AMO - Advanced Modeling and Optimization, Volume 12, Number 1, 2010

A Structure-Preserving Doubling Algorithm for Quadratic Matrix Equations arising form damped mass-spring system [*]

Bo Yu[†] [‡] and Ning Dong [‡]

Abstract. We are concerned with the quadratic matrix equation with nonsingular M-matrices arising from the damped mass-spring system. We propose a sufficient condition for the existence of the solvents to the equation. We also develop a nonsingular M-matrix structure-preserving doubling algorithm (MSD) to calculate the extreme solvents of the equation. Under appropriate conditions, we establish the global and quadratic convergence of the proposed method. Numerical experiments show that the proposed MSD algorithm outperforms Newton's method with exact line searches and Bernoulli's method.

Key words. quadratic matrix equation, extreme solvents, structure-preserving doubling algorithm, Newton's method, Bernoulli's method

AMS Subject Classification. 65F30; 65H10; 15A24

1. Introduction

We consider the quadratic matrix equation (QME)

$$Q(X) = AX^2 + BX + C = 0, \tag{1.1}$$

where A, B, C are all real matrices of sizes $n \times n$. Throughout, we assume that

$$\text{matrix } A \ \text{ is a nonnegative and nonsingular matrix,}$$

$$\text{matrices } B \ \text{ and } C \text{ are nonsingular M} - \text{matrices such that } B^{-1}C \geq 0.$$

We will introduce the concept of M-matrix and its properties in the latter part of this section.

The above conditions are motivated by a quadratic eigenvalue problem (QEP)

$$Q(\lambda)x = (\lambda^2 A + \lambda B + C)x \tag{1.2}$$

in a damped mass-spring system with following structure [18, 19]:

$$A = \text{diag}(m_1, \cdots, m_n),\ B = \tau\text{tridiag}(-1, 3, -1),\ C = \kappa\text{tridiag}(-1, 3, -1),$$

[*] This work was supported in part by the major project of the Ministry of Education of China granted 309023.

[†] College of Mathematics and Econometrics, Hunan University, Changsha, 410082, P. R. China.
E-mail: wenyubwenyub@yahoo.com.cn (B. Yu)

[‡] College of Science, Hunan University of Technology, Zhuzhou, 412008, P.R.China.
E-mail: dnsx216@tom.com (N.Dong)

AMO - Advanced Modeling and Optimization. ISSN: 1841-4311

where for each $i = 1, 2, \cdots, n$, m_i is the weight of the ith mass and τ (κ) is the damping (stiffness) constant. It is easy to see that the above structure is a special case of $Q(X)$.

A solution of QME (1.1) is called a solvent. The existence of solvents to (1.1) has been extensively studied. For details, see [2, 14, 11] and the references there in. Among all the solvents (if exists), two extreme solvents, the dominant and minimal solvents, are of great interests. Let the eigenvalues of QEP (1.2) $\{\lambda_i\}_{i=1}^{2n}$ be ordered by absolute values:

$$|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_{2n}|. \tag{1.3}$$

A solvent $X^{(1)}$ of $Q(X)$ is called a dominant solvent if the set of the eigenvalues is $\lambda(X^{(1)}) = \{\lambda_1, \cdots, \lambda_n\}$ and $|\lambda_n| > |\lambda_{n+1}|$. A solvent $X^{(2)}$ of $Q(X)$ is a minimal solvent if its eigenvalue set is $\lambda(X^{(2)}) = \{\lambda_{n+1}, \cdots, \lambda_{2n}\}$ and $|\lambda_n| > |\lambda_{n+1}|$.

Newton's method and basic fixed-point iterations are widely used to find the solvents (not necessary extreme solvents) of (1.1). Higham and Kim showed that Newton's method with exact line searches is globally and quadratically convergent [11]. However, solving the subproblem, a generalized Sylvester equation, of Newton's method requires $102n^3$ flops by generalized Schur decomposition or $52n^3$ flops by Hessenberg-triangular decomposition [3, 4, 5, 11]. It will become a huge cost when the dimension increases largely. Bernoulli's method, one of the fixed-point iterations, is regarded as a more effective algorithm than Newton's method for extreme solvents due to its low cost per iteration [12]. Nevertheless, its linear convergence rate results in too many iterations definitely. Recently, Chu et al. constructed a structure-preserving doubling (SD) algorithm for Riccati-type matrix equation [1]. Attractive properties of the SD algorithm, as pointed out by Lin and Xu [15], include its quadratic convergence rate, low computational cost per iteation and nice numerical reliability. Guo et al. developed an SD algorithm to nonsymmetric algebraic Riccati equation (NARE) where the four coefficient matrices form a nonsingular M-matrix [10]. Guo et al. considered the case of irreducible singular M-matrix formed by four coefficient matrices. In this case, the convergence rate of SD algorithm was shown to be linear with 1/2 [8].

In this paper, we first propose a sufficient condition for the existence of the solvents to QME (1.1) with nonsingular coefficient matrices. This condition, based on a fixed-point iterations, is similar to Eisenfeld's condition. In order to compute the extreme solvents to (1.1), we then develop a M-matrix structure-preserving doubling (MSD) algorithm. An advantage of the MSD algorithm is that it can preserve a nonsingular M-matrix structure during iterations. This algorithm is highly efficient in that at each iteration, only one LU-factorization and several matrix multiplications are required. Under mild conditions, we establish the global and quadratic convergence of the method. Similar to the arguments in [7, 8, 9] for nonsymmetric algebraic Riccati equations, we construct an example to illustrate the linear convergence rate of MSD (with 1/2) when the solvents contains a unit eigenvalue (in module). Our numerical experiments show that the MSD algorithm outperforms Newton's method with exact line searches in [11] and the Bernoulli's method in [12].

In what follows, we introduce the concept of M-matrix and give its interesting properties. For matrices A, $B \in \mathbb{R}^{n\times n}$, we write $A \geq B (A > B)$ if $a_{ij} \geq b_{ij} (a_{ij} > b_{ij})$ for all i, j. A real square matrix A is called a Z-matrix if all its off-diagonal elements are nonpositive. It is clear that any Z-matrix A can be written as $sI - B$ with $B \geq 0$. A Z-matrix $A = sI - B$ with $B \geq 0$ is called an M-matrix if $s \geq \rho(B)$, where $\rho(\cdot)$ denotes the spectral radius. It is called a singular M-matrix if $s = \rho(B)$ and a nonsingular M-matrix if $s > \rho(B)$. Given a square matrix A, we will denote by $\lambda(A)$ the set of eigenvalues of A, and $\|A\|$ the Euclid norm of A.

The following lemma comes from [21] which shows some interesting properties of the nonsingular M-matrix. It should be pointed out that the M-matrix defined in [21] is the nonsingular M-matrix in this paper.

Lemma 1.1 *For a Z-matrix A, the following statements are equivalent:*
(a) A is a nonsingular M-matrix.
(b) A is nonsingular and satisfies $A^{-1} \geq 0$.
(c) $Av > 0$ for some vector $v > 0$.
(d) All eigenvalues of A have positive real parts.

The rest of this paper is organized as follows. We give a sufficient condition for the existence of the solvents of QME (1.1) in the next section. In Section 3, we develop a nonsingular M-matrix structure-preserving doubling algorithm to find the extreme solvents to QME (1.1). In Section 4, we establish the global and quadratic convergence of the method. We do some numerical experiments to test the proposed method and compare its performance with Newton's method and Bernoulli's method in Section 5. We conclude the paper by discussion in Section 6.

2. A sufficient condition for the existence of solvents

The existence of the solvents to (1.1) has been studied by some scholars. By the use of contraction mapping principle, Eisenfeld [2] showed that if A, B and C are nonsingular and

$$4\|B^{-1}C\|\|B^{-1}A\| < 1, \tag{2.1}$$

then at least two solvents exist [2]. A similar but more restrictive condition for the existence of solvents was derived by McFarland [16].

In this section, we also show, by a fixed-point iterative process, that condition (2.1) is sufficient for the existence of the solvents to (1.1). Consider the iterative process

$$Y_{k+1} = -B - CY_k^{-1}A, \quad k = 0, 1, \cdots \tag{2.2}$$

with initial matrix $Y_0 = -B$. This iterative scheme was also considered in [7, 17]. Here we use it to derive a similar sufficient condition to [2] for the existence of the solvents to (1.1).

Theorem 2.1 *If A, B in QME (1.1) are nonsingular and condition (2.1) is satisfied, then (1.1) has at least a solvent.*

Proof. We first prove the inequality $\|B^{-1}C\|\|Y_k^{-1}A\| < 1/2$ by induction. From the fixed-point iteration (2.2) and condition (2.1), it is clear that

$$\|B^{-1}C\|\|Y_0^{-1}A\| < 1/2.$$

We assume that $\|B^{-1}C\|\|Y_{m-1}^{-1}A\| < 1/2$ holds for all $m \leq k-1$. Then we get from (2.2)

$$\begin{aligned}\|B^{-1}C\|\|Y_k^{-1}A\| &\leq \|B^{-1}C\|\|B^{-1}A\|\|(I+B^{-1}CY_{k-1}^{-1}A)^{-1}\| \\ &\leq \frac{\|B^{-1}C\|\|B^{-1}A\|}{1-\|B^{-1}C\|\|Y_{k-1}^{-1}A\|} \\ &\leq \frac{1/4}{1-\frac{1/4}{1-\|B^{-1}C\|\|Y_{k-2}^{-1}A\|}}\cdots, \end{aligned} \tag{2.3}$$

where the first inequality follows from (2.2) and second inequality follows from the perturbation lemma $\|(I-A)^{-1}\| \leq 1/(1-\|A\|)$ when $\|A\| < 1$ (see [20] for example). By Worpitzky theorem of continued fraction (see [13] for example), we have from (2.3) $\|B^{-1}C\|\|Y_k^{-1}A\| < 1/2$. This results in

$$\|Y_k^{-1}A\| = \|(I+B^{-1}CY_{k-1}^{-1}A)^{-1}B^{-1}A\| \leq \frac{\|B^{-1}A\|}{1-\|B^{-1}C\|\|Y_{k-1}^{-1}A\|} < 2\|B^{-1}A\| \tag{2.4}$$

and

$$\|Y_k^{-1}B\| = \|(I+B^{-1}CY_{k-1}^{-1}A)^{-1}\| \leq \frac{1}{1-\|B^{-1}C\|\|Y_{k-1}^{-1}A\|} < 2.$$

In particular, $\{\|Y_k^{-1}\|\}$ is bounded above and hence $\{\|Y_k\|\}$ is bounded away from zero. Therefore, we get from (2.2) and (2.3)

$$\begin{aligned}\|B^{-1}(Y_{k+1}-Y_k)\| &= \|B^{-1}C(Y_k^{-1}-Y_{k-1}^{-1})A\| \\ &\leq \|B^{-1}C\|\|Y_k^{-1}B\|\|B^{-1}(Y_k-Y_{k-1})\|\|Y_{k-1}^{-1}A\| \\ &< \theta\|B^{-1}(Y_k-Y_{k-1})\|\end{aligned}$$

where $\theta = 4\|B^{-1}C\|\|B^{-1}A\| < 1$. The last inequality together with the boundedness of $\{\|Y_k^{-1}\|\}$ shows that $\{Y_k\}$ has a nonsingular limit. Let $Y^* = \lim_{k\to\infty} Y_k$. Then Y^* is a solvent of $Y = -B - CY^{-1}A$. Consequently, $A^{-1}Y^*$ is a solvent of QME (1.1) by some transformations . □

The proof of Theorem 2.1 has found a solvent $A^{-1}Y^*$ of (1.1) where Y^* is the limit of $\{Y_k\}$ determined by (2.2). In what follows, we find another solvent of (1.1) in a similar way.

Consider the iterative process

$$Y_{k+1} = -B - AY_k^{-1}C, \quad k = 0, 1, \cdots \tag{2.5}$$

with $Y_0 = -B$. Similar to the proof of Theorem 2.1, we can derive another solvent $\hat{Y}^{-1}C$ of (1.1), where $\hat{Y}$ is the limit of (2.5). Therefore, we have the following theorem.

Theorem 2.2 *If A, B and C in QME (1.1) are nonsingular, then condition (2.1) implies that QME (1.1) has at least two solvents $A^{-1}Y^*$ and $\hat{Y}^{-1}C$, where Y^* and $\hat{Y}$ are the limits of $\{Y_k\}$ generated by (2.2) and (2.5) with $Y_0 = -B$, respectively.*

We are particularly interested in the dominant and minimal solvents of (1.1). The latter parts of this paper are dedicated to the numerical method for the computation of these solvents of (1.1).

We will develop a M-matrix structure-preserving doubling (MSD) algorithm and established its convergence. To this end, let us investigate some nice properties of the solvents $A^{-1}Y^*$ and $\hat{Y}^{-1}C$ specified in Theorem 2.2. Suppose that matrices A, B and C are nonsingular and satisfied the condition

$$\|B^{-1}C\| + \|B^{-1}A\| < 1. \tag{2.6}$$

We are going to show by induction that inequality $\|Y_k^{-1}A\| < \frac{\|B^{-1}A\|}{1-\|B^{-1}C\|}$ holds for all k. Clearly, condition (2.6) implies $\|Y_0^{-1}A\| = \|B^{-1}A\| < \frac{\|B^{-1}A\|}{1-\|B^{-1}C\|} < 1$. Suppose $\|Y_{k-1}^{-1}A\| < \frac{\|B^{-1}A\|}{1-\|B^{-1}C\|} < 1$. We get from (2.2) and condition (2.6)

$$\|Y_k^{-1}A\| \leq \frac{\|B^{-1}A\|}{1-\|B^{-1}C\|\|Y_{k-1}^{-1}A\|} < \frac{\|B^{-1}A\|}{1-\|B^{-1}C\|} < 1. \tag{2.7}$$

So we conclude that $\|Y_k^{-1}A\| < \frac{\|B^{-1}A\|}{1-\|B^{-1}C\|}$ for all k and hence $\|Y^{*-1}A\| < 1$. This shows that all eigenvalues (in module) of $(Y^*)^{-1}A$ are less than one, and hence the eigenvalues of its inverse are all greater than one. Similarly, we can show that the eigenvalues of $\hat{Y}^{-1}C$ are all less than one. This means that $A^{-1}Y^*$ and $\hat{Y}^{-1}C$ are the dominant solvent and the minimal solvent to QME (1.1), respectively.

3. MSD algorithm for QME

In this section, we propose an M-matrix structure-preserving doubling algorithm which we call the MSD algorithm. We first introduce some definitions of transformation and symplectic forms.

Let $H - \lambda J \in \mathbb{R}^{2n\times 2n}$ be a matrix pencil and

$$\mathcal{N}(H,J) = \left\{[L_*, U_*] : L_*, U_* \in \mathbb{R}^{2n\times 2n}, \text{rank}[L_*, U_*] = 2n, [U_*, L_*]\begin{bmatrix} H \\ -J \end{bmatrix} = 0\right\}.$$

It is clear that $\mathcal{N}(H,J) \neq \varnothing$ since $\text{rank}\begin{pmatrix} H \\ -J \end{pmatrix} \leq 2n$. For each $[L_*, U_*] \in \mathcal{N}(H,J)$, we define

$$\hat{H} = L_* H, \quad \hat{J} = U_* J.$$

The transformation

$$H - \lambda J \longrightarrow \hat{H} - \lambda\hat{J}$$

is called a *doubling transformation.*

A subspace $\mathcal{W}$ of $\mathbb{R}^{2n\times 2n}$ is called a *generalized eigenspace* of a pencil $H - \lambda J \in \mathbb{R}^{2n\times 2n}$, if $\mathcal{W}$ is spanned by the columns of $W \in \mathbb{R}^{2n\times n}$, where W has full rank columns and satisfies $HW = JWR_1$ with some $R_1 \in \mathbb{R}^{n\times n}$. The following theorem, proved in [15], shows that the doubling transformation can preserve the generalized eigenspace $\mathcal{W}$ and square the eigenvalues of R_1.

Theorem 3.1 *Let the matrix pencil $H - \lambda J \in \mathbb{R}^{2n\times 2n}$ satisfy*

$$HW = JWR_1 \quad or \quad JV = HVR_2,$$

where $W, V \in \mathbb{R}^{2n\times n}$ and $R_1, R_2 \in \mathbb{R}^{n\times n}$. Let $\hat{H} - \lambda\hat{L}$ be a doubling transformation of $H - \lambda J$. Then we have

$$\hat{H}W = \hat{J}WR_1^2 \quad or \quad \hat{J}V = \hat{H}VR_2^2.$$

Different symplectic structures of matrix pencil $H - \lambda J$ have been constructed for various nonlinear matrix equations. For example, Lin and Xu [15] defined a first (second) standard symplectic form for discreet-time algebraic Riccati equation (nonlinear matrix equation with plus or minus sign). Guo et al. [10] gave another generalized standard symplectic form for nonsymmetric algebraic Riccati equation. Based on these symplectic forms, they proposed an SD algorithm and established its convergence. We are going to develop an MSD algorithm to solve the quadratic matrix equation (1.1). To this end, we define a new symplectic form with a nonsingular M-matrix as follows.

A matrix pencil $H - \lambda J$ is called in *Composed M-matrix symlectic form* (CMSF) if

$$H = \begin{bmatrix} 0 & E \\ F & -G \end{bmatrix}, \quad J = \begin{bmatrix} I & S \\ 0 & T \end{bmatrix}, \tag{3.1}$$

where $\pm E$, $-F$, $\pm T$ and S are nonnegative matrices with dimension $n\times n$, and G is a nonsingular M-matrix.

In what follows, we propose a doubling algorithm which can preserve the above CMSF. Let X be the minimal solvent of QME (1.1). It is clear that equation (1.1) can be rewritten as

$$\begin{bmatrix} 0 & I \\ -C & -B \end{bmatrix} \begin{bmatrix} I \\ X \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & A \end{bmatrix} \begin{bmatrix} I \\ X \end{bmatrix} X.$$

A direct left multiplication of $\begin{bmatrix} I & 0 \\ 0 & B^{-1} \end{bmatrix}$ in the above equation gives rise to

$$H \begin{bmatrix} I \\ X \end{bmatrix} = J \begin{bmatrix} I \\ X \end{bmatrix} X, \tag{3.2}$$

where

$$H = \begin{bmatrix} 0 & I \\ -B^{-1}C & -I \end{bmatrix} \quad \text{and} \quad J = \begin{bmatrix} I & 0 \\ 0 & B^{-1}A \end{bmatrix}.$$

Set $E = I$, $F = -B^{-1}C$, $G = I$, $S = 0$ and $T = B^{-1}A$. It is easy to see that $H - \lambda J$ is in CMSF and that matrix $G + FS$ is a nonnegative M-matrix. Consider the triangular structure of the above H and J. Following the techniques in [1] and [15], we select

$$L_* = \begin{bmatrix} E(G+FS)^{-1}F & 0 \\ -T(G+FS)^{-1}F & I \end{bmatrix} \quad \text{and} \quad U_* = \begin{bmatrix} I & E(G+FS)^{-1} \\ 0 & -T(G+FS)^{-1} \end{bmatrix} \tag{3.3}$$

such that

$$L_*J = U_*H.$$

It is obvious that $[L_*, U_*] \in \mathcal{N}[H, J]$. Then we get a doubling transformation

$$\hat{H} = L_*H = \begin{bmatrix} 0 & \hat{E} \\ F & -\hat{G} \end{bmatrix}, \quad \hat{J} = U_*J = \begin{bmatrix} I & \hat{S} \\ 0 & -\hat{T} \end{bmatrix},$$

where

$$\hat{E} = E(G+FS)^{-1}FE, \qquad \hat{G} = G + T(G+FS)^{-1}FE,$$
$$\hat{S} = S + E(G+FS)^{-1}T, \quad \hat{T} = -T(G+FS)^{-1}T.$$

Based on the above argument, we propose the MSDA method as follows.

Algorithm 3.1 M-matrix Structure-Preserving Doubling Algorithm (MSDA):

Step 1: *Given matrix $F \in \mathbb{R}^{n\times n}$. Let $E_0 = I$, $F = F_0 = -B^{-1}C$,$G_0 = I$, $S_0 = 0$ and $T_0 = B^{-1}A$. Let $k = 0$.*

Step 2: *For $k \geq 0$, until convergence, do*

$$\begin{aligned} E_{k+1} &= E_k(G_k + FS_k)^{-1}FE_k, \\ G_{k+1} &= G_k + T_k(G_k + FS_k)^{-1}FE_k, \\ S_{k+1} &= S_k + E_k(G_k + FS_k)^{-1}T_k, \\ T_{k+1} &= -T_k(G_k + FS_k)^{-1}T_k. \end{aligned}$$

It is not difficult to see that at each iteration, the above algorithm only requires one *LU*-factorization and several matrix multiplications. The computation cost is $\frac{50}{3}n^3$ flops, one sixth of that by generalized Schur decomposition in Newton's method with exact line searches.

The following theorem shows that the MSDA is well defined and can preserve the structure of CMSF. In the next section we shall show that under suitable conditions the matrix sequence $\{G_k\}$ ($\{S_k\}$) is monotonically decreasing (increasing) and quadratically convergent.

Theorem 3.2 *Let matrix sequences $\{E_k\}$, $\{G_k\}$, $\{S_k\}$, $\{T_k\}$ be generated by the MSD algorithm and $V_k = T_k(G_k + FS_k)^{-1}FE_k$. If condition (2.1) holds, then we have for $k \geq 1$*
(a) $E_k \leq 0$, $\|E_k\| \leq 2^{2^k-k-1}\|F\|^{2^k-1}$;
(b) $T_k \leq 0$, $\|T_k\| \leq 2^{2^k-k-1}\|T_0\|^{2^k}$;
(c) $S_k \geq 0$, $\|S_k\| \leq (2 - 2^{-(k-1)})\|T_0\|$;
(d) $G_k + FS_k$ are nonsingular M-matrices and $\|(G_k + FS_k)^{-1}\| \leq 2^k$;
(e) $V_k \leq 0$, $\|V_k\| \leq 2^{-(k+2)}$.

Proof. We prove the theorem by induction. It is clear that $E_1 = F = -B^{-1}C \leq 0$, $T_1 = -T_0^2 = -(B^{-1}A)^2 \leq 0$, $S_1 = T_0 = B^{-1}A \geq 0$. Since $\|V_0 + FS_1\| \leq 1/4 < 1$, $G_1 + FS_1$ is a nonsingular M-matrix and

$$\|(G_1 + FS_1)^{-1}\| = \|(I + V_0 + FS_1)^{-1}\| \leq \frac{1}{1 - 2\|B^{-1}C\|\|B^{-1}A\|} \leq 2.$$

It then follows that $V_1 = T_1(G_1 + FS_1)^{-1}FE_1 \leq 0$ and

$$\|V_1\| \leq \|T_1\|\|(G_1 + FS_1)^{-1}\|\|F\|\|E_1\| \leq 2^{-3}.$$

This proves the theorem for $k = 1$. Suppose that $(a) - (e)$ are true for all positive integers less than or equal to k. We are going to show that they are true for $k+1$ too. It follows from Step

2 of MSD algorithm that $E_{k+1} \leq 0$, $T_{k+1} \leq 0$ and

$$\begin{aligned} E_{k+1} &= E_k(G_k + FS_k)^{-1}FE_k \leq 0, \\ T_{k+1} &= -T_k(G_k + FS_k)^{-1}T_k \leq 0, \\ S_{k+1} &= S_k + E_k(G_k + FS_k)^{-1}T_k \geq 0. \end{aligned} \tag{3.4}$$

We also have by induction

$$\begin{aligned} \|E_{k+1}\| &\leq \|E_k\|\|(G_k + FS_k)^{-1}\|\|F\|\|E_k\| \leq 2^{2^{k+1}-(k+1)-1}\|F\|^{2^{k+1}-1}, \\ \|T_{k+1}\| &\leq \|T_k\|\|(G_k + FS_k)^{-1}\|\|T_k\| \leq 2^{2^{k+1}-(k+1)-1}\|T_0\|^{2^{k+1}}, \\ \|S_{k+1}\| &\leq \|S_k\| + \|E_k\|\|(G_k + FS_k)^{-1}\|\|T_k\| \\ &\leq \|T_0\|(2 - \frac{1}{2^{k-1}} + \frac{2^{2^{k+1}-(k+1)-1}}{4^{2^k-1}}) = (2 - 2^{-k})\|T_0\|. \end{aligned}$$

By Step 2 of the MSD algorithm again, we have

$$G_{k+1} - G_k = V_k \leq 0. \tag{3.5}$$

Denote by $W_{k+1} = \sum_{0\leq l\leq k} V_l + FS_{k+1}$. It then follows that $G_{k+1} + FS_{k+1} = I + W_{k+1}$. Since

$$\|W_{k+1}\| \leq \sum_{0\leq l\leq k} \|V_l\| + \|F\|\|S_{k+1}\| \leq 1 - 2^{-(k+1)} < 1,$$

we claim that $G_{k+1} + FS_{k+1}$ is a nonsingular M-matrix with norm

$$\|(G_{k+1} + FS_{k+1})^{-1}\| \leq \frac{1}{1 - \|W_{k+1}\|} \leq 2^{k+1}.$$

Finally, Step 2 of MSD algorithm implies

$$V_{k+1} = T_{k+1}(G_{k+1} + FS_{k+1})^{-1}FE_{k+1} \leq 0$$

with norm

$$\|V_{k+1}\| \leq \|T_{k+1}\|\|(G_{k+1} + FS_{k+1})^{-1}\|\|F\|\|E_{k+1}\| \leq 2^{-((k+1)+2)}.$$

The proof is complete. □

The above theorem yields the following corollary immediately, which shows that the structure of CMSF can be preserved in iterations.

Corollary 3.1 *Let the initial matrix pencil $H - \lambda J$ be in CMSF form. If (2.1) holds, then we can choose a matrix $[L_*, U_*]$ with form (3.3) in $\mathcal{N}(H, J)$ such that its corresponding doubling transformation $\hat{H} - \lambda\hat{J}$ is still in CMSF.*

4. Convergence for the MSD algorithm

In this section, we establish the convergence of the MSD algorithm. Let X^* and Z^* be solvents of QME (1.1) and its dual equation

$$R(Z) = CZ^2 + BZ + A = 0, \tag{4.1}$$

respectively. As shown in Section 2, under the condition (2.6), there exists $X^* = (\hat{Y})^{-1}C$ and $Z^* = (Y^*)^{-1}A$ such that $\rho(X^*) < 1$ and $\rho(Z^*) < 1$. Moreover, X^* and Z^* are the minimal solvent and the inverse of dominant solvent for QME (1.1), respectively.

The following theorem establishes the global convergence of the MSD algorithm. Moreover, it shows that matrix sequences $\{S_k\}$ and $\{G_k\}$ converge monotonically.

Theorem 4.1 *Let matrix sequences $\{E_k\}$, $\{G_k\}$, $\{S_k\}$, $\{T_k\}$ be generated by the MSD algorithm. Assume condition (2.6) holds. Let X^* and Z^* be the minimal solvents of (1.1) and its dual (4.1),respectively. Then the following statements hold.*
(a) $E_k = (I + S_k X^*)(X^*)^{2^k-1}$ and converges to 0.
(b) $T_k = (FZ^* - G_k)(Z^*)^{2^k}$ and converges to 0.
(c) The sequence $\{S_k\}_{k\geq 0}$ is monotonically increasing and converges to $-Z^*$.
(d) The sequence $\{G_k\}_{k\geq 0}$ is monotonically decreasing and converges to $I + B^{-1}AX^*$.

Proof. For each k, we let

$$H_k = \begin{bmatrix} 0 & E_k \\ F & -G_k \end{bmatrix}, \quad J_k = \begin{bmatrix} I & S_k \\ 0 & T_k \end{bmatrix}.$$

QME (1.1) and its dual equation (4.1) can be respectively rewritten as

$$H_0 \begin{bmatrix} I \\ X^* \end{bmatrix} = J_0 \begin{bmatrix} I \\ X^* \end{bmatrix} X^* \quad \text{and} \quad J_0 \begin{bmatrix} Z^* \\ I \end{bmatrix} = H_0 \begin{bmatrix} Z^* \\ I \end{bmatrix} Z^*.$$

Since the matrix pencil $H_k - \lambda J_k$ in the MSD algorithm is a doubling transformation of $H_{k-1} - \lambda J_{k-1}$, an application of Theorem 3.1 gives rise to

$$H_k \begin{bmatrix} I \\ X^* \end{bmatrix} = J_k \begin{bmatrix} I \\ X^* \end{bmatrix} (X^*)^{2^k} \quad \text{and} \quad J_k \begin{bmatrix} Z^* \\ I \end{bmatrix} = H_k \begin{bmatrix} Z^* \\ I \end{bmatrix} (Z^*)^{2^k}.$$

By direct computation, we obtain

$$E_k X^* = (I + S_k X^*)(X^*)^{2^k}, \tag{4.2}$$
$$F - G_k X^* = T_k (X^*)^{2^k+1}, \tag{4.3}$$
$$Z^* + S_k = E_k (Z^*)^{2^k}, \tag{4.4}$$
$$T_k = (FZ^* - G_k)(Z^*)^{2^k}. \tag{4.5}$$

It follows from Step 2 of MSD algorithm and Theorem 3.2 (c) that the matrix sequence $\{S_k\}$ is monotonically increasing and bounded above. So we get from (4.2) and (4.4) that the matrix sequence $\{E_k\}$ converges to zero and hence $\{S_k\}$ has the limit $-Z^*$ since $\rho(X^*) < 1$ and $\rho(Z^*) < 1$.

We turn to the convergence of $\{G_k\}$ and $\{T_k\}$. Inequality (3.5) means that $\{G_k\}$ is monotonically decreasing. Since

$$\|G_k\| = \|I + \sum_{i=0}^{k-1} V_i\| \leq 1 + (2^{-1} - 2^{-(k+2)}) < 3/2,$$

the sequence $\{G_k\}$ has a limit. It then follows from (4.5) that $\{T_k\}$ converges to zero. The fact $\rho(X^*) < 1$ together with equality (4.3) implies

$$\lim_{k\to\infty} G_k = F(X^*)^{-1} = I + B^{-1}AX^*.$$

The proof is complete. □

Theorem 4.1 shows that sequence $\{S_k^{-1}\}$ converges to the dominant solvent of (1.1) while sequence $\{A^{-1}B(G_{k+1} - I)\}$ converges to the minimal solvent of (1.1). In addition, conclusions (a) and (b) in the theorem show the quadratic convergence of the MSD algorithm.

Remark 4.1. We consider the extreme case where the equality holds in (2.6). In this case the eigenvalues (in module) of the extreme solvents may equal one. In a way similar to the proof of Theorem 3.2, it is not difficult to show that the sequence $\{G_k\}$ is still monotonically decreasing and bounded below and hence has a limit. However, as in the SD method [8], the convergence rate of MSD algorithm will reduce to linear. Here we give an example to show this fact. Let $A = \frac{1}{2}I$, $B = I$, and $C = \frac{1}{2}I$ such that $\|B^{-1}C\| + \|B^{-1}A\| = 1$. Then $X^* = -I$ is a solvent of QME (1.1). It is easy to verify that $G_{k+1} - G_k = \frac{1}{2}(G_k - G_{k-1})$, $S_{k+1} = \frac{1}{2}S_k$, $E_{k+1} = \frac{1}{2}E_k$, $T_{k+1} = \frac{1}{2}T_k$. Consequently, the sequence $\{G_k\}$ is monotonically decreasing and has a limit $G^* = \frac{1}{2}I$. However, the convergence rate is linear with factor 1/2.

Remark 4.2. In the proof of Theorem 4.1, the monotone convergence of $\{S_k\}$ and $\{G_k\}$ requires the condition $B^{-1}C \geq 0$. Nevertheless, our numerical experiments in the next section seem to show that this restriction might be unnecessary.

5. Numerical experiments

The purpose of this section is to show the effectiveness of the MSD algorithm 3.1 in computation. We tested the proposed algorithm and compare the performance of the MSD algorithm with the well-known Newton's method with exact line searches and the Bernoulli method. Our experiments were done in Matlab 7.0 on a PC with 2.13GHz Intel Celeron D processor, which has unit roundoff $u = 2^{-53} \approx 1.1 \times 10^{-16}$. We first give some implementation details of Newton's method with exact line searches (NME), Bernoulli's method (BM) and MSD algorithm.

For Newton's method, the default starting matrix is, as in [11],

$$X_0 = (\frac{\|B\|_F + (\|B\|_F^2 + 4\|A\|_F\|C\|_F)^{\frac{1}{2}}}{2\|A\|_F})I_n,$$

which was designed to have norm roughly the same order of magnitude as a solvent. The termination criterion is that the relative residual $\mu(X_k)$ satisfies

$$\mu(X_k) \triangleq \frac{\|fl(Q(X_k))\|_F}{\|A\|_F\|X_k\|_F^2 + \|B\|_F\|X_k\|_F + \|C\|_F} \leq nu,$$

where n is the dimension of the problem. The next iterate X_{k+1} is obtained by solving the generalized Sylvester system

$$\begin{cases} AE_kX_k + (AX_k + B)E_k = -Q(X_k), \\ X_{k+1} = X_k + tE_k \end{cases}$$

with generalized Schur decomposition, where t is the step length determined by the exact line search. When Bernoulli's method is implemented to find the dominant solvent, X_{k+1} is a solution of following system of linear equations.

$$(AX_{k+1} + B)X_k + C = 0, \quad k = 0, 1, \cdots$$

starting from $X_0 = -A^{-1}B$, while for the minimal solvent, X_{k+1} is a solution of the system of linear equations

$$(AX_k + B)X_{k+1} + C = 0$$

with initial matrix $X_0 = 0$, respectively. Although it is a fixed-point iterations with linear convergence rate, Bernoulli's method has been observed to possess better numerical performance in CPU times than Newton's method [12]. The stopping criterion in BM and MSD algorithm is, as in [12],

$$\frac{\|X_{k+1} - X_k\|_1}{\|X_k\|_1} \leq nu.$$

We tested the three methods on three problems with various dimensions.

Example 5.1 Consider the QME (1.1) with

$$B = \text{tridiag}(-I, T, -I) \in \mathbb{R}^{m^2 \times m^2},$$

where

$$T = \text{tridiag}(-1, 4 + \frac{150}{(m+1)^2}, -1) \in \mathbb{R}^{m \times m}$$

is a tridiagonal matrix. Let $n = m^2$ and

$$A = \frac{1}{a}\text{tridiag}(1, 2, 1) \in \mathbb{R}^{n \times n},\ C = \frac{1}{b}\text{tridiag}(-1, 4 + \frac{150}{(m+1)^2}, -1) \in \mathbb{R}^{n \times n},$$

where a and b are positive constants such that $\|B^{-1}C\|_2 + \|B^{-1}A\|_2 < 1$.

This problem comes from [10] with a little modification. The numerical results were listed in Tables 5.1, 5.2 with $a = 1/20$, $b = 1/15$, $m = 5,\ 10,\ 15,\ 20$ and in Tables 5.3, 5.4 with $m = 16$, $a = 1/8,\ 1/12,\ 1/120$, $b = 1/5,\ 1/12,\ 1/120$. Since it only converges to a minimal solvent with default initial matrix X_0, Newton's method is omitted in Tables 5.2 and 5.4. In the tables, "Iter" stands for the number of iterations used, while "CPU" denotes the CPU time used in seconds. We see from Tables 5.1-5.4, that in most cases the MSDA used least number of iterations and CPU time.

Table 5.1 *Results for minimal solvent with $a = 1/20$ and $b = 1/15$*

	m=5		m=10		m=15		m=20	
	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)
MSDA	4	0.1406	5	0.4219	5	2.9844	7	23.5156
BM	8	0.2969	10	0.6094	15	5.1563	30	58.9688
NME	6	1.5594	7	7.1719	8	63.1563	9	529.1250

Table 5.2 *Results for dominant solvent with $a = 1/20$ and $b = 1/15$*

	m=5		m=10		m=15		m=20	
	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)
MSDA	4	0.1406	5	0.5156	5	2.8750	7	23.6563
BM	5	0.2813	7	0.6255	8	3.3806	22	42.8438

Table 5.3 *Results for minimal solvent with $m = 16$*

	a=b=1/12		a=1/8, b=1/120		a=1/120, b=1/5		a=b=1/120	
	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)
MSDA	7	4.1563	5	4.3750	5	4.5156	4	3.2813
BM	30	12.8750	11	5.6094	12	5.5625	7	3.5000
NME	9	231.4844	8	209.1250	8	208.5469	8	209.2969

Table 5.4 *Results for dominant solvent with $m = 16$*

	a=b=1/12		a=1/8, b=1/120		a=1/120, b=1/5		a=b=1/120	
	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)
MSDA	7	3.1563	5	4.0906	5	4.2063	4	3.5469
BM	22	9.9063	7	4.3781	8	4.6875	4	2.4531

Example 5.2 We consider the QME (1.1) arising from a damped mass-spring system (see [18] or [12] for details) by choosing the masses, damping and stiffness constants to give an $n \times n$ problem with $A = I, B = \text{tridiag}(-10, 30, -10)$ except $B(1,1) = B(n,n) = 20$, and $C = \text{tridiag}(-5, 15, -5)$.

Since $\|B^{-1}C\|_2 + \|B^{-1}A\|_2 = 0.9248 < 1$, all the eigenvalues (in module) of the extreme solvents are less than one. We took the dimension n of problem varying from 100 to 450 to test the number of iterations and CPU times used for different algorithms. Only the results about the minimal solvent were listed in Table 5.5 for the purpose of comparison. Since Newton's method did not converge to the dominant solvent with default initial matrix X_0, we omitted the results. We can see from Table 5.5 that the number of iterations used for the MSDA was less than Newton's method and was about 1/3 as the Bernoulli's method. In addition, the CPU time used by the MSDA was less than Bernoulli's method did and was at most 1/15 as that used by Newton's method.

TABLE 5.5 *Results for minimal solvent with dimension from $n = 100$ to $n = 450$*

	n=100		n=150		m=n=200		n=250	
	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)
MSDA	5	0.1094	5	0.4531	5	0.9219	5	2.4063
BM	15	0.5313	15	1.3125	15	2.7657	15	5.3750
NME	6	4.0469	6	15.2657	6	43.3438	6	88.6407
	n=300		n=350		n=400		n=450	
	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)
MSDA	5	5.4688	5	9.0313	5	15.5313	5	38.7656
BM	15	10.9375	15	18.3438	14	27.5938	14	47.7969
NME	6	179.6563	6	273.8594	6	442.9063	6	575.5781

Finally, we tested the three algorithms on the equation in which matrices A, B and C are dense matrices.

Example 5.3 Consider the QME (1.1). Let $(A)_{ij}$, $(B)_{ij}$ and $(C)_{ij}$ be the (i,j)-th element of matrices A, B and C specified by

$$(A)_{ij} = \left\{ \begin{array}{ll} 2/\alpha & i+j=n+1 \\ 1/\alpha & \text{others} \end{array} \right. , (B)_{ij} = \left\{ \begin{array}{ll} 15 & i=j \\ -3 & i+1=j \\ -3 & i=j+n-2 \\ -1 & \text{others} \end{array} \right. , (C)_{ij} = \left\{ \begin{array}{ll} 15/\beta & i=j \\ -1/\beta & \text{others} \end{array} \right. ,$$

where $\alpha = 1/15$ and $\beta = 1/2$. It is not difficult to show that inequality $\|B^{-1}C\|_2 + \|B^{-1}A\|_2 < 1$ is satisfied.

We tested the algorithms on the problem with different dimensions $n = 10, 100,$ 200 and 300. The condition $B^{-1}C \geq 0$ hold for $n = 10$ while does not hold for $n =$ 100, 200 and 300. The results were listed in Table 5.6. We see from the table that even when condition $B^{-1}C \geq 0$ is not satisfied, the MSD and Bernoulli's algorithms terminated at extreme solvents. Moreover, the MSD algorithm outperformed the Bernoulli's algorithm in the number of iterations as well as CPU time used. On the other hand, Newton's method terminated at non-extreme solvents and used CPU time at least 10 times as that by the MSD algorithm.

Table 5.6 *Results for solvents with different algorithms*

	n=10		n=100		n=200		n=300	
	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)	Iter.	CPU (s)
	Convergence to the dominant solvent							
MSDA	6	0.1563	5	0.2188	5	1.6875	5	3.6213
BM	21	0.5313	9	0.2969	9	2.4062	9	4.7031
	Convergence to the minimal solvent							
MSDA	6	0.1250	5	0.2500	5	2.1875	5	3.8719
BM	25	0.6094	30	0.9063	30	7.1563	30	16.2656
	Convergence to a solvent (not necessary the extreme)							
NME	9	1.2562	6	4.3125	6	59.14	5	167.6094

6. Concluding remarks

We have proposed an M-matrix structure-preserving doubling method for solving the quadratic matrix equation. The method retains the advantage of the structure-preserving doubling method in [10] that it is globally and quadratically convergent. Moveover, the generated sequence converges to the extreme solution of the equation monotonically. The proposed algorithm saves computational cost in that at each iteration, only an LU-factorization and several matrix multiplications are required, which is nearly 1/6 the cost of Newton's method. The preliminary numerical results show that the proposed method outperforms the well-know Newton's method with exact line searches and Bernoulli's method. They also show that the global convergence of the MSD method holds even if the condition $B^{-1}C \geq 0$ is not satisfied. At the moment, we are not aware if it is possible to prove it theoretically. We leave it as a possible topic for further study.

Acknowledgement. The authors would like to thank Prof. Hyun-Min Kim and Prof. Beatrice Meini for providing us their Matlab codes.

References

[1] E.K.W. Chu, H.Y. Fan and W.W. Lin, *Structure-preserving doubling algorithm for continuous-time algebraic Riccati equation*, Linear Algebra Appl., 396 (2005), pp. 55-88.

[2] J. Eisenfeld, *Operator equations and nonlinear eigenparameter problems*, J. Funct. Anal., 12 (1973), pp. 475-490.

[3] J.D. Gardiner, A.J. Laub, J.J. Amato and C.B. Moler, *Solution of the Sylvester matrix equation $AXB^T + CXD^T = E$*, ACM Trans. Math. Software., 18 (1992), pp. 223-231.

[4] J.D. Gardiner, A.J. Laub, J.J. Amato and C.B. Moler, *Algorithm 705: A FORTRAN-77 software package for solving the Sylvester matrix equation $AXB^T + CXD^T = E$*, ACM Trans. Math. Software., 18 (1992), pp. 232-238.

[5] G.H. Golub and C.F. Van Loan, *Matrix Computations*, (3rd Edition), Johns Hopkins University Press, Baltimore, MD, (1996).

[6] C.H. Guo, *Nonsymmetric algebraic Riccati equation and Wiener-Hopf fractorization for M-matrices*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 225-242.

[7] C.H. Guo, *Convergence rate of an iterative method for a nonlinear matrix equation*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 295-302.

[8] C.H. Guo, B. Iannazzo and B. Meini, *On the doubling algorithm for a (shifted) nonsymmetric algebraic Riccati equations*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 1083-1100.

[9] C.H. Guo and A. Laub, *On the iterative solution of a class of nonsymmetric algebraic Riccati equations*, SIAM J. Matrix Anal. Appl., 22 (2000), pp. 376-391.

[10] X.X. Guo, W.W. Lin and S.F. Xu, *A structure-preserving doubling algorithm for nonsymmetric algebraic Riccati equation*, Numer. Math. 103 (2006), pp. 393-412.

[11] N.J. Higham and H.M. Kim, *Solving a quadratic matrix equation by Newton's method with exact line search*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 303-316.

[12] N.J. Higham and H.M. Kim, *Numerical ananlysis of a quadratic matrix equation*, IMA. J. Numer. Anal., 20 (2000), pp. 499-519.

[13] W. Jones and W. Thron, *Continued Fractions: Analytic Theory and Applications*, Encyclopedia of Mathematics and Its Applications, Vol 2, Addison-Wesley, Reading, (1980).

[14] P. Lancaster, *Lambda-Matrices and Vibrating Systems*, (2nd Edition), Dover, Mineola, (2002).

[15] W.W. Lin and S.F. Xu, *Analysis of structure-preserving doubling algorithms for Riccati-type matrix equations*, SIAM J. Matrix Anal. Appl. (2005).

[16] J.E. McFarland, *An iterative solution to the quadratic equation in Bananch space*, Proc. Am. Math. Soc., 9(5) (1958), pp. 824-830.

[17] Y.F. Su, K.J. Zhang and M.L. Liu, *A sufficient condition on the existence of maximal and minimal solvents of the quadratic matrix equation*, J. Fudan Uni. (Nature Science), Vol. 43, No. (2004), pp. 303-306.

[18] F. Tisseur, *Backward error and condition of polynomial eigenvalue problems*, Linear Algebra Appl., 309 (2000), pp. 339-361.

[19] F. Tisseur and K. Meerbergen, *The quadratic eigenvalue problems*, SIAM Rev., 43, 2, (2001), pp. 235-286.

[20] S.G. Wang, M.X. Wu and Z.Z. Jia, *Inequalities of Matrix*, (2nd Edition), Science Press., (2006)

[21] R. Varga, *Matrix Iterative Analysis*, (2nd Edition), Springer-Verlag Berlin Heidelberg, (2000).