1}$, $T = V_1$.
2. Compute

$$V_0 = V_1 V_{-1} + V_{-1} V_1,$$

$$V_1 = (I - V_0)^{-1} V_1^2, \quad V_{-1} = (I - V_0)^{-1} V_{-1}^2,$$

$$G_{\mathrm{LR}} = G_{\mathrm{LR}} + T V_{-1},$$

$$T = T V_1$$

3. If $\|\mathbf{1}-G_{\mathrm{LR}}\mathbf{1}\|_\infty > \varepsilon$, then repeat from step 2, else move to step 4
4. Output G_{LR} and stop.

Work count: One matrix inversion and 8 matrix products per iteration for a rough total of. $18m^3$ floating point operations.

Algorithm 6.2. (Cyclic reduction for positive recurrent QBDs)

Input: The positive integer m and the $m \times m$ matrices A_{-1}, A_0, A_1, defining a QBD; a real $\varepsilon > 0$.

Output: An approximation of the minimal nonnegative solution of (6).

Computation:

1. Set $V_i = A_i$, $i = -1,0,1$, $\widehat{V} = A_0$.
2. Compute

$$X = (I - V_0)^{-1}V_1, \quad Y = (I - V_0)^{-1}V_{-1},$$

$$V_1' = V_1X, \quad V_{-1}' = V_{-1}Y,$$

$$W = V_1Y, \quad \widehat{V}' = \widehat{V} + W,$$

$$V_0' = V_0 + W + V_{-1}X,$$

and set $V_i = V_i'$, $i = -1,0,1$, $\widehat{V} = \widehat{V}'$.

3. If $\|V_1\|_\infty > \varepsilon$, then repeat from step 2, else move to step 4
4. Output $G_{\mathrm{CR}} = (I - \widehat{V})^{-1}A_{-1}$ and stop.

Work count: One matrix inversion and 6 matrix products per iteration for a rough total of $14m^3$ floating point operations.

Algorithm 6.3. (Newton iteration for positive recurrent QBDs)

Input: The positive integer m and the $m \times m$ matrices A_{-1}, A_0, A_1, defining a QBD; a real $\varepsilon > 0$.

Output: An approximation of the minimal nonnegative solution of (6).

Computation:

1. Set $V_1 = (I - A_0)^{-1}A_1$, $V_{-1} = (I - A_0)^{-1}A_{-1}$, $G_{\mathrm{N}} = 0$.
2. Compute

$$U_1 = (I - V_1G_{\mathrm{N}})^{-1}V_1, \quad U_{-1} = (I - V_1G_{\mathrm{N}})^{-1}V_{-1},$$

$$C = U_{-1} - U_1G_{\mathrm{N}}U_{-1},$$

$$\textit{Solve} \quad X - U_1XU_{-1} = C,$$

and set $G_N = X$.

3. If $\|\mathbf{1} - G_N\mathbf{1}\|_\infty > \varepsilon$, then repeat from step 2, else move to step 4
4. Output G_N and stop.

Work count: One matrix inversion, 4 matrix products and the resolution of the Sylvester equation, for a rough total of $65m^3$ floating point operations per iteration.

Acknowledgment

Nigel Bean and Peter Taylor's research is supported by the Australian Research Council (ARC) Centre of Excellence for the Mathematical and Statistical Frontiers (ACEMS). Peter Taylor's research is also supported by ARC Laureate Fellowship FL130100039. All authors would like to thank an anonymous referee for helpful suggestions that improved the manuscript.

References

[1] Bean, N., Kontoleon, N., & Taylor, P. (2008). Markovian trees: properties and algorithms. *Annals of Operations Research*, 10, 31–50.

[2] Bean, N., O'Reilly, M, & Taylor, P. (2005). Algorithms for return probabilities for stochastic fluid flows. *Stochastic Models*, 21, 149–184

[3] Bini, D. A., Latouche, G., & Meini, B. (2005). Numerical Methods for Structured Markov Chains. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford.

[4] Bini, D. A., & Meini, B. (1995). On Cyclic Reduction Applied to a Class of Toeplitz-like Matrices Arising in Queueing Problems. In W. J. Stewart, editor, *Computations With Markov Chains*, 21–38. Kluwer Academic Publishers, Boston, MA.

[5] Bini, D. A., Meini, B., Steffé, S., & Houdt, B. V. (2006). Structured Markov chains solver: software tools. In *SMCtools '06: Proceeding from the 2006 workshop on Tools for solving structured Markov chains*, 14, New York, NY, USA, 2006. ACM.

[6] Gardiner, J. D., Laub, A. J., Amato, J. J., & Moler, C. B. (1992). Solution of the Sylvester matrix equation $AXB^t + CDX^t = E$. *ACM Transactions on Mathematical Software*, 18, 223–231.

[7] Latouche, G. (1993). Algorithms for infinite Markov chains with repeating columns. In Meyer, C. D. and Plemmons, R. J. editors, *Linear Algebra, Markov Chains and Queueing Models*, 231–265. Springer-Verlag, New York.

[8] Latouche, G. (1994). Newton's iteration for nonlinear equations in Markov chains. *IMA Journal on Numerical Analysis*, 14, 583–598.

[9] Latouche, G., & Nguyen, G. (2018). Analysis of fluid flow models. *Queueing Models and Service Management*, to appear.

[10] Latouche, G., & Ramaswami, V. (1999). Introduction to Matrix Analytic Methods in Stochastic Modeling. ASA-SIAM Series on Statistics and Applied Probability. SIAM, Philadelphia PA.

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