Electronic Transactions on Numerical Analysis.
Volume 17, pp. 151-167, 2004.

ISSN 1068-9613.

ETNA
Kent State University
etna@mcs.kent.edu

A QUADRATICALLY CONVERGENT BERNOULLI-LIKE ALGORITHM FOR SOLVING MATRIX POLYNOMIAL EQUATIONS IN MARKOV CHAINS*

C. HE †, B. MEINI ‡, N.H. RHEE §, AND K. SOHRABY ¶

Abstract. A quadratically convergent algorithm is developed for solving matrix polynomial equations arising in M/G/1 and G/M/1 type Markov chains. The algorithm is based on the computation of generalized block eigenvalues/vectors of a suitable pair of matrices by means of a Bernoulli-like method. The use of the displacement structure allows one to reduce the computational cost per step. A shifting technique speeds up the rate of convergence.

Key words. polynomial matrix equations, Markov chains, generalized eigenvalues/eigenvectors, displacement structure.

AMS subject classifications. 15A24, 60J22, 65F15.

1. Introduction. We develop a quadratically convergent algorithm for computing the component-wise minimal nonnegative solution of the matrix polynomial equation

$$G = \sum_{i=0}^{n} A_i G^i, \tag{1.1}$$

where A_i, $i = 0, 1, \ldots, n$, are all nonnegative $m \times m$ matrices and $A = \sum_{i=0}^{n} A_i$ is irreducible and stochastic. The computation of G is fundamental in the numerical solution of M/G/1 type Markov chains. In fact, Markov chains of M/G/1 type, introduced by M.F. Neuts in [27], are characterized by block upper Hessenberg transition probability matrices, which are "almost" block Toeplitz. Due to the structure of the probability transition matrix, the computation of the steady state vector, and of other important performance measures, is ultimately reduced to the computation of the matrix G [27].

We also consider the dual problem

$$R = \sum_{i=0}^{n} R^i A_i, \tag{1.2}$$

under the same conditions on A_i. Such a problem arises in G/M/1 type Markov chains [26] having a transition probability matrix in block lower Hessenberg form, which is "almost" block Toeplitz. Also, for this class of Markov chains, the computation of the steady state vector, as well as of other important performance measures, is ultimately reduced to solving (1.2).

We assume that in both problems the associated Markov chain is irreducible and positive recurrent. Under this assumption there exists a componentwise minimal nonnegative solution of (1.1) and (1.2), and these solutions are such that $\rho(G) = 1$, $\rho(R) < 1$, respectively, where the symbol $\rho(\cdot)$ denotes the spectral radius. Our algorithm will compute such minimal solutions.

In the last years, several algorithms for solving the above nonlinear matrix equations have been designed. Besides the classical fixed point iterations [28, 22, 25], quadratically

*Received May 16, 2003. Accepted for publication March 2, 2004. Recommended by Martin Gutknecht.
†Sprint Corporation, Network Planning and Design, 7171 West 95th Street Overland Park, KS 66212, U.S.A.
‡Dipartimento di Matematica, Università di Pisa, via Buonarroti 2, 56127 Pisa, Italy.
§Dept. of Mathematics and Statistics, University of Missouri-Kansas City, Kansas City, MO 64110, U.S.A.
¶Dept. of Computer Science and Telecommunications, University of Missouri-Kansas City, Kansas City, MO 64110, U.S.A. This research was supported by DARPA under Grant F0316.

convergent algorithms have been developed, based on Newton's method [23], on cyclic reduction [24, 7, 8, 6], and on the computation of invariant subspaces [1]. Here we propose a different approach, based on the following observation: by setting

$$\boldsymbol{g} = \begin{bmatrix} I \\ G \\ \vdots \\ G^{n-1} \end{bmatrix}, \tag{1.3}$$

and

$$C = \left[\begin{array}{c|ccccc} 0 & I & 0 & \dots & & 0 \\ \hline 0 & 0 & I & & & 0 \\ \vdots & \vdots & \ddots & \ddots & & \\ 0 & 0 & \dots & 0 & & I \\ -A_0 & I-A_1 & \dots & -A_{n-2} & & -A_{n-1} \end{array}\right], \qquad D = \left[\begin{array}{c|cccc} I & 0 & \dots & \dots & 0 \\ \hline 0 & I & & & 0 \\ \vdots & & \ddots & & \\ \vdots & & & I & \\ 0 & 0 & & & A_n \end{array}\right], \tag{1.4}$$

we find that (1.1) implies

$$C\boldsymbol{g} = D\boldsymbol{g}G. \tag{1.5}$$

As pointed out in [19], equation (1.5) means that G solves (1.1) if and only if the columns of the block vector $\boldsymbol{g}$ span a deflating subspace [29] for the pair (C, D). We can also say that G is a block generalized eigenvector of the pair (C, D), with corresponding block eigenvector $\boldsymbol{g}$.

Observe that C is a block Frobenius matrix, possibly singular. Moreover, also the matrix D could be singular.

Our algorithm provides an approximation of the generalized block eigenvalue G in this way: we generate two sequences of matrices $\{C^{(k)}\}_{k\geq 0}$, $\{D^{(k)}\}_{k\geq 0}$ satisfying

$$C^{(k)}\boldsymbol{g} = D^{(k)}\boldsymbol{g}G^{2^k}, \quad k = 0, 1, \dots .$$

Due to the spectral properties of G, we show that a suitable submatrix of $C^{(k)}$ quadratically converges to zero, as $k \to \infty$. This allows one to compute a finite number of matrices $C^{(k)}$, $D^{(k)}$, $k = 1, \dots, K$, for a suitable K, and then to recover an approximation of G by solving an $m \times m$ linear system. We show that the matrices $D^{(k)}$, $k \geq 0$, are sparse, and that their computation requires only $O(m^3 n)$ arithmetic operations (ops). The structure of the matrices $C^{(k)}$, $k \geq 0$, is less evident, since they are full matrices. However, we show that the block displacement rank (see [20]) of $C^{(k)}$ is at most 3. Thus, the concept of displacement rank allows one to exploit the structure of C, and to represent $C^{(k)}$ by means of a few block vectors. Such vectors can be computed by means of Fast Fourier Transforms, with a computational cost of $O(m^2 n \log n + m^3 n)$ ops.

The resulting algorithm is quadratically convergent, and the computational cost of each step is $O(m^2 n \log n + m^3 n)$ ops. Finally, we increase the speed of convergence by means of the shifting technique introduced in [17].

ETNA
Kent State University
etna@mcs.kent.edu

For the dual matrix equation (1.2) we propose a similar algorithm.

The idea of solving polynomial matrix equations by computing a block eigenvalue/eigenvector, or a deflating subspace, is not new. In [12, 14, 16, 19] a block Bernoulli iteration is applied to compute a block eigenvalue of the block Frobenius matrix associated with the matrix polynomial equation. In the Markov chains framework, the matrix G is approximated by computing the invariant subspace of a suitable block Frobenius matrix [1]. More generally, in [19, 2] the solution of the polynomial matrix equation is expressed in terms of a generalized Schur decomposition of C and D, and a Schur method is applied to compute such decomposition. In particular, in [2] this approach is applied to several classes of polynomial and rational matrix equations; however, the authors write that, for polynomial matrix equations of degree greater than 2, they don't know how the structure of the block Frobenius matrix can be exploited to compute the generalized Schur decomposition.

The paper is organized as follows. In Section 2 we describe our Bernoulli-like algorithm. In Section 3 we analyze the displacement structure of the matrices $C^{(k)}$, $k \geq 0$. In Section 4 the algorithm is adapted for solving (1.2). In Section 5 we propose a shifting technique to speed up the convergence. Finally, in Appendix A we recall the concept of displacement rank and its main properties.

2. The basic algorithm for G. In the following, for a positive integer h, we will denote by I_h the $h \times h$ identity matrix. Moreover, we will denote by $\boldsymbol{e}_1$ the $m(n-1) \times m$ matrix made up by the first m columns of $I_{m(n-1)}$, i.e.,

$$\boldsymbol{e}_1 = [I_m, O, \ldots, O]^{\mathrm{T}}.$$

The matrix $I_m - A_1$ is a nonsingular M-matrix [27], thus we will assume without loss of generality that $I_m - A_1 = I_m$, i.e., $A_1 = 0$. Indeed, in the general case, we may multiply the matrix equation (1.1), on the left, by $(I_m - A_1)^{-1}$.

Let G be the minimal nonnegative solution of the matrix equation (1.1), and let C and D be the $n \times n$ block matrices defined in (1.4) such that

$$Cg = DgG, \tag{2.1}$$

where $\boldsymbol{g}$ is the n-block dimensional vector defined in (1.3). Let us denote by $C_{n-1,n-1}$ the $(n-1) \times (n-1)$ block trailing principal submatrix of C. Since we are assuming $A_1 = 0$, it is a simple matter to verify that $C_{n-1,n-1}$ is nonsingular and that its inverse is

$$C_{n-1,n-1}^{-1} = \begin{bmatrix} A_2 & \ldots & A_{n-1} & I_m \\ I_m & 0 & \ldots & 0 \\ & \ddots & \ddots & \vdots \\ 0 & & I_m & 0 \end{bmatrix}.$$

In particular, by defining

$$S = \begin{bmatrix} I_m & 0 \\ 0 & C_{n-1,n-1}^{-1} \end{bmatrix}, \quad \widehat{C} = SC, \quad \widehat{D} = SD,$$

we obtain, from (2.1), that

$$\widehat{C}\boldsymbol{g} = \widehat{D}\boldsymbol{g}G.$$

ETNA
Kent State University
etna@mcs.kent.edu

The matrices $\widehat{C}$ and $\widehat{D}$ are explicitly given by

$$\widehat{C} = \left[\begin{array}{c|cccc} 0 & I_m & 0 & \dots & 0 \\ \hline -A_0 & I_m & & & 0 \\ 0 & & \ddots & & \\ \vdots & & & \ddots & \\ 0 & 0 & & & I_m \end{array}\right], \quad \widehat{D} = \left[\begin{array}{c|cccc} I_m & 0 & \dots & \dots & 0 \\ \hline 0 & A_2 & A_3 & \dots & A_n \\ 0 & I_m & 0 & \dots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & & I_m & 0 \end{array}\right].$$

Set $C^{(0)} = \widehat{C}$, $D^{(0)} = \widehat{D}$, and partition these matrices into 2×2 block matrices as follows:

$$C^{(0)} = \left[\begin{array}{c|ccccc} 0 & W^{(0)} & 0 & \dots & \dots & 0 \\ \hline -A_0 & & & & & \\ 0 & & & T^{(0)} & & \\ \vdots & & & & & \\ 0 & & & & & \end{array}\right], \quad D^{(0)} = \left[\begin{array}{c|c} I_m & s^{(0)\mathrm{T}} \\ \hline 0 & \\ 0 & V^{(0)} \\ \vdots & \\ 0 & \end{array}\right],$$

where

$$W^{(0)} = I_m, \;\; T^{(0)} = I_{m(n-1)} + \boldsymbol{d}^{(0)}\boldsymbol{e}_1^{\mathrm{T}}, \;\; \boldsymbol{d}^{(0)} = 0, \;\; \boldsymbol{s}^{(0)} = 0,$$

(2.2)
$$V^{(0)} = \begin{bmatrix} A_2 & A_3 & \dots & A_n \\ I_m & 0 & \dots & 0 \\ & \ddots & \ddots & \vdots \\ 0 & & I_m & 0 \end{bmatrix}.$$

Define the sequences of matrices $\{C^{(k)}\}_{k\geq 0}$, $\{D^{(k)}\}_{k\geq 0}$,

(2.3)
$$C^{(k)} = \left[\begin{array}{c|cccc} 0 & W^{(k)} & 0 & \dots & 0 \\ \hline -A_0 & & & & \\ 0 & & T^{(k)} & & \\ \vdots & & & & \\ 0 & & & & \end{array}\right], D^{(k)} = \left[\begin{array}{c|c} I_m & \boldsymbol{s}^{(k)\mathrm{T}} \\ \hline 0 & \\ 0 & V^{(k)} \\ \vdots & \\ 0 & \end{array}\right],$$

such that

$$T^{(k)} = I_{m(n-1)} + \boldsymbol{d}^{(k)}\boldsymbol{e}_1^{\mathrm{T}},$$

and $\boldsymbol{d}^{(k)}$, $W^{(k)}$, $V^{(k)}$, $\boldsymbol{s}^{(k)}$ are defined by means of the following recursions, starting from (2.2),

(2.4)
$$\begin{aligned} \boldsymbol{d}^{(k+1)} &= \boldsymbol{d}^{(k)} - V^{(k)}Y^{(k)^{-1}}\boldsymbol{e}_1 A_0 W^{(k)}, \\ W^{(k+1)} &= W^{(k)}(\boldsymbol{e}_1^{\mathrm{T}}Y^{(k)^{-1}}\boldsymbol{e}_1)A_0 W^{(k)}, \\ V^{(k+1)} &= V^{(k)}Y^{(k)^{-1}}V^{(k)}, \\ \boldsymbol{s}^{(k+1)\mathrm{T}} &= \boldsymbol{s}^{(k)\mathrm{T}} - W^{(k)}\boldsymbol{e}_1^{\mathrm{T}}Y^{(k)^{-1}}V^{(k)}, \; k \geq 0, \end{aligned}$$

where

(2.5)
$$Y^{(k)} = I_{m(n-1)} + \boldsymbol{d}^{(k)}\boldsymbol{e}_1^{\mathrm{T}} + \boldsymbol{e}_1 A_0 \boldsymbol{s}^{(k)\mathrm{T}} = T^{(k)} + \boldsymbol{e}_1 A_0 \boldsymbol{s}^{(k)\mathrm{T}},$$

provided that $Y^{(k)}$ is nonsingular for any $k \geq 0$.

THEOREM 2.1. *Assume that the matrix $Y^{(k)}$ of (2.5) is nonsingular for any $k \geq 0$. Then the matrices $C^{(k)}$, $D^{(k)}$, $k \geq 1$, are well defined and satisfy the following properties:*

ETNA
Kent State University
etna@mcs.kent.edu

1. *$C^{(k+1)} = L^{(k)}C^{(k)}$, $D^{(k+1)} = U^{(k)}D^{(k)}$, $k \geq 0$, where $L^{(k)}$, $U^{(k)}$ are matrices with the structure*

$$L^{(k)} = \left[\begin{array}{c|ccc} L_1^{(k)} & 0 & \dots & 0 \\ \hline \boldsymbol{l}_2^{(k)} & & I_{m(n-1)} & \\ & & & \end{array}\right], \quad U^{(k)} = \left[\begin{array}{c|c} I_m & \boldsymbol{u}_1^{(k)T} \\ \hline 0 & \\ 0 & U_2^{(k)} \\ \vdots & \\ 0 & \end{array}\right]$$

such that $U^{(k)}C^{(k)} = L^{(k)}D^{(k)}$;

2. *$C^{(k)}\boldsymbol{g} = D^{(k)}\boldsymbol{g}G^{2^k}$, for $k = 0, 1, \dots$.*

Proof. Let us prove the first part of the theorem by using an induction argument. For $k = 0$ let the matrices $L^{(0)}$, $U^{(0)}$ have the structure stated in the theorem, and satisfy $U^{(0)}C^{(0)} = L^{(0)}D^{(0)}$. After equating the block entries in the latter equality we obtain that

$$\begin{aligned} L_1^{(0)} &= -\boldsymbol{u}_1^{(0)\mathrm{T}}\boldsymbol{e}_1 A_0, \\ L_1^{(0)}\boldsymbol{s}^{(0)\mathrm{T}} &= W^{(0)}\boldsymbol{e}_1^{\mathrm{T}} + \boldsymbol{u}_1^{(0)\mathrm{T}}T^{(0)}, \\ \boldsymbol{l}_2^{(0)} &= -U_2^{(0)}\boldsymbol{e}_1 A_0, \\ \boldsymbol{l}_2^{(0)}\boldsymbol{s}^{(0)\mathrm{T}} + V^{(0)} &= U_2^{(0)}T^{(0)}, \end{aligned}$$

whence we deduce that

$$\begin{aligned} L_1^{(0)} &= W^{(0)}\boldsymbol{e}_1^{\mathrm{T}}{Y^{(0)}}^{-1}\boldsymbol{e}_1 A_0, \\ \boldsymbol{l}_2^{(0)} &= -V^{(0)}{Y^{(0)}}^{-1}\boldsymbol{e}_1 A_0, \\ \boldsymbol{u}_1^{(0)\mathrm{T}} &= -W^{(0)}\boldsymbol{e}_1^{\mathrm{T}}{Y^{(0)}}^{-1}, \\ U_2^{(0)} &= V^{(0)}{Y^{(0)}}^{-1}. \end{aligned} \tag{2.6}$$

If we define the matrices $H = L^{(0)}C^{(0)}$, $K = U^{(0)}D^{(0)}$, from the structure of $C^{(0)}$, $D^{(0)}$, $L^{(0)}$, $U^{(0)}$, we find that

$$H = \left[\begin{array}{c|cccc} 0 & H_1 & 0 & \dots & 0 \\ \hline -A_0 & & & & \\ 0 & & H_2 & & \\ \vdots & & & & \\ 0 & & & & \end{array}\right], \quad K = \left[\begin{array}{c|c} I_m & \boldsymbol{k}_1^{\mathrm{T}} \\ \hline 0 & \\ 0 & K_2 \\ \vdots & \\ 0 & \end{array}\right],$$

where

$$\begin{aligned} H_1 &= L_1^{(0)}W^{(0)}, \\ H_2 &= T^{(0)} + \boldsymbol{l}_2^{(0)}W^{(0)}\boldsymbol{e}_1^{\mathrm{T}}, \\ \boldsymbol{k}_1^{\mathrm{T}} &= \boldsymbol{s}^{(0)\mathrm{T}} + \boldsymbol{u}_1^{(0)\mathrm{T}}V^{(0)}, \\ K_2 &= U_2^{(0)}V^{(0)}. \end{aligned}$$

By substituting relations (2.6) in the above equations, we deduce from (2.4) that $H = C^{(1)}$, $K = D^{(1)}$. The inductive step can be proved by using the same arguments. Concerning the second part of the theorem, observe that by using the property $C^{(1)} = L^{(0)}C^{(0)}$ and $D^{(1)} = U^{(0)}D^{(0)}$ we have

$$\begin{gathered} C^{(1)}\boldsymbol{g} = L^{(0)}C^{(0)}\boldsymbol{g} = L^{(0)}D^{(0)}\boldsymbol{g}G = \\ U^{(0)}C^{(0)}\boldsymbol{g}G = U^{(0)}D^{(0)}\boldsymbol{g}G^2 = D^{(1)}\boldsymbol{g}G^2. \end{gathered}$$

ETNA
Kent State University
etna@mcs.kent.edu

Thus, by induction, it follows that $C^{(k)}\boldsymbol{g} = D^{(k)}\boldsymbol{g}G^{2^k}$, for $k = 0, 1, \ldots$. □

REMARK 2.2 (Relation with cyclic reduction). We observe that the matrices $V^{(k)}$, $Y^{(k)}$ can be viewed as the blocks generated at the k-th step of the cyclic reduction applied to a suitable block tridiagonal infinite matrix [13, 7]. For this purpose, let us define the matrices

$$F^{(k)} = \boldsymbol{e}_1 A_0 W^{(k)} \boldsymbol{e}_1^{\mathrm{T}}.$$

Then, from (2.4) it follows that

$$F^{(k+1)} = F^{(k)} {Y^{(k)}}^{-1} F^{(k)}.$$

Moreover, we can write

$$T^{(k+1)} = T^{(k)} - V^{(k)} {Y^{(k)}}^{-1} F^{(k)},$$

and we can easily observe that

$$Y^{(k+1)} = Y^{(k)} - V^{(k)} {Y^{(k)}}^{-1} F^{(k)} - F^{(k)} {Y^{(k)}}^{-1} V^{(k)}.$$

The above recurrences, together with the relation for the matrices $V^{(k)}$ in (2.4), allow one to conclude that the matrices $T^{(k)}$, $V^{(k)}$, $F^{(k)}$, $Y^{(k)}$ are the blocks obtained at the k-th step of cyclic reduction applied to the infinite matrix

$$\begin{bmatrix} T^{(0)} & -V^{(0)} & & 0 \\ -F^{(0)} & Y^{(0)} & -V^{(0)} & \\ & -F^{(0)} & Y^{(0)} & \ddots \\ 0 & & \ddots & \ddots \end{bmatrix}. \tag{2.7}$$

Consider now the relation $C^{(k)}\boldsymbol{g} = D^{(k)}\boldsymbol{g}G^{2^k}$, $k = 0, 1, \ldots$. By deleting the first block row in this equality, we obtain that

$$\left[\begin{array}{c|c} -A_0 & \\ 0 & \\ \vdots & T^{(k)} \\ 0 & \end{array}\right] \boldsymbol{g} = \left[\begin{array}{c|c} 0 & \\ 0 & \\ \vdots & V^{(k)} \\ 0 & \end{array}\right] \boldsymbol{g}G^{2^k},$$

i.e.,

$$T^{(k)}\boldsymbol{g}_1 - \begin{bmatrix} A_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = V^{(k)}\boldsymbol{g}_1 G^{2^k}, \quad k = 0, 1, \ldots, \tag{2.8}$$

where

$$\boldsymbol{g}_1 = \begin{bmatrix} G \\ G^2 \\ \vdots \\ G^{n-1} \end{bmatrix}.$$

If the right hand side of equation (2.8) converges to zero, we may approximate G with the first block entry X_1 of the solution of the linear system

$$T^{(k)} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{n-1} \end{bmatrix} = \begin{bmatrix} A_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{2.9}$$

for a sufficiently large value of k. Moreover, due to the special structure of $T^{(k)}$, in order to compute X_1 in (2.9), it is sufficient to solve the $m \times m$ linear system

$$(I_m + \boldsymbol{d}_1^{(k)})X_1 = A_0, \tag{2.10}$$

where $\boldsymbol{d}_1^{(k)}$ denotes the first block component of the block vector $\boldsymbol{d}^{(k)}$. In fact, such convergence property holds, as we will show in the next theorem. Before stating this convergence result, we need to introduce the polynomial of degree at most mn

$$a(z) = \det(zI - \sum_{i=0}^{n} z^i A_i). \tag{2.11}$$

We may assume, without loss of generality, that 1 is the only zero of $a(z)$ having modulus one [15]. Under this assumption, since the M/G/1 Markov chain is positive recurrent, the function $a(z)$ has m zeros in the closed unit disk (including a zero equal to 1) and $m(n-1)$ zeros outside the unit disk, where we put zeros equal to infinity if A_n is singular; moreover, the matrix G is stochastic and its eigenvalues are the zeros of $a(z)$ in the closed unit disk (see [27, 15]).

THEOREM 2.3. *Let* $\sigma = 1/\min\{|z| : |z| > 1, \ a(z) = 0\}$. *Then for any matrix norm and for any* $\epsilon > 0$ *such that* $\epsilon + \sigma < 1$, *one has* $||V^{(k)}|| = O((\sigma + \epsilon)^{2^k})$.

Proof. Consider the polynomial $q(z) = \det(zI - F^{(0)} - z^2 V^{(0)})$, having degree at most $2m(n-1)$. Since the rank of $F^{(0)}$ is at most m, it follows that $\xi = 0$ is a zero of $q(z)$ of multiplicity at least $(n-2)m$. Moreover, for the properties of block Frobenius matrices, if ξ is a zero of $a(z)$, then ξ is a zero of $q(z)$. Thus, ξ is a zero of $q(z)$ if and only if ξ is a zero of $z^{(n-2)m}a(z)$. From this property, since the M/G/1 Markov chain is positive recurrent, it follows that $q(z)$ has exactly $m(n-1)$ zeros inside the closed unit disk (see [15]). Hence, from the results of [15], the matrix equations $X = V^{(0)} + X^2 F^{(0)}$ and $X = F^{(0)} + V^{(0)} X^2$ have a minimal nonnegative solution $\hat{R}$ and $\hat{G}$, respectively, such that $\rho(\hat{R}) = \sigma$, and $\hat{G}$ is stochastic. From Remark 2.2 and from the results of [7], it follows that

$$-V^{(k)} + \hat{R}^{2^k} Y^{(k)} - \hat{R}^{2 \cdot 2^k} F^{(k)} = 0, \quad F^{(k)} + Y^{(k)} \hat{G}^{2^k} - V^{(k)} \hat{G}^{2 \cdot 2^k} = 0, \quad k \geq 0.$$

From the results on cyclic reduction applied to quadratic matrix polynomials of [7, 4], one has that the sequence $\{F^{(k)}\}_k$ is uniformly bounded and that $\{V^{(k)}\}_k$ converges to zero as $O((\sigma + \epsilon)^{2^k})$, for any $\epsilon > 0$ such that $\sigma + \epsilon < 1$. ☐

Since G is a stochastic matrix, the entries of $\boldsymbol{g}_1$ and of G^{2^k} are nonnegative and bounded from above by a constant. Therefore, for the right hand-side of (2.8), we have that

$$||V^{(k)} \boldsymbol{g}_1 G^{2^k}|| = O((\sigma + \epsilon)^{2^k}).$$

As a corollary of the above theorem we have also that, if the norm of ${Y^{(k)}}^{-1}$ remains bounded from above by a constant for any k, then the sequence $\{\boldsymbol{d}^{(k)}\}_k$ is convergent.

ETNA
Kent State University
etna@mcs.kent.edu

The resulting algorithm is the following:

ALGORITHM 2.4.

INPUT: Nonnegative matrices A_i, $i = 0, \dots, n$, a small real number $\epsilon > 0$ for the stopping condition, a matrix norm $||\cdot||$.

OUTPUT: An approximation $\tilde{G}$ of the minimal nonnegative solution of (1.1).

COMPUTATION:

1. Set $A_0 = (I - A_1)^{-1}A_0$, $A_i = (I - A_1)^{-1}A_i$, $i = 2, \dots, n$, $A_1 = 0$.
2. Set $k = 0$ and

$$W^{(0)} = I_m, \ \ \boldsymbol{d}^{(0)} = 0, \ \ \boldsymbol{s}^{(0)} = 0, \ \ V^{(0)} = \begin{bmatrix} A_2 & A_3 & \dots & A_n \\ I_m & 0 & \dots & 0 \\ & \ddots & \ddots & \vdots \\ 0 & & I_m & 0 \end{bmatrix}.$$

3. Compute

$$\begin{aligned} \boldsymbol{d}^{(k+1)} &= \boldsymbol{d}^{(k)} - V^{(k)}{Y^{(k)}}^{-1}\boldsymbol{e}_1 A_0 W^{(k)}, \\ W^{(k+1)} &= W^{(k)}(\boldsymbol{e}_1^{\mathrm{T}}{Y^{(k)}}^{-1}\boldsymbol{e}_1)A_0 W^{(k)}, \\ V^{(k+1)} &= V^{(k)}{Y^{(k)}}^{-1}V^{(k)}, \\ \boldsymbol{s}^{(k+1)\mathrm{T}} &= \boldsymbol{s}^{(k)\mathrm{T}} - W^{(k)}\boldsymbol{e}_1^{\mathrm{T}}{Y^{(k)}}^{-1}V^{(k)}, \end{aligned}$$

   where $Y^{(k)} = I_{m(n-1)} + \boldsymbol{d}^{(k)}\boldsymbol{e}_1^{\mathrm{T}} + \boldsymbol{e}_1 A_0 \boldsymbol{s}^{(k)\mathrm{T}}$.
4. If $||\boldsymbol{d}_1^{(k+1)} - \boldsymbol{d}_1^{(k)}|| \geq \epsilon$, where $\boldsymbol{d}_1^{(k)}$ is the first block component of $\boldsymbol{d}^{(k)}$, then set $k = k + 1$ and repeat step 3. Otherwise solve the linear system (2.10) and output $\tilde{G} = X_1$.

From (2.5) it follows that the matrix $Y^{(k)}$, $k \geq 0$, is the identity matrix plus a correction in the first block row and in the first block column. In particular, we have

$${Y^{(k)}}^{-1} = I_{m(n-1)} - \left[\begin{array}{c|c} \boldsymbol{e}_1 A_0 & \boldsymbol{d}^{(k)} \end{array}\right] \Theta^{(k)} \left[\begin{array}{c} \boldsymbol{s}^{(k)\mathrm{T}} \\ \hline \boldsymbol{e}_1^{\mathrm{T}} \end{array}\right],$$

where

$$\Theta^{(k)} = \left(I_{2m} + \begin{bmatrix} \boldsymbol{s}^{(k)\mathrm{T}}\boldsymbol{e}_1 A_0 & \boldsymbol{s}^{(k)\mathrm{T}}\boldsymbol{d}^{(k)} \\ A_0 & \boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{d}^{(k)} \end{bmatrix} \right)^{-1}.$$

Hence the computation of $\boldsymbol{d}^{(k)}$, ${Y^{(k)}}^{-1}$, $W^{(k)}$ and $\boldsymbol{s}^{(k)}$ can be reduced to operations between block vectors, with a computational cost of $O(m^3 n)$ arithmetic operations. The matrix $V^{(k)}$, $k > 1$, does not have any evident structure, since it is a full matrix. For this reason the computation of this matrix can amount to $O(m^3 n^2)$ arithmetic operations, which is a very high cost. In section 3 we will show that also the matrix $V^{(k)}$ has a structure, which relies on the concept of displacement rank. By using this concept we will show that the matrix $V^{(k)}$ is represented by at most 4 block vectors, and these block vectors can be computed by means of FFTs with $O(m^3 n + m^2 n \log n)$ ops.

Concerning the nonsingularity of $Y^{(k)}$, from Remark 2.2 and from the results of [4] we have that $Y^{(k)}$ is nonsingular if and only if the $(2^k - 1) \times (2^k - 1)$ block leading principal submatrix of the matrix of (2.7) is nonsingular. We were not able to show that these submatrices are nonsingular. In the numerical experiments we have never encountered singularity. In any case, if we would encounter singularity, we could apply a doubling strategy, as described in [3], until nonsingularity is obtained. We refer the reader to [3] for details. The same technique can be used to overcome a possible ill-conditioning of the matrices $Y^{(k)}$. It can be shown (see [7]) that the sequence $\{Y^{(k)}\}_k$ converges to a nonsingular matrix. However, it is not clear what the conditioning of the limit matrix is.

In principle, we could directly apply our algorithm to the pair (C, D), instead of to the pair $(\hat{C}, \hat{D})$. In other words, we could choose $C^{(0)} = C$ and $D^{(0)} = D$, instead of $C^{(0)} = \hat{C}$ and $D^{(0)} = \hat{D}$. However, by starting with $C^{(0)} = \hat{C}$ and $D^{(0)} = \hat{D}$ the structure of the matrices $\{C^{(k)}\}_{k\geq 1}$, $\{D^{(k)}\}_{k\geq 1}$ that we generate is simplified, compared to the case $C^{(0)} = C$ and $D^{(0)} = D$. Thus we have chosen to use the pair $(\hat{C}, \hat{D})$.

3. The algorithm revised with displacement properties. In this section we show that the matrices $V^{(k)}$, $k \geq 0$, have a displacement structure. In particular, we show that they have block displacement rank at most 3 and we provide recursive formulas for the generators. These formulas allow us to implement a version of our algorithm with low computational cost, which fully exploits the Toeplitz structure of the involved matrices. We recall in Appendix A the concept and the properties of displacement rank, the definition of block Toeplitz-like matrices, and the fast algorithms for computing the product of a block Toeplitz matrix and a block vector.

For any $k \geq 0$, let us define the quadratic matrix polynomial

$$\phi^{(k)}(z) = -V^{(k)} + zY^{(k)} - z^2 \boldsymbol{e}_1 A_0 W^{(k)} \boldsymbol{e}_1^{\mathrm{T}}.$$

The next theorem shows that, for any z and for any $k \geq 0$, the matrix $\phi^{(k)}(z)$ has displacement rank at most 3, with respect to the block displacement operator $\Delta(A) = ZA - AZ$, where

$$Z = \begin{bmatrix} 0 & & & 0 \\ I_m & 0 & & \\ & \ddots & \ddots & \\ 0 & & I_m & 0 \end{bmatrix}.$$

THEOREM 3.1. *For any z and for any $k \geq 0$ we have*

$$\begin{aligned}&\Delta(\phi^{(k)}(z)) = \\ &(\boldsymbol{u}^{(k)} + z\boldsymbol{e}_1 Q^{(k)})\boldsymbol{e}_1^{\mathrm{T}}\phi^{(k)}(z) + \phi^{(k)}(z)\boldsymbol{e}_1(\boldsymbol{r}^{(k)\mathrm{T}} + zW^{(k)}\boldsymbol{e}_1^{\mathrm{T}}) - \boldsymbol{e}_1\boldsymbol{\gamma}^{\mathrm{T}}\phi^{(k)}(z),\end{aligned} \tag{3.1}$$

where

$$\begin{aligned}
&\boldsymbol{\gamma}^{\mathrm{T}} = [0, A_3, \ldots, A_n], \\
&\boldsymbol{u}^{(0)} = \boldsymbol{e}_2, \quad Q^{(0)} = -I, \\
&\boldsymbol{r}^{(0)\mathrm{T}} = \boldsymbol{0}^{\mathrm{T}}, \\
&\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + V^{(k)}{Y^{(k)}}^{-1}\boldsymbol{e}_1 Q^{(k)}, \\
&Q^{(k+1)} = A_0 W^{(k)}(\boldsymbol{e}_1^{\mathrm{T}}{Y^{(k)}}^{-1}\boldsymbol{e}_1)Q^{(k)}, \\
&\boldsymbol{r}^{(k+1)\mathrm{T}} = \boldsymbol{r}^{(k)\mathrm{T}} + W^{(k)}\boldsymbol{e}_1^{\mathrm{T}}{Y^{(k)}}^{-1}V^{(k)}.
\end{aligned} \tag{3.2}$$

Proof. Let us prove the theorem by induction:

$$\begin{gathered}\Delta(\phi^{(0)}(z)) = Z\phi^{(0)}(z) - \phi^{(0)}(z)Z = \\ -\boldsymbol{e}_2[A_2 + z^2 A_0, A_3, \ldots, A_n] + \boldsymbol{e}_1[A_3, \ldots, A_n, 0] = \\ \boldsymbol{e}_2\boldsymbol{e}_1^{\mathrm{T}}\phi^{(0)}(z) - \boldsymbol{e}_2 z\boldsymbol{e}_1^{\mathrm{T}} - \boldsymbol{e}_1\boldsymbol{e}_1^{\mathrm{T}}\phi^{(0)}(z)Z.\end{gathered}$$

ETNA
Kent State University
etna@mcs.kent.edu

It is a simple calculation to show that

$$-\boldsymbol{e}_2 z\boldsymbol{e}_1^{\mathrm{T}} - \boldsymbol{e}_1\boldsymbol{e}_1^{\mathrm{T}}\phi^{(0)}(z)Z = -\boldsymbol{e}_1 z\boldsymbol{e}_1^{\mathrm{T}}\phi^{(0)}(z) + \phi^{(0)}(z)\boldsymbol{e}_1 z\boldsymbol{e}_1^{\mathrm{T}} - \boldsymbol{e}_1\boldsymbol{\gamma}^{\mathrm{T}}\phi^{(0)}(z).$$

Hence

$$\Delta(\phi^{(0)}(z)) = (\boldsymbol{e}_2 - z\boldsymbol{e}_1)\boldsymbol{e}_1^{\mathrm{T}}\phi^{(0)}(z) + \phi^{(0)}(z)\boldsymbol{e}_1 z\boldsymbol{e}_1^{\mathrm{T}} - \boldsymbol{e}_1\boldsymbol{\gamma}^{\mathrm{T}}\phi^{(0)}(z).$$

Now, let us assume that (3.1) holds for a $k \geq 0$, and let us show it for $k+1$. For Remark 2.2 and for the results of [9] $\phi^{(k+1)}(z^2) = 2zB(z)^{-1}$, where $B(z) = (\phi^{(k)}(z))^{-1} - (\phi^{(k)}(-z))^{-1}$. Thus, since (compare [9])

$$B(z)^{-1} = -\phi^{(k)}(z){Y^{(k)}}^{-1}\phi^{(k)}(-z)/(2z) = -\phi^{(k)}(-z){Y^{(k)}}^{-1}\phi^{(k)}(z)/(2z),$$

we obtain, after some algebraic manipulations:

$$\begin{aligned}
&\Delta(\phi^{(k+1)}(z^2)) = 2z\Delta(B(z)^{-1}) = \\
&2zB(z)^{-1}\left(\phi^{(k)}(z)^{-1}\Delta(\phi^{(k)}(z))\phi^{(k)}(z)^{-1}\right.\\
&\quad\left. -\ \phi^{(k)}(-z)^{-1}\Delta(\phi^{(k)}(-z))\phi^{(k)}(-z)^{-1}\right)B(z)^{-1} = \\
&\boldsymbol{u}^{(k)}\boldsymbol{e}_1^{\mathrm{T}}\phi^{(k+1)}(z^2) - \left(\phi^{(k)}(z) + \phi^{(k)}(-z)\right){Y^{(k)}}^{-1}\boldsymbol{e}_1 Q^{(k)}\boldsymbol{e}_1^{\mathrm{T}}\phi^{(k+1)}(z^2)/2 + \\
&\quad\phi^{(k+1)}(z^2)\boldsymbol{e}_1\boldsymbol{r}^{(k)\mathrm{T}} - \phi^{(k+1)}(z^2)\boldsymbol{e}_1 W^{(k)}\boldsymbol{e}_1^{\mathrm{T}}{Y^{(k)}}^{-1}\left(\phi^{(k)}(z)) + \phi^{(k)}(-z)\right)/2 - \\
&\quad\boldsymbol{e}_1\boldsymbol{\gamma}^{\mathrm{T}}\phi^{(k+1)}(z^2) = \\
&\left((\boldsymbol{u}^{(k)} + V^{(k)}{Y^{(k)}}^{-1}\boldsymbol{e}_1 Q^{(k)}) + z^2\boldsymbol{e}_1 A_0 W^{(k)}(\boldsymbol{e}_1^{\mathrm{T}}{Y^{(k)}}^{-1}\boldsymbol{e}_1)Q^{(k)}\right)\boldsymbol{e}_1^{\mathrm{T}}\phi^{(k+1)}(z^2) + \\
&\phi^{(k+1)}(z^2)\boldsymbol{e}_1\left((\boldsymbol{r}^{(k)\mathrm{T}} + W^{(k)}\boldsymbol{e}_1^{\mathrm{T}}{Y^{(k)}}^{-1}V^{(k)}) + z^2 W^{(k)}(\boldsymbol{e}_1^{\mathrm{T}}{Y^{(k)}}^{-1}\boldsymbol{e}_1)A_0 W^{(k)}\right) - \\
&\boldsymbol{e}_1\boldsymbol{\gamma}^{\mathrm{T}}\phi^{(k+1)}(z^2).
\end{aligned}$$

□

From the above theorem, since $V^{(k)} = -\phi^{(k)}(0)$ we immediately obtain the following property of the matrices $V^{(k)}$, $k \geq 0$:

THEOREM 3.2. *For any $k \geq 0$ the matrices $V^{(k)}$ have the following displacement structure:*

$$\Delta(V^{(k)}) = \boldsymbol{u}^{(k)}\boldsymbol{e}_1^{\mathrm{T}}V^{(k)} + V^{(k)}\boldsymbol{e}_1\boldsymbol{r}^{(k)\mathrm{T}} - \boldsymbol{e}_1\boldsymbol{\gamma}^{\mathrm{T}}V^{(k)},$$

where $\boldsymbol{u}^{(k)}$, $\boldsymbol{r}^{(k)}$, $\boldsymbol{\gamma}$ are defined in (3.2).

From the above theorem, the matrix $V^{(k)}$ is only defined in terms of the block vectors $\boldsymbol{u}^{(k)}$, $\boldsymbol{r}^{(k)}$, $\boldsymbol{e}_1^{\mathrm{T}}V^{(k)}$, $V^{(k)}\boldsymbol{e}_1$, $\boldsymbol{\gamma}^{\mathrm{T}}V^{(k)}$. The computation of these vectors can be performed by using the fast techniques recalled in Appendix A, which rely on the displacement structure of $V^{(k)}$. In particular, from (2.4), we have

$$\boldsymbol{e}_1^{\mathrm{T}}V^{(k+1)} = \boldsymbol{e}_1^{\mathrm{T}}V^{(k)}{Y^{(k)}}^{-1}V^{(k)}.$$

Thus, given $\boldsymbol{e}_1^{\mathrm{T}}V^{(k)}$, the vector $\boldsymbol{y}^{\mathrm{T}} = \boldsymbol{e}_1^{\mathrm{T}}V^{(k)}{Y^{(k)}}^{-1}$ is computed by means of $O(nm^3)$ ops, by using the structure of ${Y^{(k)}}^{-1}$. The product $\boldsymbol{y}^{\mathrm{T}}V^{(k)}$ is computed by exploiting the displacement structure of $V^{(k)}$, by means of $O(m^2 n\log n + m^3 n)$ ops.

4. Algorithm for R. Our algorithm can be used to solve the dual problem, that is, computing R which satisfies the polynomial equation (1.2).

Since the Markov chain is positive recurrent, the polynomial $a(z)$ of (2.11) has m zeros inside the open unit disk and $m(n-1)$ eigenvalues outside the open unit disk. Moreover, we may assume, without loss of generality, that $z = 1$ is the only zero of modulus one. The matrix R has spectral radius less than one, and its eigenvalues are the zeros of $a(z)$ in the open unit disk.

Note that the equation (1.2) is equivalent to

$$R^{\mathrm{T}} = A_0^{\mathrm{T}} + A_1^{\mathrm{T}} R^{\mathrm{T}} + \cdots + A_n^{\mathrm{T}} (R^{\mathrm{T}})^n$$

and R^{T} satisfies $C\boldsymbol{r} = D\boldsymbol{r}R^{\mathrm{T}}$, where

$$\boldsymbol{r} = \begin{bmatrix} I \\ R^{\mathrm{T}} \\ (R^{\mathrm{T}})^2 \\ \vdots \\ (R^{\mathrm{T}})^{n-1} \end{bmatrix},$$

and C, D are as defined in (1.4), where the A_i are replaced with A_i^{T}.

Thus, we may apply exactly the same algorithm. As for the algorithm for G, ξ is a zero of $q(z) = \det(zI - F^{(0)} - z^2 V^{(0)})$ if and only if ξ is a zero of $a(z)z^{m(n-2)}$. In particular, since $a(z)$ has exactly m zeros inside the open unit disk, then $q(z)$ has exactly $m(n-1)$ zeros inside the open unit disk.

The matrix $V^{(0)} + F^{(0)}$ is not stochastic, but it is a simple calculation to show that the matrix $\hat{V}^{(0)} + \hat{F}^{(0)}$ is stochastic, where $\hat{V}^{(0)} = (I \otimes D(\boldsymbol{\pi}))^{-1} V^{(0)} (I \otimes D(\boldsymbol{\pi}))$, $\hat{F}^{(0)} = (I \otimes D(\boldsymbol{\pi}))^{-1} F^{(0)} (I \otimes D(\boldsymbol{\pi}))$, $D(\boldsymbol{\pi})$ is the $m \times m$ diagonal matrix whose diagonal elements are the entries of $\boldsymbol{\pi}$, and $\boldsymbol{\pi}$ is the steady state vector of $A = \sum_{i=0}^n A_i$.

From the results of [15], the matrix equations $X = \hat{V}^{(0)} + X^2 \hat{F}^{(0)}$ and $X = \hat{F}^{(0)} + V^{(0)} \hat{X}^2$ have a minimal nonnegative solution $\hat{R}$ and $\hat{G}$, respectively, such that $\rho(\hat{G}) = \theta$, where $\theta = \max\{|z| : |z| < 1,\ a(z) = 0\}$, and $\rho(\hat{R}) = 1$. Let us define $\hat{F}^{(k)} = (I \otimes D(\boldsymbol{\pi}))^{-1} F^{(k)} (I \otimes D(\boldsymbol{\pi}))$, $\hat{V}^{(k)} = (I \otimes D(\boldsymbol{\pi}))^{-1} V^{(k)} (I \otimes D(\boldsymbol{\pi}))$. As for the algorithm for G, from Remark 2.2 and from the results of [7, 4], the sequence $\{\hat{V}^{(k)}\}_k$ is uniformly bounded and $\{\hat{F}^{(k)}\}_k$ converges to zero of order $O((\theta + \epsilon)^{2^k})$, for any $\epsilon > 0$ such that $\theta + \epsilon < 1$. In particular, the same convergence properties hold for $\{V^{(k)}\}_k$ and $\{F^{(k)}\}_k$. Since $\rho(R) = \theta$, for the right hand-side of (2.8), we have

$$||V^{(k)} \boldsymbol{r}_1 R^{2^k}|| = O((\epsilon + \theta)^{2 \cdot 2^k})$$

for any matrix norm.

5. Shifting technique. The speed of convergence of our algorithm can be improved by applying the shifting technique introduced in [17] for Quasi-Birth-Death problems, and expressed in [5, 10] in functional form as follows. Let $\boldsymbol{u}$ be a nonnegative vector of length k such that $\boldsymbol{u}^{\mathrm{T}}\boldsymbol{e} = 1$. Define the rank one matrix $E = \boldsymbol{e}\boldsymbol{u}^{\mathrm{T}}$ and the matrix Laurent polynomial $E(z) = I - z^{-1}E$. Observe that, by setting

$$\hat{A}(z) = \hat{A}_0 + z\hat{A}_1 + z^2\hat{A}_2 + \ldots + z^n\hat{A}_n,$$

where

$$\begin{aligned}
&\hat{A}_0 = A_0 + \left(\sum_{j=1}^n A_j - I\right) E = A_0(I - E), \\
(5.1)\qquad &\hat{A}_i = A_i + \left(\sum_{j=i+1}^n A_j\right) E, \quad \text{for} \quad i = 1, \ldots, n-1, \\
&\hat{A}_n = A_n,
\end{aligned}$$

ETNA
Kent State University
etna@mcs.kent.edu

since $\hat{A}_0 E = 0$ and $E^i = E$ for any $i > 0$, we have

$$zI - A(z) = (zI - \hat{A}(z))E(z).$$

Now, for the sake of simplicity assume that A_n is nonsingular (the case where $\det A_n = 0$ can be treated by means of a continuity argument). Since $\widehat{A}_n = A_n$ the polynomials $a(z) = \det(zI - A(z))$ and $\hat{a}(z) = \det(zI - \hat{A}(z))$ have the same degree, and, therefore the same number of roots.

Since the matrix $I - z^{-1}E$ is defined for $z \neq 0$ and is singular only for $z = 1$, we deduce that if λ is a zero of $a(z)$ and $\lambda \neq 0, 1$, then it is a zero of $\hat{a}(z)$ and vice-versa. Moreover, $a(1) = 0$. Thus, we obtain that $\hat{a}(z)$ has the same zeros as $a(z)$ except for $z = 1$ which is replaced in the case of $\hat{a}(z)$ by $z = 0$.

Let us now consider the problem of computing G. Since we have assumed that 1 is the only zero of $a(z)$ having modulus 1, we may define, as in [17], the matrix $H = G - \boldsymbol{e}\boldsymbol{u}^{\mathrm{T}}$. Then, it is easy to see that the eigenvalues of H are those of G except that in the case of H the eigenvalue 1 of G is replaced by 0. Moreover, $\boldsymbol{e}$ is an eigenvector of H corresponding to the eigenvalue 0, and hence $H\boldsymbol{e} = \boldsymbol{0}$. So we have

$$G^i = (H + \boldsymbol{e}\boldsymbol{u}^{\mathrm{T}})^i = H^i + \boldsymbol{e}\boldsymbol{u}^{\mathrm{T}}H^{i-1} + \ldots + \boldsymbol{e}\boldsymbol{u}^{\mathrm{T}}H + \boldsymbol{e}\boldsymbol{u}^{\mathrm{T}}, \quad i = 1, 2 \ldots, n.$$

So replacing G by $H + \boldsymbol{e}\boldsymbol{u}^{\mathrm{T}}$ in (1.1), we obtain the following shifted equation

$$H = B_0 + B_1 H + \ldots + B_n H^n \tag{5.2}$$

where $B_i = \hat{A}_i$. As before, the solution H of the equation (5.2) satisfies (2.1), where in C and D the matrices A_i are replaced with B_i, and G with H. Thus, assuming that all the intermediate matrices $Y^{(k)}$ are non-singular, we may apply the same algorithm. By using the same arguments of the proof of Theorem 2.3 we deduce that

$$||V^{(k)}\boldsymbol{h}_1 H^{2^k}|| = O\left(((\gamma + \epsilon)(\sigma + \epsilon))^{2^k}\right),$$

where $\gamma = \rho(H) < 1$, for any $\epsilon > 0$ such that $\gamma + \epsilon < 1$, $\sigma + \epsilon < 1$.

As before we can apply a shifting technique to the equation (1.2), where in this case the zero that is equal to 1 is moved to ∞, instead of to 0. Let

$$\begin{aligned}
\hat{A}_0 &= A_0, \\
\hat{A}_1 &= A_1 + A_0 E, \\
\hat{A}_i &= A_i + \left(\sum_{j=0}^{i-1} A_j - I\right) E = A_i - \left(\sum_{j=i}^{n} A_j\right) E, \quad \text{for} \quad i = 2, \ldots, n.
\end{aligned}$$

It turns out that R is also the solution of the shifted equation

$$R = \hat{A}_0 + R\hat{A}_1 + \ldots + R^n \hat{A}_n$$

as a direct result of the following identity

$$A_0 e = \sum_{i=1}^{n-1} R^i (\sum_{j=i+1}^{n} A_j e),$$

and that

$$zI - A(z) = (zI - \hat{A}(z))(I - zE).$$

δ	γ	$1/\sigma$	Iter. of Orig. Alg.	Iter. of Shifted Alg.
10^{-1}	0.07831112	1.33333333	8	5
10^{-2}	0.01174465	1.03030303	11	4
10^{-3}	0.02074893	1.00300300	14	4
10^{-4}	0.02164936	1.00030003	17	4
10^{-5}	0.02173941	1.00003000	21	4
10^{-6}	0.02174841	1.00000300	24	5
10^{-7}	0.02174931	1.00000030	27	4
10^{-8}	0.02174940	1.00000003	29	5

TABLE 6.1
δ, γ, σ, and the number of iterations for Example 1

If μ is the smallest zero in modulus of the polynomial $a(z)$ among the zeros which are outside the unit circle, we obtain that

$$||V^{(k)}\boldsymbol{r}_1 R^{2^k}|| = O\left(\left((\theta+\epsilon)^2(1/\mu+\epsilon)\right)^{2^k}\right)$$

for any $\epsilon > 0$ such that $\theta + \epsilon < 1$, $1/\mu + \epsilon < 1$.

6. Numerical Examples. We have implemented in MATLAB our algorithm for the computation of G and R, with and without the shifting technique. We have not used the displacement structure, thus in our results we report only the number of iterations and the residual error, and we do not report the execution time.

6.1. Computing G. We use the norm of the residual of the equation

$$Res. = \|A_0 + A_1 G + \ldots + A_n G^n - G\|_\infty$$

to check the accuracy of the computed solution. The stopping criterion for the original and the shifted algorithms is

$$\left\|\boldsymbol{d}_1^{(k+1)} - \boldsymbol{d}_1^{(k)}\right\|_\infty < 10^{-12}.$$

EXAMPLE 1. We construct $A_0 = W + \delta I$ and $A_1 = A_2 = W$, where W is the matrix having null diagonal entries and constant off-diagonal entries, and $0 < \delta < 1$. Note that the rate $\rho = \boldsymbol{\pi}^{\mathrm{T}}(A_1 + 2A_2)\boldsymbol{e} = 1 - \delta$. Thus, as δ approaches zero, the problem becomes more unstable. Table 6.1 and 6.2 report the results obtained with size $m = 16$.

EXAMPLE 2. We solve $\sum_{i=0}^{10} A_i G^i = G$. The matrices are $A_i = D^{-1}(s_i \bar{A}_i)$, for $i = 0, 1, 2, \ldots, 10$, where $\bar{A}_i$ are random matrices generated by the MATLAB command $rand$. The matrix size m is 10. The scalars s_k are respectively $s_0 = 1, s_1 = 1, s_2 = 0.5, s_3 = 0.0025, s_4 = 0.125, s_5 = 0.001, s_6 = 0.0005, s_7 = 0.0001, s_8 = 0.00005, s_9 = 0.00001, s_{10} = 0.00005$. The matrix D is a diagonal matrix whose entries are the row sums of $\sum_{i=0}^{n} s_i \bar{A}_i$ so that $(\sum_{i=0}^{n} A_i)\boldsymbol{e} = \boldsymbol{e}$. In this example $\gamma = 0.097488$ and $1/\sigma = 1.0099$. Table 6.3 reports the results.

We observe that in both the examples the residual errors are very small, and that the shifting technique provides a much faster convergence rate.

6.2. Computing R. We use the norm of the residual of the equation

$$Res. = \|A_0 + RA_1 + \ldots + R^n A_n - R\|_\infty .$$

ETNA
Kent State University
etna@mcs.kent.edu

δ	Res. of Orig. Alg.	Res. of Shifted Alg.
10^{-1}	1.6×10^{-15}	3.3×10^{-16}
10^{-2}	1.0×10^{-15}	5.8×10^{-16}
10^{-3}	1.3×10^{-15}	2.7×10^{-16}
10^{-4}	1.4×10^{-15}	3.0×10^{-16}
10^{-5}	1.3×10^{-15}	1.9×10^{-16}
10^{-6}	9.9×10^{-16}	2.6×10^{-16}
10^{-7}	1.3×10^{-15}	2.4×10^{-16}
10^{-8}	1.3×10^{-15}	2.6×10^{-16}

TABLE 6.2
The residuals of the solutions in Example 1

	Orig. Alg.	Shifted Alg.
Iter.	13	5
Res.	2.9×10^{-16}	5.0×10^{-16}

TABLE 6.3
The number of iterations and residuals of the solutions in Example 2

to measure how accurate the solution is. The stopping criterion for the original and the shifted algorithms is

$$\left\| \boldsymbol{d}_1^{(k+1)} - \boldsymbol{d}_1^{(k)} \right\|_\infty < 10^{-12}.$$

EXAMPLE 3. We construct $A_0 = A_1 = W$ and $A_2 = W + \delta I$, where W is the matrix having null diagonal entries and constant off-diagonal entries, and $0 < \delta < 1$. Note that the rate $\rho = \boldsymbol{\pi}^{\mathrm{T}}(A_1 + 2A_2)\boldsymbol{e} = 1 + \delta$. Thus, as δ approaches zero, the problem becomes more unstable. Table 6.4 and 6.5 report the results obtained with size $m = 16$.

EXAMPLE 4. We solve $R = \sum_{i=0}^{10} R^i A_i$. Here the matrices A_i are generated as in the Example 2. In this example, $\rho = 1.0006$, $\theta = 0.99892$, and $\mu = 2.7410$. Table 6.6 reports the results.

Also for the computation of the matrix R the residual errors are very small, and the shifting technique allows one to considerably reduce the number of iterations.

Appendix A. Displacement rank and fast Toeplitz computations. In this section we recall the concept of displacement rank, introduced by Kailath et al. [20], and elaborated in many other papers (see, for instance, [18, 11, 21] and the references cited therein). This concept is fundamental in devising and analyzing algorithms related to Toeplitz matrices, and will allow us to design an effective version of our algorithm that fully exploits the structure of the matrices $V^{(k)}$. Finally, we will recall the fast algorithm based on the use of FFTs for computing the product between a block Toeplitz matrix and a block vector.

Define the $h \times h$ *block down-shift* matrix

$$Z = \begin{bmatrix} 0 & & & \\ I & 0 & & \\ & \ddots & \ddots & \\ & & I & 0 \end{bmatrix},$$

where the blocks have dimension m, and consider the following *block displacement operator*

$$\Delta(A) = ZA - AZ$$

δ	θ	μ	Iter. of Orig. Alg.	Iter. of Shifted Alg.
10^{-1}	0.75000000	12.769578	8	5
10^{-2}	0.97058824	85.145135	11	4
10^{-3}	0.99700599	48.195253	14	4
10^{-4}	0.99970006	46.190730	17	4
10^{-5}	0.99997000	45.999410	21	5
10^{-6}	0.99999700	45.980366	24	4
10^{-7}	0.99999970	45.978462	27	4
10^{-8}	0.99999997	45.978272	29	4

TABLE 6.4
δ, θ, μ, and the number of iterations for Example 3

δ	Res. of Orig. Alg.	Res. of Shifted Alg.
10^{-1}	3.9×10^{-16}	2.0×10^{-16}
10^{-2}	4.9×10^{-16}	2.7×10^{-16}
10^{-3}	6.6×10^{-16}	2.7×10^{-16}
10^{-4}	5.4×10^{-16}	4.2×10^{-16}
10^{-5}	3.7×10^{-16}	3.1×10^{-16}
10^{-6}	5.8×10^{-16}	1.7×10^{-16}
10^{-7}	4.0×10^{-16}	2.9×10^{-16}
10^{-8}	4.9×10^{-16}	3.2×10^{-16}

TABLE 6.5
The residuals of the solutions in Example 3

defined for any $h \times h$ block matrix A.

We say that the block matrix A has *block displacement rank* r if r is the minimum integer such that there exist block vectors $\boldsymbol{u}^{(i)}$, $\boldsymbol{v}^{(i)}$, $i = 1, \ldots, r$, satisfying the equation $\Delta(A) = \sum_{i=1}^{r} \boldsymbol{u}^{(i)}\boldsymbol{v}^{(i)\mathrm{T}}$.

If $\Delta(A) = \sum_{i=1}^{r} \boldsymbol{u}^{(i)}\boldsymbol{v}^{(i)\mathrm{T}}$, then the matrix A can be represented in terms of block Toeplitz matrices defined by the block vectors $\boldsymbol{u}^{(i)}$, $\boldsymbol{v}^{(i)}$, $i = 1, \ldots, r$, and by its first block column, according the following formula (see [11]):

$$A = \mathcal{L}(A\boldsymbol{e}_1) - \sum_{i=1}^{r} \mathcal{L}(\boldsymbol{u}^{(i)})\mathcal{U}(\boldsymbol{v}^{(i)\mathrm{T}} Z^{\mathrm{T}}),$$

where $\mathcal{L}(\boldsymbol{w})$ ($\mathcal{U}(\boldsymbol{w}^{\mathrm{T}})$) denotes the $h \times h$ block lower (upper) triangular block Toeplitz matrix defined by its first block column $\boldsymbol{w}$ (row $\boldsymbol{w}^{\mathrm{T}}$). The block vectors $A\boldsymbol{e}_1$, $\boldsymbol{u}^{(i)}$, $\boldsymbol{v}^{(i)}$, $i = 1, \ldots, r$, will be called generators of the block Toeplitz-like matrix A.

From the above results, it follows that a matrix with small block displacement rank can be expressed by means of a sum of a few products of block lower triangular and block upper triangular block Toeplitz matrices. In particular, the product of one such matrix and a block vector can be reduced, by means of formula (3.2), to a few products of block Toeplitz matrices and a block vector. This computation can be performed by means of a fast algorithm based on the polynomial evaluation/interpolation at the roots of 1 with FFTs. More specifically, let $A = (A_{i-j+h-1})_{i,j=1,\ldots,h}$ be a block Toeplitz matrix, $\boldsymbol{b}$ a block vector with block entries $B_0, B_1, \ldots, B_{k-1}$, and $\boldsymbol{c} = A\boldsymbol{b}$, where $\boldsymbol{c}$ has block entries $C_0, C_1, \ldots, C_{k-1}$. Then, $\boldsymbol{c}$ can be efficiently computed by means of the following scheme [11, 7, 8]:

1. Evaluate the matrix polynomial $\alpha(z) = A_{h-1} + A_h z + \cdots + A_{2h-2} z^{h-1} + A_0 z^{h+1} +$

	Orig. Alg.	Shifted Alg.
Iter.	16	6
Res.	2.8×10^{-16}	6.3×10^{-16}

TABLE 6.6
The number of iterations and residuals of the solutions in Example 4

$\cdots + A_{h-2}z^{2h-1}$ at the $2h$ roots of 1 ω^j, $j = 0, \ldots, 2h-1$, where ω is a primitive $2h$-th root of 1, by means of m^2 DFT's of order $2h$, and obtain the matrices $\alpha(\omega^j)$, $j = 0, \ldots, 2h-1$.

2. Evaluate the matrix polynomial $\beta(z) = B_0 + B_1 z + \cdots + B_{h-1}z^{h-1}$ at the $2h$ roots of 1 ω^j, $j = 0, \ldots, 2h-1$, by means of m^2 DFT's of order $2h$, and obtain the matrices $\beta(\omega^j)$, $j = 0, \ldots, 2h-1$.
3. Compute the products $\gamma(\omega^j) = \alpha(\omega^j)\beta(\omega^j)$, $j = 0, \ldots, 2h-1$.
4. Interpolate $\gamma(\omega^j)$ by means of m^2 IDFT's of order $2h$ and obtain the coefficients $\gamma_0, \gamma_1, \ldots, \gamma_{2h-1}$ such that $\gamma(z) = \sum_{i=0}^{2h-1} \gamma_i z^i$ and set $C_i = \gamma_i$, $i = 0, \ldots, h-1$.

The computational cost of the above algorithm is $O(m^3 h + m^2 h \log h)$ arithmetic operations. Thus, the computation of the product of a block Toeplitz-like matrix and a block vector can be performed at the same cost.

Acknowledgments. The authors wish to thank the anonymous referees for the helpful suggestions which improved the presentation of this paper.

REFERENCES

[1] N. AKAR AND K. SOHRABY, *An invariant subspace approach in M/G/1 and G/M/1 type Markov chains*, Comm. Statist. Stochastic Models, 13 (1997), pp. 381–416.

[2] P. BENNER, R. BYERS, V. MEHRMANN, AND H. XU, *A unified deflating subspace approach for classes of polynomial and rational matrix equations*, Preprint SFB393/00-05, Technische Universität Chemnitz, Jan. 2000.

[3] D. A. BINI, L. GEMIGNANI, AND B. MEINI, *Factorization of analytic functions by means of Koenig's theorem and Toeplitz computations*, Numer. Math., 89 (2001), pp. 49–82.

[4] D. A. BINI, L. GEMIGNANI, AND B. MEINI, *Computations with infinite Toeplitz matrices and polynomials*, Linear Algebra Appl., 343/344 (2002), pp. 21–61. Special issue on structured and infinite systems of linear equations.

[5] D. A. BINI, L. GEMIGNANI, AND B. MEINI, *Solving certain matrix equations by means of Toeplitz computations: algorithms and applications*, in Fast algorithms for structured matrices: theory and applications, vol. 323 of Contemp. Math., Amer. Math. Soc., Providence, RI, 2003, pp. 151–167.

[6] D. A. BINI, G. LATOUCHE, AND B. MEINI, *Solving matrix polynomial equations arising in queueing problems*, Linear Algebra Appl., 340 (2002), pp. 225–244.

[7] D. A. BINI AND B. MEINI, *On the solution of a nonlinear matrix equation arising in queueing problems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 906–926.

[8] ———, *Improved cyclic reduction for solving queueing problems*, Numer. Algorithms, 15 (1997), pp. 57–74.

[9] ———, *Effective methods for solving banded Toeplitz systems*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 700–719.

[10] ———, *Non-skip-free M/G/1-type Markov chains and Laurent matrix power series*, Linear Algebra Appl., 386 (2004), pp. 187–206.

[11] D. A. BINI AND V. PAN, *Matrix and Polynomial Computations, Vol. 1: Fundamental Algorithms*, Birkhäuser, Boston, 1994.

[12] R. C. BUSBY AND W. FAIR, *Iterative solution of spectral operator polynomial equations and a related continued fraction*, J. Math. Anal. Appl., 50 (1975), pp. 113–134.

[13] B. L. BUZBEE, G. H. GOLUB, AND C. W. NIELSON, *On direct methods for solving Poisson's equation*, SIAM J. Numer. Anal., 7 (1970), pp. 627–656.

[14] J. E. DENNIS, JR., J. F. TRAUB, AND R. P. WEBER, *Algorithms for solvents of matrix polynomials*, SIAM J. Numer. Anal., 15 (1978), pp. 523–533.

[15] H. R. GAIL, S. L. HANTLER, AND B. A. TAYLOR, *Spectral analysis of $M/G/1$ and $G/M/1$ type Markov chains*, Adv. in Appl. Probab., 28 (1996), pp. 114–165.
[16] I. GOHBERG, P. LANCASTER, AND L. RODMAN, *Matrix polynomials*, Academic Press Inc. [Harcourt Brace Jovanovich Publishers], New York, 1982. Computer Science and Applied Mathematics.
[17] C. HE, B. MEINI, AND N. H. RHEE, *A shifted cyclic reduction algorithm for quasi-birth-death problems*, SIAM J. Matrix Anal. Appl., 23 (2001/02), pp. 673–691.
[18] G. HEINIG AND K. ROST, *Algebraic Methods for Toeplitz-like Matrices and Operators*, Akademie-Verlag, Berlin, and Birkhäuser, Boston, 1984.
[19] N. J. HIGHAM AND H.-M. KIM, *Numerical analysis of a quadratic matrix equation*, IMA J. Numer. Anal., 20 (2000), pp. 499–519.
[20] T. KAILATH, S. KUNG, AND M. MORF, *Displacement ranks of matrices and linear equations*, J. Math. Anal. Appl., 68 (1979), pp. 395–407.
[21] T. KAILATH AND A. H. SAYED, eds., *Fast reliable algorithms for matrices with structure*, SIAM, Philadelphia, 1999.
[22] G. LATOUCHE, *Algorithms for infinite Markov chains with repeating columns*, in Linear Algebra, Queueing Models and Markov Chains, C. Meyer and R. Plemmons, eds., Springer-Verlag, New York, 1993, pp. 231–265.
[23] ———, *Newton's iteration for non-linear equations in Markov chains*, IMA J. Numer. Anal., 14 (1994), pp. 583–598.
[24] G. LATOUCHE AND V. RAMASWAMI, *A logarithmic reduction algorithm for Quasi-Birth-Death processes*, J. Appl. Probab., 30 (1993), pp. 650–674.
[25] B. MEINI, *New convergence results on functional iteration techniques for the numerical solution of M/G/1 type Markov chains*, Numer. Math., 78 (1997), pp. 39–58.
[26] M. F. NEUTS, *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*, The Johns Hopkins University Press, Baltimore, MD, 1981.
[27] ———, *Structured Stochastic Matrices of M/G/1 Type and Their Applications*, Marcel Dekker, New York, 1989.
[28] V. RAMASWAMI, *Nonlinear matrix equations in applied probability - Solution techniques and open problems*, SIAM Rev., 30 (1988), pp. 256–263.
[29] G. W. STEWART AND J. G. SUN, *Matrix perturbation theory*, Computer Science and Scientific Computing, Academic Press Inc., Boston, MA, 1990.