Solving QBD problems: the cyclic reduction algorithm versus the invariant subspace method

Beatrice Meini
Dipartimento di Matematica
Università di Pisa - Italy
meini@dm.unipi.it

Abstract

The problem of the computation of the rate matrix G associated with discrete-time QBD Markov chains is analyzed. We present theoretical and numerical comparisons between the cyclic reduction algorithm of [7, 2, 3, 4] and the method based on invariant subspaces of [1].

1 Introduction

The block tridiagonal and block Toeplitz structure of the probability transition matrix P associated with QBD problems, i.e.,

$$P = \begin{bmatrix} B_0 & B_1 & & 0 \\ A_0 & A_1 & A_2 & \\ & A_0 & A_1 & \ddots \\ 0 & & \ddots & \ddots \end{bmatrix},$$

allowed the design of fast and reliable methods for the computation of the rate matrix G. Here, A_0, A_1, A_2, B_0, B_1 are $k \times k$ nonnegative matrices such that $A_0+A_1+A_2$ is irreducible, and P is irreducible, stochastic and positive recurrent [8, 9]. These fast methods rely on the property that the matrix G, which solves the nonlinear matrix equation

$$G = A_0 + A_1 G + A_2 G^2, \tag{1}$$

can be computed by solving the infinite block tridiagonal, block Toeplitz system

$$\begin{bmatrix} I-A_1 & -A_2 & & 0 \\ -A_0 & I-A_1 & -A_2 & \\ & -A_0 & I-A_1 & \ddots \\ 0 & & \ddots & \ddots \end{bmatrix} \begin{bmatrix} G \\ G^2 \\ G^3 \\ \vdots \end{bmatrix} = \begin{bmatrix} A_0 \\ 0 \\ 0 \\ \vdots \end{bmatrix}. \tag{2}$$

The pioneering method that exploits the Toeplitz structure of the above matrix is the one devised by Latouche and Ramaswami in [7]. In this paper the authors apply a cyclic reduction based method to solve system (2) and derive a quadratically convergent and numerically stable algorithm for the computation of the matrix G. More recently, in [2, 3, 4], Bini and Meini have devised a new quadratically convergent and numerically stable algorithm for the computation of G, based on a functional representation of cyclic reduction, which applies to general M/G/1 type Markov chains [9] and which, in the QBD case, reduces to the method of Latouche and Ramaswami.

A different approach, which apply to M/G/1 type Markov chains having rational generating function (thus in particular to QBD problems), is proposed by Akar and Sohraby in [1]. Here the authors compute the matrix G by finding, by means of matrix sign function iterations, an invariant subspace of a suitable matrix.

The aim of this paper is to compare the cyclic reduction based algorithm of [7, 2, 3, 4] with the approach of [1], in terms of computational cost, numerical stability and convergence rate. Several numerical results show differences in the performance of the two approaches. The cyclic reduction algorithm appears to be more accurate, more robust and faster than the invariant subspace approach.

2 Overview of the two algorithms

Let us first recall the method based on cyclic reduction. For more details we refer to the papers [7, 2, 3, 4].

Let us consider system (2). By recursively applying one step of block cyclic reduction, i.e, an odd-even permutation of block rows and block columns, followed by one step of Gaussian elimination, we generate the sequence of infinite block tridiagonal systems

$$\begin{bmatrix} I-\widehat{A}_1^{(j)} & -A_2^{(j)} & & 0 \\ -A_0^{(j)} & I-A_1^{(j)} & -A_2^{(j)} & \\ & -A_0^{(j)} & I-A_1^{(j)} & \ddots \\ 0 & & \ddots & \ddots \end{bmatrix} \begin{bmatrix} G \\ G^{2^j+1} \\ G^{2\cdot 2^j+1} \\ \vdots \end{bmatrix} = \begin{bmatrix} A_0 \\ 0 \\ 0 \\ \vdots \end{bmatrix}, \quad j \geq 0, \tag{3}$$

where $A_0^{(0)} = A_0$, $\widehat{A}_1^{(0)} = A_1^{(0)} = A_1$, $A_2^{(0)} = A_2$. The blocks $\widehat{A}_1^{(j)}$, $A_i^{(j)}$, $i = 0, 1, 2$, $j \geq 1$, are defined by the recurrences:

$$\begin{aligned} A_0^{(j+1)} &= A_0^{(j)} \left(I - A_1^{(j)}\right)^{-1} A_0^{(j)} \\ A_1^{(j+1)} &= A_1^{(j)} + A_0^{(j)} \left(I - A_1^{(j)}\right)^{-1} A_2^{(j)} + A_2^{(j)} \left(I - A_1^{(j)}\right)^{-1} A_0^{(j)} \\ A_2^{(j+1)} &= A_2^{(j)} \left(I - A_1^{(j)}\right)^{-1} A_2^{(j)} \\ \widehat{A}_1^{(j+1)} &= \widehat{A}_1^{(j)} + A_2^{(j)} \left(I - A_1^{(j)}\right)^{-1} A_0^{(j)}, \end{aligned} \tag{4}$$

and are such that $A_0^{(j)} + A_1^{(j)} + A_2^{(j)}$ and $A_0 + \widehat{A}_1^{(j)} + A_2^{(j)}$ are stochastic. On the other hand, it is proved (see [7, 3]) that the sequence $A_2^{(j)}$, $j \geq 0$, converges in norm to zero as σ^{2^j}, where $0 < \sigma < 1$ is given by

$$\sigma = \max\left\{|\lambda| : \det\left(A_2 + \lambda A_1 + \lambda^2 A_0 - \lambda I\right) = 0, |\lambda| < 1\right\}. \tag{5}$$

Thus, from (3), an approximation of the matrix G is given by $\left(I - \widehat{A}_1^{(j)}\right)^{-1} A_0$, for a sufficiently large value of j.

Let $\epsilon > 0$ be a fixed error bound and denote with $\boldsymbol{e}$ the k-dimensional vector having all the entries equal to 1. The stopping criterion is the stochasticity condition of $\left(I - \widehat{A}_1^{(j)}\right)^{-1} A_0$, i.e.,

$$\left(I - \left(I - \widehat{A}_1^{(j)}\right)^{-1} A_0\right) \boldsymbol{e} < \epsilon \boldsymbol{e}. \tag{6}$$

Before describing the method presented in [1], we recall the definition of matrix sign and the definition of left invariant subspace:

Definition 1 *Let M be an $m \times m$ real matrix, with no pure imaginary eigenvalues. Let $M = S(D+N)S^{-1}$ be the Jordan decomposition of M, where $D = \mathrm{Diag}(\lambda_1, \lambda_2, \ldots, \lambda_m)$ and N is nilpotent and commutes with D. Then the matrix sign of M is given by*

$$Z = sgn(M) = S\,\mathrm{Diag}(sgn(\lambda_1), sgn(\lambda_2), \ldots, sgn(\lambda_m))S^{-1},$$

where, for any complex number z with $\mathrm{re}\, z \neq 0$,

$$sgn(z) = \begin{cases} 1 & \text{if } \mathrm{re}\, z > 0 \\ -1 & \text{if } \mathrm{re}\, z < 0. \end{cases}$$

Definition 2 *Let A be an $m \times m$ real matrix. Let $\mathcal{S}$ be a k-dimensional subspace of $\boldsymbol{R}^m$ such that $A\boldsymbol{x} \in \mathcal{S}$, for any $\boldsymbol{x} \in \mathcal{S}$. Let S be an $m \times k$ matrix, whose columns are a basis of $\mathcal{S}$. Denote with A_1 a $k \times k$ matrix such that $AS = SA_1$. Then the subspace $\mathcal{S}$ is called the (closed) left invariant subspace of A if the eigenvalues of A_1 are contained in the (closed) left half-plane of the complex plane, and there is no larger subspace for which this inclusion holds.*

Let us now recall the approach proposed in [1], here adapted to the case of QBD problems. For more details we refer to [1].

Define the matrix polynomial

$$\begin{array}{l} H(z) = \sum_{i=0}^{2} H_i z^i = (1-z)^2 (tI - A_0 - A_1 t - A_2 t^2)|_{t=\frac{1+z}{1-z}} = \\ -A_0(1-z)^2 + (I - A_1)(1-z)(1+z) - A_2(1+z)^2, \end{array}$$

such that $\det H(z) = 0$ has exactly $k-1$ roots with imaginary part less than zero, and a simple root at $z = 0$.

Define the matrices $\widehat{H}_0 = H_2^{-1} H_0$, $\widehat{H}_1 = H_2^{-1} H_1$ and consider the 2×2 block companion matrix

$$E = \begin{bmatrix} 0 & I \\ -\widehat{H}_0 & -\widehat{H}_1 \end{bmatrix}.$$

Let T be a $2k \times k$ matrix whose columns are a basis of the closed left invariant subspace of the matrix E, and partition the matrix T as $T = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$, where T_1, T_2 are $k \times k$ matrices. Then the matrix G is given by

$$G = (T_1 + T_2)(T_1 - T_2)^{-1}. \tag{7}$$

The authors observe that the closed left invariant subspace of E coincides with the left invariant subspace of the matrix

$$\widehat{E} = E - \frac{\boldsymbol{y}\boldsymbol{x}^T}{\boldsymbol{x}^T\boldsymbol{y}},$$

where

$$\boldsymbol{x}^T = \begin{bmatrix} \boldsymbol{x}_0^T \widehat{H}_1 & \boldsymbol{x}_0^T \end{bmatrix} \quad \boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{0} \end{bmatrix}$$

and $\boldsymbol{x}_0$, $\boldsymbol{y}_0$ are two vectors such that $\boldsymbol{x}_0^T \widehat{H}_0 = \mathbf{0}$, $\widehat{H}_0 \boldsymbol{y}_0 = \mathbf{0}$. Now, a basis of the left invariant subspace of $\widehat{E}$, forming the columns of the matrix T, is given by k linearly independent columns of $I - Z$, where Z is the matrix sign of $\widehat{E}$. In order to compute the matrix sign Z the authors use the scaled Newton iteration (matrix sign function iteration)

$$\begin{cases} Z_0 = \widehat{E} \\ Z_{n+1} = \frac{1}{2}\left(\gamma_n Z_n + \gamma_n Z_n^{-1}\right), \ \gamma_n = |\det Z_n|^{-1/k}, \quad n \geq 0, \end{cases} \tag{8}$$

with the stopping criterion

$$||Z_{n+1} - Z_n||_1 < \epsilon ||Z_n||_1,$$

for a fixed small error bound ϵ. The above sequence quadratically converges to Z. Once Z is known, the matrix T is computed by means of the rank revealing QR decomposition of the matrix $I - Z$.

3 Theoretical and numerical comparisons

Both the algorithms consist of generating a sequence of matrices that quadratically converge to a limit, which provides the solution G. However the role of the two sequences is quite different.

In the cyclic reduction algorithm, a sequence of nonlinear matrix equations

$$X = A_0 + \widehat{A}_1^{(j)} X + A_2^{(j)} X^{2^j+1}, \ j \geq 0, \tag{9}$$

is generated such that its solution is the matrix G. That is, the initial problem (1) is transformed in the equivalent problems (9). Since the sequence $A_2^{(j)}$, $j \geq 0$, converges to zero, we have

$$G = \left(I - \widehat{A}_1^*\right)^{-1} A_0,$$

where $\widehat{A}_1^* = \lim_j \widehat{A}_1^{(j)}$.

In the invariant subspace approach, the authors prove that the matrix G can be recovered, by means of formula (7), after computing the matrix sign Z of $\widehat{E}$. In order to compute Z the quadratically convergent sequence (8) is generated.

Concerning the speed of convergence of the two algorithms, it is proved that the sequences $A_2^{(j)}$ and $\widehat{A}_1^{(j)}$ of cyclic reduction converge to zero and $\widehat{A}_1^*$, respectively, as σ^{2^j}, where $0 < \sigma < 1$ is defined in (5). For the sequence Z_n of (8), the authors only claim that the convergence is quadratic, and do not relate it with the blocks A_0, A_1, A_2, defining the solution G.

From (4), each step of cyclic reduction requires six $k \times k$ matrix multiplications and one matrix inversion, and thus the computational cost of one step is about $14k^3$ arithmetic operations. A matrix sign function iteration, i.e., the computation of Z_n, requires one inversion of a $2k \times 2k$ matrix, that is performed by means of LU factorization with $16k^3$ arithmetic operations.

Concerning numerical stability, the operations involved in (4) consist of multiplications and additions of nonnegative matrices and inversions of M-matrices [10], i.e., matrices having non-positive off-diagonal entries and nonnegative inverse. These computations, if the diagonal adjustment technique [6] is used, can be reduced to performing additions of positive numbers, multiplications and divisions. This avoids cancellation errors. Moreover, the matrices $\left(I - \widehat{A}_1^{(j)}\right)^{-1} A_0$ form a sequence of sub-stochastic matrices, that monotonically converges to G. If the stopping criterion (6) is satisfied, and if $\widetilde{G} = \left(I - \widehat{A}_1^{(j)}\right)^{-1} A_0$ is the corresponding approximation of G, then, from (6), we have

$$||G - \widetilde{G}||_\infty = ||(G - \widetilde{G})\boldsymbol{e}||_\infty = ||(I - \widetilde{G})\boldsymbol{e}||_\infty < \epsilon.$$

Thus condition (6) is a good stopping criterion for cyclic reduction.

An analysis of the numerical stability of the invariant subspace approach is not presented by the authors. The effect of the round-off errors generated in the computation of the matrices Z_n, in particular in the case where Z_n is close to a singular matrix, is not clear. In practice, it may happen that the sequence Z_n converges to a matrix $\widetilde{Z}$ which is not the matrix sign of $\widehat{E}$. In this case it is not clear what the error of the approximation $\widetilde{G}$ of the matrix G would be. In the extensive numerical results presented in [1], the authors report the error of the matrix $\widetilde{G}$ as $||\boldsymbol{e} - \widetilde{G}\boldsymbol{e}||_\infty$. This error does not give any information on the accuracy of the computed solution, unless $\widetilde{G}$ is sub-stochastic. Note that according to this definition of error, the identity matrix would be an approximation of the solution of any matrix equation (1) with *error zero*.

We have tested the cyclic reduction algorithm and the invariant subspace method on some examples. The cyclic reduction algorithm for QBD problems has been implemented in Fortran90 (the code can be obtained at the URL http://www.dm.unipi.it/pages/meini/public_html). For the invariant subspace method we have used the C implementation provided by the authors of [1] in the TELPACK package. This package can be taken at the URL http://www.cstp.umkc.edu/org/tn/telpack/home.html). The programs have run on an alpha workstation, with the IEEE standard double precision arithmetic. For the stopping conditions of both the algorithms we have chosen $\epsilon = 10^{-12}$. In the tables of results we have reported, for the two algorithms, the CPU time (in seconds) needed to compute an approximation $\widetilde{G}$ of the matrix G, the residual

	Cyclic Reduction			Invariant Subspace		
p	Iter.	Time	Error	Iter.	Time	Error
10^{-1}	8	0.00	$4.4 \cdot 10^{-15}$	7	0.00	$6.6 \cdot 10^{-16}$
10^{-2}	12	0.00	0.0	8	0.00	$8.0 \cdot 10^{-16}$
10^{-3}	15	0.00	0.0	8	0.00	$1.5 \cdot 10^{-15}$
10^{-4}	19	0.00	0.0	9	0.00	$2.2 \cdot 10^{-16}$
10^{-5}	22	0.00	0.0	9	0.00	0.0
10^{-6}	25	0.00	0.0	9	0.00	$6.7 \cdot 10^{-16}$
10^{-7}	29	0.00	0.0	10	0.02	$9.4 \cdot 10^{-16}$
10^{-8}	32	0.00	0.0	10	0.02	$1.3 \cdot 10^{-15}$
10^{-9}	35	0.00	0.0	10	0.00	$1.8 \cdot 10^{-15}$
10^{-10}	38	0.00	0.0	10	0.00	$1.9 \cdot 10^{-16}$

Table 1: Example 1

error of the computed approximation (as $err = ||\widetilde{G} - A_0 - A_1\widetilde{G} - A_2\widetilde{G}^2||_\infty$) and the number of iterations.

Example 1 The first problem we have tested is Example 1 of [1]. Here the blocks, having dimension 2, depend on a parameter $p > 0$, and are defined as

$$A_0 = \begin{bmatrix} 1-p & 0 \\ 0 & 0 \end{bmatrix}, \; A_1 = \begin{bmatrix} 0 & p \\ 2p & 0 \end{bmatrix}, \; A_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1-2p \end{bmatrix}.$$

The authors of [1] claim that their method "appears to be slightly faster than the logarithmic reduction algorithm of Latouche and Ramaswami" and that the accuracy of the latter algorithm "appears to deteriorate as the parameter p decreases". We have tested the same values of the parameter p of [1]. From Table 1 it follows that the performances of both the algorithms are insensitive to the parameter p. The execution times and the residual errors are negligible, for any value of p, for the cyclic reduction algorithm and for the invariant subspace method.

Example 2 The problem we now consider, already tested in [1], is the teletraffic system of [5], modeled as a continuous time QBD process, whose infinitesimal generator is defined by the 24×24 blocks

$$(A'_0)_{i,j} = \begin{cases} 192\rho_d & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}, \quad (A'_2)_{i,j} = \begin{cases} 192(1 - i/24) & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases},$$

$$(A'_1)_{i,j} = \begin{cases} ar(M-i)/M & \text{if } i - j = -1 \\ ir & \text{if } i - j = 1 \\ 0 & \text{elsewhere} \end{cases}, \quad i, j = 0, 1, \ldots, 23.$$

Here a, r, M, ρ_d are parameters of the teletraffic model. We reduce the above process to a discrete-time QBD Markov chain, defined by the 24×24 blocks A_0, A_1, A_2, and we apply the algorithms for the computation of the matrix G. Tables 2 and 3 report the results obtained with the parameters tested in [1]. From Table 2 we observe that the cyclic reduction outperforms the invariant

	Cyclic Reduction			Invariant Subspace		
M	Iter.	Time	Error	Iter.	Time	Error
64	18	0.09	$7.8 \cdot 10^{-16}$	9	0.23	$3.0 \cdot 10^{-15}$
128	19	0.09	$9.9 \cdot 10^{-16}$	10	0.23	$1.6 \cdot 10^{-15}$
256	20	0.10	$1.0 \cdot 10^{-15}$	11	0.23	$4.0 \cdot 10^{-15}$
512	21	0.10	$8.0 \cdot 10^{-16}$	12	0.27	$1.4 \cdot 10^{-14}$
1024	22	0.11	$7.9 \cdot 10^{-16}$	13	0.28	$4.5 \cdot 10^{-14}$
2048	23	0.11	$8.9 \cdot 10^{-16}$	13	0.28	$1.8 \cdot 10^{-13}$
4096	24	0.12	$8.9 \cdot 10^{-16}$	14	0.30	$6.7 \cdot 10^{-13}$
8192	25	0.12	$9.1 \cdot 10^{-16}$	15	0.32	$2.9 \cdot 10^{-12}$
16384	26	0.13	$8.9 \cdot 10^{-16}$	16	0.35	$2.5 \cdot 10^{-11}$
32768	28	0.14	$7.8 \cdot 10^{-16}$	43	0.85	$2.1 \cdot 10^{-10}$
65536	33	0.16	$8.9 \cdot 10^{-16}$	*	*	*

Table 2: Example 2: $r = 1/300$, $a = 18.244$, $\rho_d = 0.280$

subspace method, both for the execution time, and for the accuracy of the result. Concerning the accuracy, we observe that cyclic reduction is insensitive to the parameter M, while the results obtained with the invariant subspace approach deteriorate, as M grows. In the case $M = 65536$ the TELPACK program has an abnormal termination, with the segmentation fault signal. By increasing to $\epsilon = 10^{-8}$ the threshold value for the stopping condition, the program terminates after 133 iterations, in 258 seconds, with the residual error $7.0 \cdot 10^{-7}$.

Analogous results are obtained in Table 3, where r, a and M are fixed. Also in this case the cyclic reduction outperforms the invariant subspace approach.

Indeed, the accuracy of the results of the latter method deteriorates, as ρ_d grows, while cyclic reduction seems more robust. Moreover, in the case $\rho_d = 0.29568$, which is close to $\rho_d = 0.296$, that yields an unstable queuing system, the TELPACK program has an abnormal termination, with the segmentation fault signal. By increasing to $\epsilon = 10^{-8}$ the threshold value for the stopping condition, the program terminates after 83 iterations, in 1.63 seconds, with the residual error $3.6 \cdot 10^{-8}$.

	Cyclic Reduction			Invariant Subspace		
ρ_d	Iter.	Time	Error	Iter.	Time	Error
0.01	5	0.03	$8.9 \cdot 10^{-16}$	10	0.25	$5.3 \cdot 10^{-16}$
0.025	6	0.03	$9.5 \cdot 10^{-16}$	10	0.23	$8.4 \cdot 10^{-16}$
0.05	9	0.05	$8.9 \cdot 10^{-16}$	10	0.23	$3.0 \cdot 10^{-15}$
0.075	11	0.06	$1.2 \cdot 10^{-15}$	10	0.23	$1.7 \cdot 10^{-15}$
0.1	13	0.07	$9.0 \cdot 10^{-16}$	10	0.23	$1.3 \cdot 10^{-15}$
0.12	13	0.06	$1.0 \cdot 10^{-15}$	10	0.23	$6.5 \cdot 10^{-16}$
0.14	14	0.07	$7.8 \cdot 10^{-16}$	10	0.23	$1.5 \cdot 10^{-15}$
0.16	15	0.07	$9.0 \cdot 10^{-16}$	9	0.20	$9.0 \cdot 10^{-16}$
0.18	15	0.07	$1.0 \cdot 10^{-15}$	9	0.22	$8.5 \cdot 10^{-16}$
0.2	16	0.08	$8.9 \cdot 10^{-16}$	9	0.20	$1.0 \cdot 10^{-15}$
0.22	17	0.08	$8.1 \cdot 10^{-16}$	9	0.22	$3.0 \cdot 10^{-15}$
0.24	17	0.08	$1.0 \cdot 10^{-15}$	10	0.25	$2.4 \cdot 10^{-15}$
0.26	18	0.09	$8.9 \cdot 10^{-16}$	10	0.23	$4.0 \cdot 10^{-15}$
0.28	19	0.09	$8.1 \cdot 10^{-16}$	12	0.27	$2.8 \cdot 10^{-14}$
0.29	21	0.10	$6.7 \cdot 10^{-16}$	13	0.28	$1.0 \cdot 10^{-13}$
0.29568	30	0.12	$6.8 \cdot 10^{-16}$	*	*	*

Table 3: Example 2: $r = 1/100$, $a = 18.244$, $M = 512$

Example 3 In this example we construct a QBD problem defined by the $k \times k$ blocks $A_0 = R + \delta I$, $A_1 = A_2 = R$, where R is a matrix having null diagonal entries and constant off-diagonal entries, and $0 < \delta < 1$. We observe that, in this case, the rate $\rho = \boldsymbol{a}^T(A_1 + 2A_2)\boldsymbol{e}$, where $\boldsymbol{a}^T(A_0 + A_1 + A_2) = \boldsymbol{a}^T$, $\boldsymbol{a}^T\boldsymbol{e} = 1$, is exactly $\rho = 1 - \delta$. Thus, as δ approaches zero, the Markov chain becomes more unstable. Here we test the two algorithms for different values of the parameter δ. Tables 4, 5, 6 report the results obtained with size $k = 16$, $k = 32$, $k = 64$. In the last column of each table we also report the closeness to stochasticity of the matrix $\widetilde{G}$ obtained with the invariant subspace approach, i.e., $||\boldsymbol{e} - \widetilde{G}\boldsymbol{e}||_\infty$.

	Cyclic Reduction			Invariant Subspace			
δ	Iter.	Time	Error	Iter.	Time	Error	Stoc.
10^{-1}	7	0.01	$4.8 \cdot 10^{-16}$	8	0.07	$6.1 \cdot 10^{-15}$	$1.3 \cdot 10^{-15}$
10^{-2}	10	0.02	$5.6 \cdot 10^{-16}$	11	0.08	$8.1 \cdot 10^{-14}$	$4.7 \cdot 10^{-15}$
10^{-3}	13	0.02	$5.1 \cdot 10^{-16}$	13	0.10	$8.0 \cdot 10^{-12}$	$2.9 \cdot 10^{-15}$
10^{-4}	16	0.03	$6.5 \cdot 10^{-16}$	16	0.12	$6.7 \cdot 10^{-10}$	$1.3 \cdot 10^{-15}$
10^{-5}	20	0.03	$5.3 \cdot 10^{-16}$	18	0.12	$5.9 \cdot 10^{-8}$	$3.1 \cdot 10^{-15}$
10^{-6}	23	0.04	$7.1 \cdot 10^{-16}$	25	0.15	$4.4 \cdot 10^{-6}$	$3.2 \cdot 10^{-15}$
10^{-7}	26	0.05	$6.7 \cdot 10^{-16}$	69	0.45	$4.3 \cdot 10^{-4}$	$1.8 \cdot 10^{-15}$
10^{-8}	29	0.05	$6.3 \cdot 10^{-16}$	25	0.17	$1.4 \cdot 10^{-2}$	$1.3 \cdot 10^{-15}$

Table 4: Example 3: $k = 16$

	Cyclic Reduction			Invariant Subspace			
δ	Iter.	Time	Error	Iter.	Time	Error	Stoc.
10^{-1}	7	0.08	$9.9 \cdot 10^{-16}$	8	0.45	$7.8 \cdot 10^{-15}$	$3.1 \cdot 10^{-15}$
10^{-2}	10	0.11	$1.2 \cdot 10^{-15}$	11	0.58	$1.1 \cdot 10^{-13}$	$7.8 \cdot 10^{-15}$
10^{-3}	13	0.14	$9.8 \cdot 10^{-16}$	14	0.72	$9.4 \cdot 10^{-12}$	$4.2 \cdot 10^{-15}$
10^{-4}	16	0.19	$8.9 \cdot 10^{-16}$	17	0.85	$1.0 \cdot 10^{-9}$	$4.0 \cdot 10^{-15}$
10^{-5}	20	0.22	$8.5 \cdot 10^{-16}$	24	1.17	$6.6 \cdot 10^{-8}$	$1.7 \cdot 10^{-14}$
10^{-6}	23	0.27	$1.2 \cdot 10^{-15}$	93	4.43	$6.3 \cdot 10^{-6}$	$8.2 \cdot 10^{-15}$
10^{-7}	26	0.30	$1.3 \cdot 10^{-15}$	31	1.52	$8.1 \cdot 10^{-4}$	$2.0 \cdot 10^{-15}$
10^{-8}	29	0.33	$6.2 \cdot 10^{-16}$	55	2.65	$4.9 \cdot 10^{-2}$	$2.4 \cdot 10^{-15}$

Table 5: Example 3: $k = 32$

	Cyclic Reduction			Invariant Subspace			
δ	Iter.	Time	Error	Iter.	Time	Error	Stoc.
10^{-1}	7	0.76	$2.8 \cdot 10^{-15}$	8	3.87	$2.5 \cdot 10^{-14}$	$2.4 \cdot 10^{-14}$
10^{-2}	10	1.07	$1.9 \cdot 10^{-15}$	11	5.05	$2.6 \cdot 10^{-13}$	$2.1 \cdot 10^{-14}$
10^{-3}	13	1.38	$2.5 \cdot 10^{-15}$	14	6.30	$1.7 \cdot 10^{-11}$	$3.5 \cdot 10^{-14}$
10^{-4}	16	1.69	$2.0 \cdot 10^{-15}$	17	7.50	$2.0 \cdot 10^{-8}$	$1.1 \cdot 10^{-14}$
10^{-5}	20	2.10	$2.4 \cdot 10^{-15}$	20	8.70	$1.8 \cdot 10^{-7}$	$7.1 \cdot 10^{-15}$
10^{-6}	23	2.46	$2.3 \cdot 10^{-15}$	59	24.90	$2.3 \cdot 10^{-5}$	$3.5 \cdot 10^{-14}$
10^{-7}	26	2.79	$3.3 \cdot 10^{-15}$	43	18.30	$2.1 \cdot 10^{-3}$	$4.4 \cdot 10^{-11}$
10^{-8}	29	3.04	$1.7 \cdot 10^{-15}$	78	32.70	$1.8 \cdot 10^{-2}$	$5.9 \cdot 10^{-15}$

Table 6: Example 3: $k = 64$

This value is reported by the TELPACK package as an estimate of the accuracy of the output. It is worth pointing out that, even though this value is small, the corresponding approximation error may be large. In fact, for small values of δ, the invariant subspace algorithm delivers an output matrix that, though stochastic, is a very poor approximation of the solution of (1). On the other hand, as shown by tables 4–6, the cyclic reduction algorithm shows its robustness and effectiveness for any value of δ.

References

[1] N. Akar and K. Sohraby. An invariant subspace approach in M/G/1 and G/M/1 type Markov chains. *Commun. Statist. Stochastic Models*, 13:381–416, 1997.

[2] D. Bini and B. Meini. On cyclic reduction applied to a class of Toeplitz-like matrices arising in queueing problems. In W. J. Stewart, editor, *Computations with Markov Chains*, pages 21–38. Kluwer Academic Publisher, 1995.

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

[4] D. Bini and B. Meini. Improved cyclic reduction for solving queueing problems. *Numerical Algorithms*, 15:57–74, 1997.

[5] J. N. Daigle and D. M. Lucantoni. Queueing systems having phase-dependent arrival and service rates. In W. J. Stewart, editor, *Numerical solution of Markov Chains*, pages 161–202. Marcel Dekker, New York, 1991.

[6] W. K. Grassman, M. I. Taksar, and D. P. Heyman. Regenerative analysis and steady state distribution for Markov chains. *Oper. Res.*, 33:1107–1116, 1985.

[7] G. Latouche and V. Ramaswami. A logarithmic reduction algorithm for Quasi-Birth-Death processes. *J. Appl. Probability*, 30:650–674, 1993.

[8] M. F. Neuts. *Matrix-Geometric Solutions in Stochastic Models.* Johns Hopkins University Press, Baltimore, 1981.

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

[10] R. S. Varga. *Matrix Iterative Analysis.* Prentice Hall, New Jersey, 1962.