

## Research Article

# A Numerical Method for Computing the Principal Square Root of a Matrix

F. Soleymani,<sup>1</sup> S. Shateyi,<sup>2</sup> and F. Khaksar Haghani<sup>1</sup>

<sup>1</sup>Department of Mathematics, Shahrekord Branch, Islamic Azad University, Shahrekord, Iran

<sup>2</sup>Department of Mathematics and Applied Mathematics, School of Mathematical and Natural Sciences, University of Venda, Private Bag X5050, Thohoyandou 0950, South Africa

Correspondence should be addressed to S. Shateyi; stanford.shateyi@univen.ac.za

Received 1 June 2014; Revised 25 July 2014; Accepted 7 August 2014; Published 27 August 2014

Academic Editor: Vinay Kanwar

Copyright © 2014 F. Soleymani et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

It is shown how the mid-point iterative method with cubical rate of convergence can be applied for finding the principal matrix square root. Using an identity between matrix sign function and matrix square root, we construct a variant of mid-point method which is asymptotically stable in the neighborhood of the solution. Finally, application of the presented approach is illustrated in solving a matrix differential equation.

## 1. Introductory Notes

Let us consider a scalar function  $f$  and a matrix  $A \in \mathbb{C}^{n \times n}$  and specify  $f(A)$  to be a matrix function of the same dimensions as  $A$ . Higham in [1] indicates that if  $f$  is defined on the spectrum of  $A$ , then one has the following properties for the matrix function  $f(A)$ :

- (i)  $f(A)$  commutes with  $A$ ;
- (ii)  $f(A^T) = f(A)^T$ ;
- (iii)  $f(XAX^{-1}) = Xf(A)X^{-1}$ ;
- (iv) the eigenvalues of  $f(A)$  are  $f(\lambda_i)$ , where  $\lambda_i$  are the eigenvalues of  $A$ ;
- (v) if  $X$  commutes with  $A$ , then  $X$  commutes with  $f(A)$ .

In this paper, we are interested in numerical computation of matrix square root, which is one of the most fundamental matrix functions with potential applications. An example involving the computation of matrix square roots appears in a semidefinite programming algorithm [2]. A step length is calculated using a line search algorithm that involves computing the unique symmetric positive definite (SPD) square roots of a sequence of SPD matrices, where each matrix differs little from the previous one. That is, given a SPD

$A \in \mathbb{R}^{n \times n}$  and the SPD square root  $X$  of  $A$ , one wishes to find the SPD square root  $\bar{X}$  of  $\bar{A}$ , where  $\|\bar{A} - A\|$  is small. This has to be done repeatedly with  $A$  changing slightly each time.

It is known that [3]  $A \in \mathbb{C}^{n \times n}$  has a square root if and only if in the ascent sequence of integers  $d_1, d_2, \dots$ , defined by

$$d_i = \dim(\text{null}(A^i)) - \dim(\text{null}(A^{i-1})), \quad (1)$$

no two terms are the same odd integer.

Among the square roots of a matrix, the principal square root is distinguished by its usefulness in theory and applications. Let  $A \in \mathbb{C}^{n \times n}$  have no eigenvalues on  $\mathbb{R}^-$  (the closed negative real axis). There is a unique square root  $X$  of  $A$  that all of its eigenvalues lie in the open right half-plane, and it is a primary matrix function of  $A$  [4]. We refer to  $X$  as the principal square root of  $A$  and write  $X = A^{1/2}$ . Note that if  $A$  is real, then  $A^{1/2}$  is real.

Björck and Hammarling [5] developed a numerical method for the principal square root via Schur form. Among all the available numerical procedures, the Newton's matrix method and its variants have received many attractions in order to tackle this problem. To be more precise, applying the famous Newton's method to the matrix equation

$$F(X) \equiv X^2 - A = 0 \quad (2)$$

would yield the following (simplified) matrix iteration method [4]:

$$X_{k+1} = \frac{1}{2} [X_k + AX_k^{-1}], \quad (3)$$

for suitable initial values  $X_0$ , wherein the commutativity of matrices  $X_k$  and  $A$  has been considered to simplify the procedure as much as possible. Unfortunately, this iteration becomes unstable in most cases as pointed out by Higham [4].

To remedy this, Denman and Beavers (DB) in [6] proposed a quadratically convergent variant of Newton's matrix method (3) which is numerically stable in what follows:

$$\begin{aligned} Y_0 &= A, & Z_0 &= I, & k &= 0, 1, \dots, \\ Y_{k+1} &= \frac{1}{2} [Y_k + Z_k^{-1}], \\ Z_{k+1} &= \frac{1}{2} [Z_k + Y_k^{-1}]. \end{aligned} \quad (4)$$

This method generates the sequence  $\{Y_k\}$ , which converges to  $A^{1/2}$ . For further readings, one may refer to [7, 8].

In the next section, we first apply the following mid-point cubically convergent scheme of Frontini and Sormani (given for scalar nonlinear equations [9])

$$\begin{aligned} y_k &= x_k - 2^{-1} f'(x_k)^{-1} f(x_k), \\ x_{k+1} &= x_k - f'(y_k)^{-1} f(x_k), \end{aligned} \quad (5)$$

to derive a new iterative scheme for the numerical computation of  $A^{1/2}$ . In Section 2, we theoretically prove that the scheme is convergent under some proper conditions. A variant of the mid-point rule (5) is brought forward to provide a stable iteration for finding both the principal matrix square root and the principal inverse matrix square root. We note that our results in this work are some extensions to [10]. In order to derive a higher order new version of mid-point method for computing square root, we use some of the methods in the literature and obtain a new scheme for  $A^{1/2}$  as well. Section 3 is devoted to the application of the discussed matrix methods in solving some examples. Finally, Section 4 draws a conclusion of this paper.

## 2. Extension of the Mid-Point Method

Multipoint nonlinear solvers such as (5) belong to the class of the most efficient methods for nonlinear scalar equations. Recent interests in the research and development of this type of methods have arisen from their capability to overcome theoretical limits of one-point methods concerning the convergence order and computational efficiency.

Let us apply (5) for solving the nonlinear matrix equation (2). We attain the following scheme:

$$X_{k+1} = [3AX_k + X_k^3] [A + 3X_k^2]^{-1}, \quad (6)$$

considering the fact that the initial approximation is of the following forms:

$$X_0 = \alpha I, \quad (7)$$

or

$$X_0 = \beta A, \quad (8)$$

wherein  $\alpha$  and  $\beta$  are suitable positive parameters to ensure convergence. The derivation of (6) may not be clear yet. Hence, we give a Mathematica code written for obtaining it as fast as possible in a symbolic language as follows:

```
ClearAll["Global`*"]
f[X_] := X^2 - A;
Y = X - f[X]/(2f'[X]) // FullSimplify;
FullSimplify[X - f[X]/f'[Y]]
```

**Lemma 1.** Let  $A = [a_{ij}]_{n \times n}$  be a nonsingular complex matrix with no eigenvalues on  $\mathbb{R}^-$ . If  $AX_0 = X_0A$  is valid, then for the sequence of  $\{X_k\}_{k=0}^{\infty}$  of (6), one has that

$$AX_k = X_kA \quad (9)$$

holds, for all  $k = 1, 2, \dots$

*Proof.* The proof is by induction. Let  $X_0$  be one of the initial values (7) or (8); then it is obvious that  $AX_0 = X_0A$ . For the inductive hypothesis, we take  $k > 0$  and  $AX_k = X_kA$  and show that

$$\begin{aligned} AX_{k+1} &= A [3AX_k + X_k^3] [A + 3X_k^2]^{-1} \\ &= [3A^2X_k + AX_k^3] [A + 3X_k^2]^{-1} \\ &= [3(X_kA)A + X_k^2(X_kA)] [A + 3X_k^2]^{-1} \\ &= [3X_kA + X_k^3] [A^{-1}A + 3A^{-1}X_k^2]^{-1} \\ &= [3X_kA + X_k^3] [AA^{-1} + 3X_k^2A^{-1}]^{-1} \\ &= X_{k+1}A. \end{aligned} \quad (10)$$

Note that, from  $AX_k = X_kA$ , we obtain that  $A^{-1}X_k = X_kA^{-1}$ . This completes the proof.  $\square$

Let  $(M, d)$  be a metric space. We say that a sequence  $\{X_k\}$  of points of  $M$  converges to a point  $X^*$  of  $M$  if

$$\lim_{k \rightarrow \infty} d(X_k, X^*) = 0. \quad (11)$$

The convergence properties will depend on the choice of a distance in  $M$ , but for a given distance the speed of convergence of the sequence  $\{X_k\}$  is characterized by the speed of convergence of the sequence of nonnegative numbers  $d(X_k, X^*)$  [11]. In what follows, we investigate the rate of convergence for (6).

**Theorem 2.** Let  $A = [a_{i,j}]_{n \times n}$  have no eigenvalues on  $\mathbb{R}^-$ . If the initial approximation  $X_0$  is according to the rules (7) or (8),

then the iterative method (6) converges to the principal  $A^{1/2}$  with local third order of convergence (without considering the round-off error).

*Proof.* Using the methodology of [10], we now prove the convergence order of (6). If  $A$  is considered to be diagonalizable and nonsingular, then

$$V^{-1}AV = D = \text{diag} \{ \lambda_1, \dots, \lambda_n \}, \quad (12)$$

where  $V$  is a nonsingular matrix. In addition assume that  $M_k = V^{-1}X_kV$ . From the iterative method (6), we attain

$$M_{k+1} = [3DM_k + M_k^3] [D + 3M_k^2]^{-1}. \quad (13)$$

$\{M_k\}$  is the sequence of real block diagonal matrices  $M_k = \text{diag}\{m_1^{(k)}, \dots, m_n^{(k)}\}$ . Note that if  $M_0$  is a diagonal matrix, then all successive  $M_k$  are diagonal too. The recursive relation (13) is equivalent to  $n$  scalar equations as follows:

$$m_i^{(k+1)} = [3\lambda_i m_i^{(k)} + (m_i^{(k)})^3] [\lambda_i + 3(m_i^{(k)})^2]^{-1}, \quad 1 \leq i \leq n. \quad (14)$$

The sequence  $\{m_i^{(k)}\}$  is real and it is obvious that

$$\lim_{k \rightarrow \infty} M_k = D^{1/2} = \text{diag} \{ \sqrt{\lambda_1}, \dots, \sqrt{\lambda_n} \}. \quad (15)$$

Then the fact that  $M_k = V^{-1}X_kV$  yields  $\lim_{k \rightarrow \infty} X_k = VD^{1/2}V^{-1} = A^{1/2}$ . Therefore,

$$M_{k+1} - D^{1/2} = \frac{1}{8} (3M_k + D^{1/2}) M_k^{-3} (M_k - D^{1/2})^3. \quad (16)$$

Using a transformation by  $V$ , we have

$$X_{k+1} - A^{1/2} = \frac{1}{8} (3X_k + A^{1/2}) X_k^{-3} (X_k - A^{1/2})^3. \quad (17)$$

Taking norm of (17), we obtain that

$$\|X_{k+1} - A^{1/2}\| \leq \frac{1}{8} \|3X_k + A^{1/2}\| \|X_k^{-1}\|^3 \|X_k - A^{1/2}\|^3. \quad (18)$$

Therefore, the method (6) possesses the local third order of convergence. Equation (11) reveals, in a metric space, how convergence could be understood from (17). Furthermore since  $A$  has no nonpositive real eigenvalues, so the mid-point approach computes the principal matrix square root.

We remark that all iterates  $\{X_k\}$  generated by (6) are well defined when the function  $f(x) = a + 3x^2$  is not zero for all nonnegative numbers  $x$ . In fact, the initial values actually determine a unique real number  $x_k$  for each  $k \geq 1$ . In general, well defined means that some object was given a description up to some arbitrary choices but that the arbitrary choices have no material effect. Accordingly, using the initial choices  $A$  and  $I$ , the sequence is well defined.  $\square$

Under the assumptions of Theorem 2. (without considering the round-off error), and when  $A$  is nonsingular,  $X_k$

tends to  $A^{1/2}$ . This means that  $X_k^{-1}$  tends to  $A^{-1/2}$ , that is, the principal inverse matrix square root.

Unfortunately, scheme (6) has turned out to be unstable when the size of the input matrix  $A$  is high or is ill-conditioned. A similar reasoning as in [4] could reveal this point using  $X_0 = A$ . Therefore, (6) can only be employed for symmetric positive definite matrices with extremely well-conditioned structure.

Iannazzo in [12] showed that in the case that  $X_0$  is the identity matrix, the Newton's method converges for any matrix  $A$  having eigenvalues with modulus less than 1 and with positive real parts. Similar results can be deduced for (6).

In order to reach third order of convergence by the mid-point approach in solving (2) as well as to obtain a stable sequence, we use a fact (identity) in what follows:

$$\text{sign} \left( \begin{bmatrix} 0 & A \\ I & 0 \end{bmatrix} \right) = \begin{bmatrix} 0 & A^{1/2} \\ A^{-1/2} & 0 \end{bmatrix}, \quad (19)$$

which indicates an important relationship between matrix square root  $A^{1/2}$  and the matrix sign function  $\text{sign}(A)$ .

We remark that (19) has an extra important advantage which is the computation of the inverse matrix square root along with the matrix square root at the same time.

Applying (5) to solve the matrix equation  $G(X) \equiv X^2 - I = 0$  yields in its reciprocal form

$$X_{k+1} = [I + 3X_k^2] [X_k (3I + X_k^2)]^{-1}. \quad (20)$$

This method which is similar to the application of Halley's method for solving  $X^2 = I$  (see, e.g., [13, 14]) computes the matrix sign of the nonsingular matrix

$$\begin{bmatrix} 0 & A \\ I & 0 \end{bmatrix} \quad (21)$$

and in the convergence phase gives

$$\begin{bmatrix} 0 & A^{1/2} \\ A^{-1/2} & 0 \end{bmatrix}. \quad (22)$$

It is easy to deduce the following convergence theorem for (20).

**Theorem 3.** *Let  $A = [a_{i,j}]_{n \times n}$  have no eigenvalues on  $\mathbb{R}^-$ . If  $X_0$  is (21), then the iterative method (20) converges to (22).*

*Proof.* The proof of this theorem is based on the diagonalization property of the sign matrix  $S$ . It is hence omitted.  $\square$

**Lemma 4.** *The sequence  $\{X_k\}_{k=0}^{k=\infty}$  generated by (20) using  $X_0 = (21)$  is asymptotically stable.*

*Proof.* Let  $\Delta_k$  be the numerical perturbation introduced at the  $k$ th iterate of (20). Next, one has

$$\widetilde{X}_k = X_k + \Delta_k. \quad (23)$$

Here, we perform a first-order error analysis; that is, we formally neglect quadratic terms such as  $(\Delta_k)^2$ . We have

$$\begin{aligned}\widetilde{X}_{k+1} &= [I + 3\widetilde{X}_k^2] [\widetilde{X}_k (3I + \widetilde{X}_k^2)]^{-1} \\ &= [I + 3(X_k + \Delta_k)^2] [(X_k + \Delta_k) (3I + (X_k + \Delta_k)^2)]^{-1},\end{aligned}\quad (24)$$

where the following identity will be used (for any nonsingular matrix  $B$  and the matrix  $C$ ):

$$(B + C)^{-1} \approx B^{-1} - B^{-1}CB^{-1}. \quad (25)$$

Further simplifying yields to

$$\begin{aligned}\widetilde{X}_{k+1} &\approx (I + 3(S + \Delta_k)^2) ((S + \Delta_k) (3I + (S + \Delta_k)^2))^{-1} \\ &\approx (4I + 3S\Delta_k + 3\Delta_k S) ((S + \Delta_k) (4I + S\Delta_k + \Delta_k S))^{-1} \\ &\approx (4I + 3S\Delta_k + 3\Delta_k S) \left( \frac{1}{4}S - \frac{5}{16}S\Delta_k S - \frac{1}{16}\Delta_k \right),\end{aligned}\quad (26)$$

where  $S^2 = I$ ,  $S^{-1} = S$ , and, for enough large  $k$ , we have considered  $X_k \approx S$ . After some algebraic manipulations and using  $\Delta_{k+1} = \widetilde{X}_{k+1} - X_{k+1} = \widetilde{X}_{k+1} - S$ , we conclude that

$$\Delta_{k+1} \approx \frac{1}{2}\Delta_k - \frac{1}{2}S\Delta_k S. \quad (27)$$

Applying (27) recursively, and after some algebraic manipulations, we have

$$\|\Delta_{k+1}\| \leq \frac{1}{2^{k+1}} \|\Delta_0 - S\Delta_0 S\|. \quad (28)$$

From (28), we can conclude that the perturbation at the iterate  $k + 1$  is bounded. Therefore, the sequence  $\{X_k\}$  generated by (20) is asymptotically stable.  $\square$

The iteration (20) requires one matrix inversion per computing step and obtains both  $A^{1/2}$  and  $A^{-1/2}$  which are of interest in most practical problems. The implementation of (20) for computing principal square roots requires a sharp attention so as to save much effort. Since the intermediate matrices are all sparse (at least half of the entries are zero), then one could simply use sparse approximation techniques to save up the memory and time.

**2.1. A Novel Variant of Mid-Point Scheme.** The analysis of the method from applying mid-point method to  $X^2 = A$  was of limited interest, because it yields an iteration method that is not in general stable. The method with cubic convergence derived via the matrix sign function (20) is more useful. However, this method is somewhat not challenging, since it could be a special case of a family of iteration methods due to Kenney and Laub [15] and derived from the (explicitly known) rational Padé approximations for  $h(\zeta) = (1 - \zeta)^{-1/2}$ . These methods are surveyed in Higham [1, Section 5.4] and the application to the matrix square root is discussed

in Section 6.7. General results on the stability and rate of convergence for this family of methods are given in Theorems 6.11 and 6.12 of [1].

To move forward, we recall that recently authors in [16] proposed an efficient and interesting variant of mid-point method for nonlinear equations and extended it for polar decomposition, while authors of [17] developed that variant for matrix sign function as follows:

$$X_{k+1} = (I + 18Y_k + 13Z_k) [X_k (7I + Y_k) (I + 3Y_k)]^{-1}, \quad (29)$$

where  $Y_k = X_k X_k$ ,  $Z_k = Y_k Y_k$ .

**Theorem 5.** Let  $A = [a_{i,j}]_{n \times n}$  have no eigenvalues on  $\mathbb{R}^-$ . If one uses  $X_0 = (21)$ , then the iterative method (29) converges to (22).

*Proof.* The proof of this theorem is similar to the proof of Theorem 3.  $\square$

**Lemma 6.** The sequence  $\{X_k\}_{k=0}^{k=\infty}$  generated by (29) using  $X_0 = (21)$  is asymptotically stable.

*Proof.* The proof of this lemma is similar to the proof of Lemma 4.  $\square$

Note that in general the inverse of the nonsingular matrix

$$\begin{bmatrix} 0 & K \\ H & 0 \end{bmatrix} \quad (30)$$

is of the form

$$\begin{bmatrix} 0 & H^{-1} \\ K^{-1} & 0 \end{bmatrix}. \quad (31)$$

In fact, all matrices  $X_k(7I + Y_k)(I + 3Y_k)$  in (29) would be of the form

$$\begin{bmatrix} 0 & \dagger \\ \ddagger & 0 \end{bmatrix}. \quad (32)$$

This itself might be helpful to reduce the computational time of matrix inversion problem.

An implementation of (29) to compute  $A^{1/2}$  and  $A^{-1/2}$  for a square matrix  $A$  having no eigenvalues on  $\mathbb{R}^-$ , in the programming package Mathematica, is brought forward as follows:

```
sqrtMatrix2[A_, maxIterations_, tolerance_]
:= Module[{k = 0}, {n, n} = Dimensions[A];
  Id = SparseArray[{{i_, i_} -> 1.}, {n, n}];
  A1 = SparseArray@ArrayFlatten[{{0, A},
    {Id, 0}}];
  Y[0] = A1; R[0] = 1;
  Id2 = SparseArray[{{i_, i_} -> 1.},
    {2 n, 2 n}]; While[k < maxIterations && R[k]
    >= tolerance,
  Y2 = SparseArray[Y[k].Y[k]];
  l1 = SparseArray[Y[k].(7 Id2 + Y2).(Id2 + 3 Y2)];
  l2 = SparseArray@ArrayFlatten[{{0,
    Inverse@l1[{n + 1; 2 n, 1; n}]}],
```

```
{Inverse@11[[1;;n,n+1;2 n],0]};
Y[k+1]=SparseArray[(Id2+18 Y2
+13 Y2.Y2).12];R[k+1]=Norm[Y[k+1]-Y[k],
Infinity]/Norm[Y[k+1],
Infinity];k++;Y[k];
```

The above three-argument function `sqrtMatrix2` computes the principal matrix square root and its inverse at the same time by entering the three arguments “matrix  $A$  with no eigenvalue in  $\mathbb{R}^-$ ,” “the maximum number of iterations that (29) could take,” and the tolerance of the stopping termination in the infinity norm  $\|X_{k+1} - X_k\|_\infty / \|X_{k+1}\|_\infty \leq \textit{tolerance}$ . Clearly `sqrtMatrix2` [ $A, \textit{maxIterations}, \textit{tolerance}$ ] [ $[1;;n,n+1;2n]$ ] and `sqrtMatrix2`[ $A, \textit{maxIterations}, \textit{tolerance}$ ][ $[n+1;2n, 1;;n]$ ] give  $A^{1/2}$  and  $A^{-1/2}$ , respectively, wherein  $n$  is the size of the input matrix  $A$ .

**2.2. Extension to Matrix  $p$ th Root.** Computing the  $p$ th roots of matrices can arise in certain computations. The unique matrix  $X$  such that  $X^p = A$ , ( $p \in \mathbb{Z}^+$ ) and whose eigenvalues are in the segment

$$S = \left\{ z \in \mathbb{C} \setminus \{0\} : \left| \arg(z) \right| < \frac{\pi}{p} \right\} \tag{33}$$

is called the principal  $p$ th root and is denoted by  $A^{1/p}$ .

Many authors have investigated methods for computing  $p$ th root of matrices. The methods are normally based on iteration or Schur normal form; see, for example, [18] and the references therein. In general, number of  $p$ th roots for a given matrix may be zero, finite, or infinite.

In this subsection, we provide a combination of Algorithm 7.9 in [1] and our method (29) to compute principal matrix  $p$ th root in what follows.

Given  $A \in \mathbb{C}^{n \times n}$  having no eigenvalues on  $\mathbb{R}^-$ ,  $p > 2$ , the following lines compute  $X = A^{1/p}$  using (29):

- (1) compute  $A^{1/2}$  by (29) and set  $B = A^{1/2}$ ;
- (2) set  $C = B/\theta$ , where  $\theta$  is any upper bound for  $\rho(B)$  (e.g.,  $\theta = \|B\|$ );
- (3) use the coupled Newton iteration (35) with  $X_0 = I$  to compute

$$X = \begin{cases} C^{2/p}, & p \text{ even,} \\ (C^{1/p})^2, & p \text{ odd;} \end{cases} \tag{34}$$

- (4)  $X \leftarrow \|B\|^{2/p} X$ ,

where

$$\begin{aligned} X_{k+1} &= X_k \left( \frac{(p-1)I + M_k}{p} \right), & X_0 &= I, \\ M_{k+1} &= \left( \frac{(p-1)I + M_k}{p} \right)^{-p} M_k, & M_0 &= C, \end{aligned} \tag{35}$$

when  $X_k \rightarrow A^{1/p}$ ,  $M_k \rightarrow I$ .

### 3. Applications

We test the contributed method (29) denoted by PM1 using Mathematica 8 in machine precision [19]. Apart from these schemes, several iterative methods (3), (4), (20) denoted by PM and the method of Meini originally developed in [20] based on the cyclic reduction algorithm (CR) have been considered as follows:

$$\begin{aligned} Z_0 &= 2(I + A), & Y_0 &= I - A, \\ Y_{k+1} &= -Y_k Z_k^{-1} Y_k, \\ Z_{k+1} &= Z_k - 2Y_k Z_k^{-1} Y_k, \\ k &= 0, 1, \dots \end{aligned} \tag{36}$$

This method generates the sequence  $\{Z_k\}$ , which converges to  $4A^{1/2}$ . Note that (36) only computes  $A^{1/2}$  while (4), (20), and (29) produce  $A^{1/2}$  and  $A^{-1/2}$ .

Note that PM is equal to Halley’s third order method [14].

The computational order of convergence for computing the principal matrix square root by matrix iterations can be estimated by

$$\rho = \frac{\ln \left( \|X_k^2 - A\|^{-1} \|X_{k+1}^2 - A\| \right)}{\ln \left( \|X_{k-1}^2 - A\|^{-1} \|X_k^2 - A\| \right)}, \tag{37}$$

wherein  $X_{k-1}, X_k, X_{k+1}$  are the last three approximations for finding  $A^{1/2}$  in the convergence phase.

*Example 7.* As the first example, we consider a matrix

$$A = \begin{pmatrix} \frac{1}{4} + \frac{\pi^2}{81} & \frac{\pi}{9} \\ \frac{\pi}{9} & 1 \end{pmatrix}, \tag{38}$$

whereas its eigenvalues are  $\{1.15549, 0.216359\}$ . The principal matrix square root of  $A$  is

$$X = A^{1/2} \approx \begin{pmatrix} 0.566105 & 0.226654 \\ 0.226654 & 0.973975 \end{pmatrix}. \tag{39}$$

We compare the behavior of different methods and report the numerical results using  $l_\infty$  for all norms involved with the stopping criterion

$$E_{k+1} = \frac{\|X_{k+1} - X_k\|_\infty}{\|X_{k+1}\|_\infty} \leq \epsilon = 10^{-8}, \tag{40}$$

in Table 1. The numerical results are in harmony with the theoretical aspects of Section 2.

A differential equation is a mathematical equation for an unknown function of one or several variables that relate the values of the function itself and its derivatives of various orders. A matrix differential equation contains more than one function stacked into vector form with a matrix relating the functions to their derivatives. We now illustrate the application of the new solver for solving matrix differential equations.

TABLE 1: Results of comparisons for Example 7.

	Newton	DB	CR	PM	PM1
IT	6	6	5	4	3
$E_{k+1}$	$9.33948 \times 10^{-15}$	$9.31636 \times 10^{-15}$	$9.22389 \times 10^{-15}$	$3.29612 \times 10^{-12}$	$7.94163 \times 10^{-11}$
$\rho$	1.99877	1.99908	2.00031	2.97414	4.02282

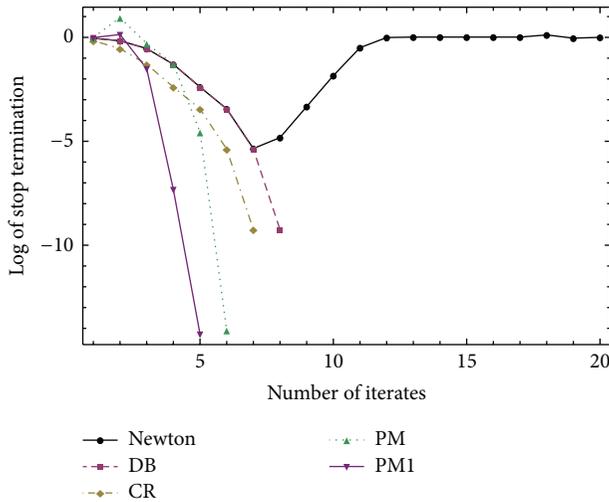


FIGURE 1: The comparison of different methods for finding  $A^{1/2}$  in Example 8.

*Example 8.* In this test, we compare the behavior of different methods for solving the following matrix differential equation:

$$\frac{\partial^2 y(t)}{\partial t^2} + Ay(t) = 0, \tag{41}$$

$$y(0) = y_0 \quad y'(0) = y'_0(0),$$

where its exact solution can be expressed as

$$y(t) = \cos(A^{1/2}t) y_0(0) + A^{-1/2} \sin(A^{1/2}t) y'_0(0), \tag{42}$$

and the sparse SPD  $100 \times 100$  matrix  $A$  in Mathematica is defined as follows:

```
A = SparseArray[{{i_., i_} -> 10., {i_., j_} /;
Abs[i - j] == 1 -> -5.}, {n, n}];
```

The interesting point in this example is that we need both  $A^{1/2}$  and  $A^{-1/2}$  at the same time and this paves up the application of (29). We compare the behavior of different methods in Figure 1 for finding the principal matrix square root of  $A$ . The results are promising for the proposed method in terms of the number of iterations. Figure 1 reveals that the fastest method is (29), while the Newton’s matrix method is numerically unstable and this makes it diverge after the seventh iterate.

Note again that among the compared methods only the method of DB and our proposed method PM1 are able to find  $A^{-1/2}$  at the same time which is fruitful in solving

the matrix differential equation (41). Due to page limitation, we could not present the final solution of (41) explicitly here. Clearly, using the obtained principal matrix square root and the principal inverse matrix square root produces the final solution.

### 4. Conclusion

In this paper, we have developed and discussed the mid-point iterative root-finding method for solving some special and important matrix problems. We discussed under what conditions the obtained method is convergent to the principal matrix square root, that is, when  $A$  has no nonpositive real eigenvalues.

Furthermore, an asymptotically stable variant of the mid-point method using an identity between the matrix square root and the matrix sign function has been derived. The numerical experiments which we have performed, reported at the end of the paper, show that the presented procedure is a reliable tool for computing the principal matrix square root and its inverse.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgment

The authors would like to record their sincerest thanks to the referees which their comments have helped a lot in enhancing the readability of this paper.

### References

- [1] N. J. Higham, *Functions of Matrices: Theory and Computation*, Society for Industrial and Applied Mathematics, Philadelphia, Pa, USA, 2008.
- [2] C. Kanzow and C. Nagel, “Semidefinite programs: new search directions, smoothing-type methods, and numerical results,” *SIAM Journal on Optimization*, vol. 13, no. 1, pp. 1–23, 2002.
- [3] G. W. Cross and P. Lancaster, “Square roots of complex matrices,” *Linear and Multilinear Algebra*, vol. 1, pp. 289–293, 1977.
- [4] N. J. Higham, “Newton’s method for the matrix square root,” *Mathematics of Computation*, vol. 46, no. 174, pp. 537–549, 1986.
- [5] Å. Björck and S. Hammarling, “A Schur method for the square root of a matrix,” *Linear Algebra and its Applications*, vol. 52/53, pp. 127–140, 1983.
- [6] E. D. Denman and J. Beavers, “The matrix sign function and computations in systems,” *Applied Mathematics and Computation*, vol. 2, no. 1, pp. 63–94, 1976.

- [7] J. M. Gutierrez, M. A. Hernandez, and M. A. Salanova, "Calculus of  $n$ th roots and third order iterative methods," *Nonlinear Analysis: Theory, Methods & Applications*, vol. 47, no. 4, pp. 2875–2880, 2001.
- [8] B. Iannazzo, "A note on computing the matrix square root," *Calcolo*, vol. 40, no. 4, pp. 273–283, 2003.
- [9] M. Frontini and E. Sormani, "Some variant of Newton's method with third-order convergence," *Applied Mathematics and Computation*, vol. 140, no. 2-3, pp. 419–426, 2003.
- [10] S. Lakić and M. S. Petković, "On the matrix square root," *Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 78, no. 3, pp. 173–182, 1998.
- [11] F. A. Potra, "On Q-order and R-order of convergence," *Journal of Optimization Theory and Applications*, vol. 63, no. 3, pp. 415–431, 1989.
- [12] B. Iannazzo, "On the Newton method for the matrix  $p$ th root," *SIAM Journal on Matrix Analysis and Applications*, vol. 28, no. 2, pp. 503–523, 2006.
- [13] W. Gander, "On Halley's iteration method," *The American Mathematical Monthly*, vol. 92, no. 2, pp. 131–134, 1985.
- [14] C. H. Guo, "On Newton's method and Halley's method for the principal  $p$ th root of a matrix," *Linear Algebra and its Applications*, vol. 432, no. 8, pp. 1905–1922, 2010.
- [15] C. S. Kenney and A. J. Laub, "Rational iterative methods for the matrix sign function," *SIAM Journal on Matrix Analysis and Applications*, vol. 12, no. 2, pp. 273–291, 1991.
- [16] F. K. Haghani and F. Soleymani, "On a fourth-order matrix method for computing polar decomposition," *Computational and Applied Mathematics*, 2014.
- [17] F. Soleymani, E. Tohidi, S. Shateyi, and F. K. Haghani, "Some matrix iterations for computing matrix sign function," *Journal of Applied Mathematics*, vol. 2014, Article ID 425654, 9 pages, 2014.
- [18] C. H. Guo and N. J. Higham, "A Schur-Newton method for the matrix  $p$ th root and its inverse," *SIAM Journal on Matrix Analysis and Applications*, vol. 28, no. 3, pp. 788–804, 2006.
- [19] J. Hoste, *Mathematica Demystified*, McGraw-Hill, New York, NY, USA, 2009.
- [20] B. Meini, "The matrix square root from a new functional perspective: theoretical results and computational issues," Tech. Rep. 1455, Dipartimento di Matematica, Universit di Pisa, Pisa, Italy, 2003.