Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2014, Article ID 425654, 9 pages
http://dx.doi.org/10.1155/2014/425654

Research Article

Some Matrix Iterations for Computing Matrix Sign Function

F. Soleymani,[1] E. Tohidi,[1] S. Shateyi,[2] and F. Khaksar Haghani[1]

[1] *Department of Mathematics, Islamic Azad University, Shahrekord Branch, Shahrekord, Iran*
[2] *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 19 December 2013; Revised 29 May 2014; Accepted 2 June 2014; Published 9 July 2014

Academic Editor: Changbum Chun

Some iterative methods are introduced and demonstrated for finding the matrix sign function. It is analytically shown that the new schemes are asymptotically stable. Convergence analysis along with the error bounds of the main proposed method is established. Different numerical experiments are employed to compare the behavior of the new schemes with the existing matrix iterations of the same type.

1. Introduction

Recently, the theory of matrix functions becomes an active topic of research in the field of advanced numerical linear algebra (see, e.g., [1–4]). In fact, the most common matrix function is the matrix inverse or the Moore-Penrose generalized inverse, routinely used in the scientific problems [5]. General matrix functions as well as the specific cases have been extensively discussed and developed in [6].

This paper is concerned with a special case known as matrix sign function, which is of clear importance in the theory of matrix functions [7]. Let us, as Higham considered in the fifth Chapter of [6], assume throughout this paper that the matrix $A \in \mathbb{C}^{n\times n}$ has no eigenvalues on the imaginary axis. This assumption implies that the matrix sign function,

$$S = \operatorname{sign}(A), \tag{1}$$

can be uniquely defined, whereas A is a nonsingular square matrix. In order to define S, we remember the matrix sector function, for any positive integer p, can be defined by

$$\operatorname{sect}_p(A) = A\left(A^p\right)^{-1/p}. \tag{2}$$

Choosing $p = 2$ in the matrix sector function will yield in the matrix sign function as $S = A(A^2)^{-1/2}$. This also clearly puts on show the importance and the relevance of this matrix function to the other important matrix functions such as matrix square root.

Bini et al. in [8] proved that the principal pth root of the matrix A is a multiple of the (2,1)-block of the matrix sign function, sign(C), for the following block companion matrix

$$C = \begin{pmatrix} 0 & I & & & \\ & 0 & I & & \\ & & \ddots & \ddots & \\ & & & \ddots & I \\ A & & & & 0 \end{pmatrix} \in \mathbb{C}^{pn\times pn}. \tag{3}$$

The matrix S has the following properties.

(1) $S^2 = I$ (S is involutory).

(2) S is diagonalizable with eigenvalues ± 1.

(3) $SA = AS$.

(4) If A is real, then S is real.

(5) $(I+S)/2$ and $(I-S)/2$ are projectors onto the invariant subspaces associated with the eigenvalues in the right half-plane and left half-plane, respectively.

Although S has eigenvalues of ± 1, its norm can be arbitrarily large. Note that, for diagonalizable A, eigenvectors of A are eigenvectors of S, with eigenvalues of -1 and 1, respectively. For more, see [9].

There are some other definitions for S in the literature based on the Jordan canonical form and the integral representation. As indicated in [6], one of the most useful and widely applicable methods for computing S is the matrix iteration of Newton given by

$$X_{k+1} = \frac{1}{2}\left(X_k + X_k^{-1}\right). \tag{4}$$

In 1991, a fundamental family of matrix iterations for finding the matrix sign function S was introduced in [10] using Padé approximants to $f(\xi) = (1-\xi)^{-1/2}$ and the following characterization:

$$\operatorname{sign}(z) = s = \frac{z}{\left(z^2\right)^{1/2}} = \frac{z}{(1-\xi)^{1/2}}, \tag{5}$$

where $\xi = 1 - z^2$. Let the (m,n)-Padé approximant to $f(\xi)$ be $P_{m,n}(\xi)/Q_{m,n}(\xi)$, and $m+n \geq 1$. The iteration

$$z_{k+1} = \frac{z_k P_{m,n}\left(1-z_k^2\right)}{Q_{m,n}\left(1-z_k^2\right)} := \varphi_{2m+1,2n} \tag{6}$$

has been proved to be convergent to 1 and –1 with order of convergence $m+n+1$ form $m \geq n-1$. We remark that iterative methods of the type $z_k P_{m,n}(1-z_k^2)/Q_{m,n}(1-z_k^2)$ are fixed-point type iterations, and if $z_k P_{m,n}(1-z_k^2)/Q_{m,n}(1-z_k^2)$ does not converge then its reciprocal; that is, $Q_{m,n}(1-z_k^2)/z_k P_{m,n}(1-z_k^2)$ converges to the sign matrix (e.g., forthcoming iterations (7) and (8)). Generally speaking, the iterations of Kenney and Laub (6), generated by the $[h/h]$ and $[(h-1)/h]$ Padé approximants, are globally convergent and their orders depend on h. A discussion about such iterations was given in [11–13].

A lot of known methods could be extracted from the Padé family (6). For example, the well-known Halley's matrix iteration of order three can be deduced as follows:

$$X_{k+1} = \left[I + 3X_k^2\right]\left[X_k\left(3I + X_k^2\right)\right]^{-1}. \tag{7}$$

Another fourth-order method could be attained as follows:

$$X_{k+1} = \left[I + 6X_k^2 + X_k^4\right]\left[4X_k\left(I + X_k^2\right)\right]^{-1}. \tag{8}$$

Note that, for lower order methods such as (4), the convergence is slow; that is, initially convergence can be slow if $|x_k| \gg 1$. Hence, a scaling approach (a.k.a. norm scaling) to accelerate the beginning of this phase is necessary and can be done in what follows [6]:

$$\begin{aligned} X_0 &= A, \\ \mu_k &= \sqrt{\frac{\left\|X_k^{-1}\right\|}{\|X_k\|}}, \\ X_{k+1} &= \frac{1}{2}\left(\mu_k X_k + \mu_k^{-1} X_k^{-1}\right). \end{aligned} \tag{9}$$

Such an approach could be done to refine the initial matrix and to provide a much more robust initial matrix to arrive at the convergence phase rapidly.

The rest of this paper has been organized as follows. Section 2 gives the basic idea of obtaining other higher order solvers for computing S, while a fourth-order family of methods has been introduced. Section 3 is devoted to find the best method of this family in terms of the lowest computational cost. An analysis will be given to show that the new matrix iteration is asymptotically stable with local quartic convergence. To find a method with fourth order and global convergence, we also give another matrix iteration therein. Convergence analysis along with the error bounds of the proposed method is established. Numerical studies will be included in Section 4 to compare the efficiency and the stability of the schemes for finding S using different tests. Finally, a short conclusion of the study will be drawn in Section 5.

2. Basic Idea

The connection between the matrix iteration of Newton and Newton's root-finding method may not be clear at the first sight. Generally speaking, in the theory of matrix functions, many of the matrix functions could effectively be calculated by the existing iterative methods for finding the solution of nonlinear equations [14].

To illustrate further, apply Newton's method on the following matrix equation:

$$X^2 = I, \tag{10}$$

in which I is the identity matrix of the appropriate size; it would yield in the matrix Newton's iteration (4). Note that S is one solution of (10). Note that, in the last decade, many efficient higher-order iterative methods have been developed for solving nonlinear equations [15], and some of them have been extended to solve nonlinear matrix equation; see, for example, [16, 17]. But our work is the first to discuss the application of high order root solvers for matrix sign function.

Let us consider the following fourth-order family of iterative methods [18]:

$$\begin{aligned} y_k &= x_k - f'(x_k)^{-1} f(x_k), \\ x_{k+1} &= y_k - \left[f'(x_k)^{-1} f(y_k)\right] \\ &\quad \times \left[1 + 2f(x_k)^{-1} f(y_k) + \beta\left(f(x_k)^{-1} f(y_k)\right)^2\right], \end{aligned} \tag{11}$$

wherein β is a free parameter in $\mathbb{R}$. Applying the uniparametric family (11) on matrix equation (10) results in the following novel family of matrix iterations:

$$\begin{aligned} X_{k+1} &= \left[128X_k^7\right]^{-1} \\ &\quad \times \left[-\beta\left(-I + X_k^2\right)^4 + 8\left(X_k^2 - 5X_k^4 + 15X_k^6 + 5X_k^8\right)\right]. \end{aligned} \tag{12}$$

The free parameter β plays an important role in the next section to derive the best possible matrix iteration out of (12).

3. Main Results

In order to reduce the computational complexity of (12), the parameter β must be chosen as if the number of matrix-matrix products gets down along the number of matrix inversions.

Choosing $\beta = 0$ will simplify the whole family into the following method with reasonable computational cost in contrast to its convergence order:

$$X_{k+1} = \left[16X_k^5\right]^{-1}\left[I - 5X_k^2 + 15X_k^4 + 5X_k^6\right]. \quad (13)$$

We now rewrite obtained iteration (13) as efficiently as possible to reduce the number of matrix-matrix multiplications in what follows:

$$X_{k+1} = \frac{1}{16}\left[Y_k^5 - 5Y_k^3 + 15Y_k + 5X_k\right], \quad (14)$$

where $Y_k = X_k^{-1}$ and $X_0 = A$.

Definition 1 (stability [6]). Consider an iteration $X_{k+1} = g(X_k)$ with a fixed point X. Assume that g is Fréchet differentiable at X. The iteration is stable in a neighborhood of X if the Fréchet derivative $L_g(X)$ has bounded powers; that is, there exists a constant c such that $\|L_g^i(X)\| \le c$ for all $i > 0$.

Now, we first investigate the stability of (14) for the matrix sign function in a neighborhood of the solution of (10). In fact, we analyze how a small perturbation at the kth iterate is amplified or damped along the iterates. Note that a general way for assessing the stability of some matrix iterations has been studied by Iannazzo in [19]. The forthcoming approach follows these results of [19].

Lemma 2. *The sequence* $\{X_k\}_{k=0}^{k=\infty}$ *generated by* (14) *is asymptotically stable.*

Proof. Stability concerns behavior close to convergence and so is an asymptotic property. Let ΔX_k be the numerical perturbation introduced at the kth iterate of (14). Next, one has

$$\widetilde{X}_k = X_k + \Delta X_k. \quad (15)$$

Here, we perform a first-order error analysis; that is, we formally neglect quadratic terms such as $(\Delta X_k)^2$. This formal manipulation is meaningful if ΔX_k is sufficiently small. We have

$$\begin{aligned}
\widetilde{X}_{k+1} &= \frac{1}{16}\left[\widetilde{Y}_k^5 - 5\widetilde{Y}_k^3 + 15\widetilde{Y}_k + 5\widetilde{X}_k\right] \\
&= \frac{1}{16}\left[\widetilde{X}_k^{-5} - 5\widetilde{X}_k^{-3} + 15\widetilde{X}_k^{-1} + 5\widetilde{X}_k\right] \\
&= \frac{1}{16}\left[\left(X_k + \Delta X_k\right)^{-5} - 5\left(X_k + \Delta X_k\right)^{-3}\right. \\
&\qquad \left. +15\left(X_k + \Delta X_k\right)^{-1} + 5\left(X_k + \Delta X_k\right)\right] \\
&\approx \frac{1}{16}\left[\left(X_k^{-1} - X_k^{-1}\Delta X_k \cdot X_k^{-1}\right)^5\right. \\
&\qquad - 5\left(X_k^{-1} - X_k^{-1}\Delta X_k \cdot X_k^{-1}\right)^3 \\
&\qquad \left. +15\left(X_k^{-1} - X_k^{-1}\Delta X_k \cdot X_k^{-1}\right) + 5\left(X_k + \Delta X_k\right)\right],
\end{aligned} \quad (16)$$

where the following identity has been used (for any nonsingular matrix B and the matrix C):

$$(B + C)^{-1} \approx B^{-1} - B^{-1}CB^{-1}. \quad (17)$$

Note that the commutativity between X_k and ΔX_k is not used throughout this paper because it does not hold. Further simplifying yields to

$$\widetilde{X}_{k+1} \approx S + \frac{1}{2}\Delta X_k - \frac{1}{2}S\Delta X_k S, \quad (18)$$

where $S^2 = I$, $S^{-1} = S$, and for large enough k we have $X_k \approx \operatorname{sign}(A) = S$, and $(\Delta X_k)^i$, $i \ge 2$ is close to zero (matrix) and can be neglected by choosing $(\Delta X_k)^i \approx 0$. After some algebraic manipulation and using $\Delta X_{k+1} = \widetilde{X}_{k+1} - X_{k+1}$, we conclude that

$$\Delta X_{k+1} \approx \frac{1}{2}\Delta X_k - \frac{1}{2}S\Delta X_k S. \quad (19)$$

Applying (19) recursively, and after some algebraic manipulations, we have

$$\left\|\Delta X_{k+1}\right\| \le \frac{1}{2^{k+1}}\left\|\Delta X_0 - S\Delta X_0 S\right\|. \quad (20)$$

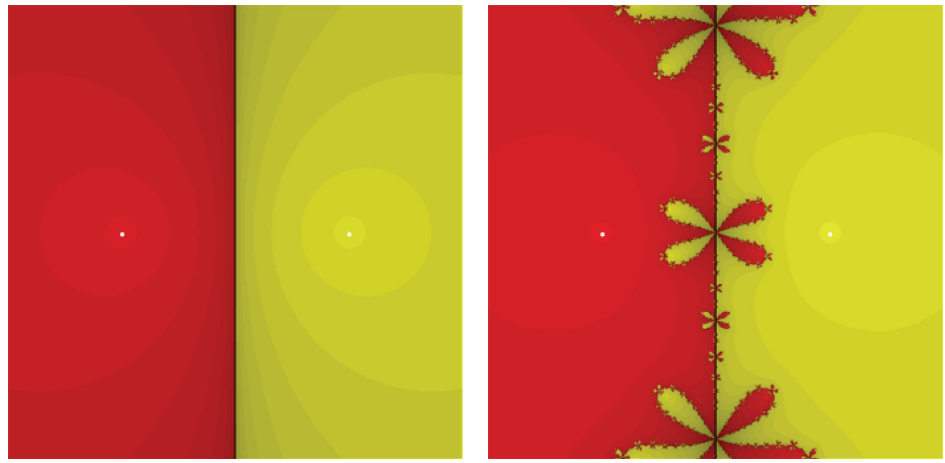
From (20), we can conclude that the perturbation at the iterate $k + 1$ is bounded. Therefore, the sequence $\{X_k\}$ generated by (14) is asymptotically stable. □

Remark 3. If X_0 is a function of A, then the iterates from (14) or (23) are all functions of A and hence commute with A.

The proposed matrix iteration requires one matrix inverse per iteration along four matrix-matrix products to achieve the convergence order four, while Halley's method requires one matrix inversion and three matrix-matrix products to reach order three.

The basins of attraction for the two iterative methods of (4) and (14) to solve $x^2 - 1 = 0$ in the complex plane have been drawn in Figure 1. It shows that the new scheme has larger basins due to its higher convergence order. Note that the roots have been identified with two white points.

Iannazzo in [20] discussed that the matrix convergence is governed by scalar convergence. That is to say, the fourth-order convergence of new method (14) might not be global unlike Newton's iteration (4). To illustrate further, if one chooses a matrix A (in Figure 1(b)) with one eigenvalue with negative real part, but in a yellow petal, then the matrix iteration will not converge to S. Therefore, it could be mentioned that scheme (14) converges with local fourth order. This restriction has encouraged us to propose another fourth-order method with global convergence in what follows and leave behind the previous iteration with interest in terms of theoretical analysis only.



(a) (b)

Figure 1: The basins of attraction for (4) (a) and fourth-order method (14) (b) for the polynomial $x^2 - 1 = 0$ (shaded by the number of iterations to obtain the solution).

Let us apply the following fourth-order nonlinear solver [16] on matrix equation (10):

$$\begin{aligned} y_k &= x_k - 2^{-1} f'(x_k)^{-1} f(x_k), \\ z_k &= x_k - f'(y_k)^{-1} f(x_k), \\ x_{k+1} &= z_k - \left[(z_k - y_k)^{-1} (f(z_k) - f(y_k))\right]^{-1} f(z_k), \end{aligned} \tag{21}$$

which reads the error equation

$$e_{k+1} = \left(\frac{c_2^3}{2} - \frac{c_2 c_3}{8}\right) e_k^4 + O\left(e_k^5\right), \tag{22}$$

wherein $c_j = f^{(j)}(\alpha)/j! f'(\alpha)$ and $e_k = x_k - \alpha$, and then to obtain its corresponding matrix iteration, as follows

$$X_{k+1} = \left(I + 18Y_k + 13Z_k\right)\left[X_k\left(7I + Y_k\right)\left(I + 3Y_k\right)\right]^{-1}, \tag{23}$$

where $Y_k = X_k X_k$, $Z_k = Y_k Y_k$, and $X_0 = A$. Figure 2 shows the basins of attraction for new method (23) and scheme (7), while both reveal global convergence. The higher order of convergence for (23) made its basins larger and lighter.

Remark 4. In this paper, we restrict the analyses to asymptotically small perturbations; that is, we use the differential error analysis.

Lemma 5. *The sequence $\{X_k\}_{k=0}^{k=\infty}$ generated by (23) is asymptotically stable.*

Proof. Using the same assumptions as in the Proof of Lemma 2 (perturbations are restricted to a neighborhood of the solution), we can write the following:

$$\begin{aligned} \widetilde{X}_{k+1} &= \left(I + 18\widetilde{Y}_k + 13\widetilde{Z}_k\right)\left[\widetilde{X}_k\left(7I + \widetilde{Y}_k\right)\left(I + 3\widetilde{Y}_k\right)\right]^{-1} \\ &= \left(I + 18\widetilde{X}_k^2 + 13\widetilde{X}_k^4\right) \\ &\quad \times \left[\left(I + 3\widetilde{X}_k\right)^{-1}\left(7I + \widetilde{X}_k\right)^{-1}\widetilde{X}_k^{-1}\right] \\ &\approx \left(I + 18\left(S + \Delta X_k S\right)^2 + 13\left(S + \Delta X_k S\right)^4\right) \\ &\quad \times \left[\left(I + 3\left(S + \Delta X_k S\right)\right)^{-1}\left(7I + \left(S + \Delta X_k S\right)\right)^{-1}\right. \\ &\qquad \left. \times \left(S + \Delta X_k S\right)^{-1}\right]. \end{aligned} \tag{24}$$

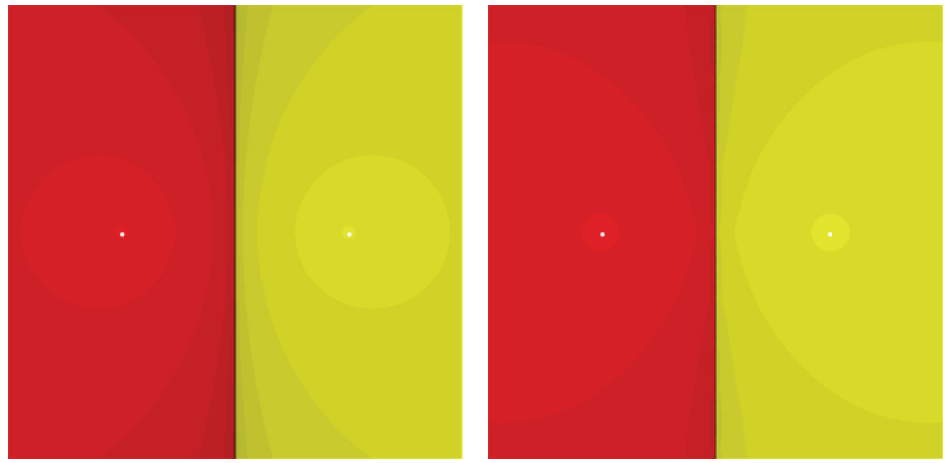
Further simplifying and by considering the terminology of Lemma 2, we attain

$$\begin{aligned} \Delta X_{k+1} &\approx \left(I + \frac{11}{8} S\Delta X_k\right)\left(S - \frac{15}{8} S\Delta X_k S - \frac{15}{8}\Delta X_k\right) \\ &\approx -\frac{1}{2}\Delta X_k - \frac{1}{2} S\Delta X_k S. \end{aligned} \tag{25}$$

It shows that the main proposed iterative scheme (23) is asymptotically stable, since recursively one has $\|\Delta X_{k+1}\| \leq (1/2^{k+1})\|\Delta X_0 + S\Delta X_0 S\|$. The proof is complete. □

Remark 6. The analysis done does not take into account the influence of the rounding errors on the convergence. Sometimes these errors may lead to slower convergence or even to the failure of the method. We remark concerning this potential danger for problems solving in single precision arithmetic or in lower precision.

Theorem 7. *Let $A \in \mathbb{C}^{n\times n}$ have no pure imaginary eigenvalues. Then, for the proposed iterates $\{X_k\}_{k=0}^{k=\infty}$ in (23), $(X_k(7I + Y_k)(I + 3Y_k))^{-1}$ is defined per stage.*



(a) (b)

FIGURE 2: The basins of attraction for (7) (a) and method (23) (b) for the polynomial $x^2 - 1 = 0$ (shaded by the number of iterations to obtain the solution).

Proof. We must show that $X_k(7I + Y_k)(I + 3Y_k)$ which is obtained at each iteration is nonsingular, since the inverse of the matrix $X_k(7I + Y_k)(I + 3Y_k)$ must be computed per computing step. Toward this goal, it is enough to show that the eigenvalues of the computed matrix at the end of each iteration are in the open half-plane.

Using the initial matrix $X_0 = A$ and based on the fact that A has no eigenvalues on the imaginary axis, the eigenvalues of the initial matrix are in the open half-plane. Let λ be the eigenvalue of the matrix X_k in the kth iterate. We have $\lambda = r \exp i\theta$, where $i = \sqrt{-1}$. Hence,

$$\begin{aligned}
&\lambda + \lambda^{-1} = \left(r + r^{-1}\right)\cos(\theta) + i\left(r - r^{-1}\right)\sin(\theta),\\
&\left(\lambda^1\left(7 + \lambda^2\right)\left(1 + 3\lambda^2\right)\right)^{-1}\\
&\quad = \frac{(\cos(\theta) - i\sin(\theta))^3}{r\left(22r^2 + \left(7 + 3r^4\right)\cos(2\theta) + i\left(-7 + 3r^4\right)\sin(2\theta)\right)}\\
&\quad = \frac{1}{7e^{i\theta}r + 22e^{3i\theta}r^3 + 3e^{5i\theta}r^5},
\end{aligned} \tag{26}$$

and consequently

$$\begin{aligned}
&\left(1 + 18\lambda^2 + 13\lambda^4\right)\left(\lambda^1\left(7 + \lambda^2\right)\left(1 + 3\lambda^2\right)\right)^{-1}\\
&= \Big((\cos(\theta) - i\sin(\theta))\\
&\quad \times \left(18r^2 + \left(1 + 13r^4\right)\cos(2\theta) + i\left(-1 + 13r^4\right)\sin(2\theta)\right)\Big)\\
&\quad \times \Big(r\left(\left(7 + r^2\right)\cos(\theta) + i\left(-7 + r^2\right)\sin(\theta)\right)\\
&\qquad \times \left(\left(1 + 3r^2\right)\cos(\theta) + i\left(-1 + 3r^2\right)\sin(\theta)\right)\Big)^{-1}.
\end{aligned} \tag{27}$$

Therefore, the eigenvalues of $X_k(7I + Y_k)(I + 3Y_k)$ remain in their open half-plane under mapping (23). And $X_k(7I + Y_k)(I + 3Y_k)$ is defined and is nonsingular for all k. □

Theorem 8. *Let $A \in \mathbb{C}^{n\times n}$ has no pure imaginary eigenvalues. Then, the proposed iterates $\{X_k\}_{k=0}^{k=\infty}$ of (23) converge to the sign matrix S.*

Proof. For our analysis, we assume that A is diagonalizable; that is, there exists a nonsingular matrix V such that

$$V^{-1}AV = \Lambda = \operatorname{diag}\left(\lambda_1, \lambda_2, \ldots, \lambda_n\right), \tag{28}$$

where $\lambda_1, \lambda_2, \ldots, \lambda_n$ are the eigenvalues of A. Note that we know that [10]

$$\operatorname{sign}(\Lambda) = \operatorname{sign}\left(V^{-1}AV\right) = V^{-1}\operatorname{sign}(A)V, \tag{29}$$

for any nonsingular matrix V. On the other hand, if we define $D_k = V^{-1}X_kV$, then we have from (23) that

$$D_{k+1} = \left(I + 18D_k^2 + 13D_k^4\right)\left[D_k\left(7I + D_k^2\right)\left(I + 3D_k^2\right)\right]^{-1}. \tag{30}$$

Notice that if D_0 is a diagonal matrix, then all successive D_k are diagonal too. From (30), it is enough to prove that $\{D_k\}$ converges to the sign of Λ and then ensure the convergence of the sequence generated by (23) to $\operatorname{sign}(A)$.

Therefore, we can write (30) as n uncoupled scalar iterations to solve $g(x) = 0$, with $g(x) = x^2 - 1$, given by

$$d_{k+1}^i = \frac{1 + 18{d_k^i}^2 + 13{d_k^i}^4}{d_k^i\left(7I + {d_k^i}^2\right)\left(I + 3{d_k^i}^2\right)}, \tag{31}$$

where $d_k^i = (D_k)_{i,i}$ and $1 \le i \le n$. On the other hand, $\operatorname{sign}(D_k) = \operatorname{sign}(\Lambda)$, for all $k \ge 0$. From (30) and (31), it is enough to study the convergence of $\{d_k^i\}$ to the sign of λ_i, for all $1 \le i \le n$.

From (31) and since the eigenvalues of A are not pure imaginary, we have that $\operatorname{sign}(\lambda_i) = s_i = \pm 1$. Thus, we attain

$$\frac{d_{k+1}^i - 1}{d_{k+1}^i + 1} = \frac{-6 + 18d_k^i - 22{d_k^i}^2 + 13{d_k^i}^3 - 3{d_k^i}^4}{8 + 18d_k^i + 22{d_k^i}^2 + 13{d_k^i}^3 + 3{d_k^i}^4}. \tag{32}$$

Since $|d_0^i| = |\lambda_i| > 0$, we obtain

$$\lim_{k\to\infty} \left| \frac{d_{k+1}^i - 1}{d_{k+1}^i + 1} \right| = 1, \tag{33}$$

and $\lim_{k\to\infty} |d_k^i| = 1 = |\operatorname{sign}(\lambda_i)|$. This shows that $\{d_k^i\}$ is convergent. Now, it could be easy to conclude that $\lim_{k\to\infty} D_k = \operatorname{sign}(\Lambda)$. Recalling $D_k = V^{-1} X_k V$, we have

$$\lim_{k\to\infty} X_k = V \left(\lim_{k\to\infty} D_k \right) V^{-1} = V \operatorname{sign}(\Lambda) V^{-1}, \tag{34}$$

and subsequently the convergence is established. The proof is complete. □

Remark 9. It is clear that convergence will be slow if either $\rho(A) \gg 1$ or A has eigenvalues close to the imaginary axis. Hence, it is better to first construct a robust seed by scaled method (9).

Theorem 10. *Let $A \in \mathbb{C}^{n\times n}$ have no pure imaginary eigenvalues. Then, new method* (23) *has fourth order to find the sign matrix S.*

Proof. Clearly, the X_k are rational functions of A and hence, like A, commute with S. On the other hand, we know that $S^2 = I$, $S^{-1} = S$, $S^{2j} = I$, and $S^{2j+1} = S$, $j \geq 1$. Choosing $B_k = X_k(7I + Y_k)(I + 3Y_k)$ (for the sake of simplicity), we have

$$\begin{aligned}
X_{k+1} - S &= \left(I + 18Y_k + 13Z_k\right) B_k^{-1} - S \\
&= \left[I + 18X_k^2 + 13X_k^4 - SB_k\right] B_k^{-1} \\
&= \left[I + 18X_k^2 + 13X_k^4 - 7X_kS - 22X_k^3S - 3X_k^5S\right] B_k^{-1} \\
&= \left[\left(X_k - S\right)^4 - 3X_k^5S^5 + 12X_k^4S^4 - 18X_k^3S \right. \\
&\qquad \left. + 12X_k^2S^2 - X_kS^3\right] B_k^{-1} \\
&= \left[\left(X_k - S\right)^4 - 3X_kS \right. \\
&\qquad \left. \times \left(X^4 - 4X_k^3S + 6X_k^2S^2 - 4X_kS^3 + I\right)\right] B_k^{-1} \\
&= \left[\left(X_k - S\right)^4 - 3X_kS\left(X_k - S\right)^4\right] B_k^{-1} \\
&= \left(X_k - S\right)^4 \left[I - 3X_kS\right] B_k^{-1}.
\end{aligned} \tag{35}$$

Now, using a matrix operator norm from both sides of (35), we attain

$$\left\|X_{k+1} - S\right\| \leq \left(\left\|B_k^{-1}\right\| \left\|I - 3X_kS\right\| \right) \left\|X_k - S\right\|^4. \tag{36}$$

This reveals the fourth order of convergence for new method (23). The proof is ended. □

4. Numerical Results

This section addresses issues related to the numerical precision of the computation of matrix sign function using Mathematica 8 built-in precision [21, 22]. The value of machine precision that produced the results included here is 15.96 digits, which corresponds to a 53-digit binary double precision number with a mantissa [23].

For numerical comparisons in this section, we have used methods (4) denoted by "Newton," (7) denoted by "Halley," (8) denoted by "M4," (14) denoted by "PM1," and (23) denoted by "PM2."

It must be noted that the Newton-Schulz iteration, which replaces the inverse of the matrix X_k in each iteration of (4) by the Schulz inverse-finder [6] and can be written as

$$X_{k+1} = \frac{1}{2} X_k \left(3I - X_k^2\right), \tag{37}$$

will not be considered in the numerical comparisons. Because due to the use of Schulz iteration instead of the matrix inversion, though the quadratic convergence of (4) will remain unchanged, it fully demands a good initial matrix and might be more risky to diverge in contrast to scheme (4). In fact, its convergence is guaranteed only if $\|I - A^2\| < 1$; see Figure 3(a).

We report the running time using the command `AbsoluteTiming[]` for the elapsed CPU time (in seconds) in the experiments. The computer specifications are Microsoft Windows XP Intel(R), Pentium(R) 4, and CPU 3.20 GHz, with 4 GB of RAM.

Example 1. The aim of this example is to compare different methods for finding the matrix sign function of a randomly generated dense 600×600 matrix as follows:

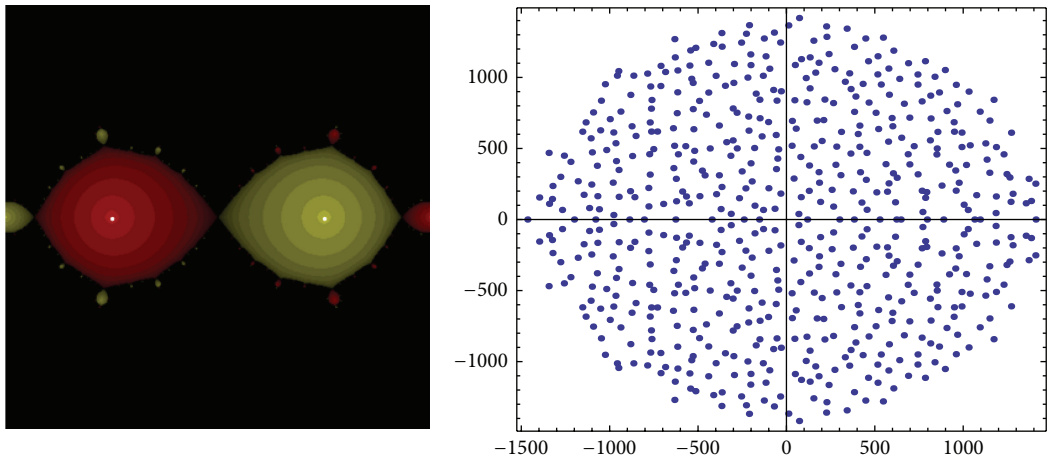
```
n = 600; SeedRandom[22];
A[1] = RandomReal[{-100, 100}, {n, n}].
```

In this test example, the prescribed tolerance is $\|X_k^2 - I\|_F \leq 10^{-8}$ and the maximum number of iterations is set to 100.

The results of comparisons in terms of the number of iterations and the computational time have been reported in Table 1, for various matrix iterations in finding the matrix sign function numerically. Note that whatever the eigenvalues of a matrix are closer to the imaginary axis, the speed of convergence for the different methods becomes slower and more risky to face with singular matrices X_k, whose inverse could not be computed; see Figure 3(b).

Note that method (13) is not competitive in terms of computational cost and local convergence and hence we will remove it from further consideration. Unlike it, new method (23) has global convergence with asymptotical stability and could be considered as an alternative over the existing iterative methods for finding S.

Example 2. In this example, 15 random dense 120×120 matrices are considered to compare the behavior of different methods in what follows:



(a) (b)

Figure 3: The basins of attraction for (37) and (a) for the complex polynomial $x^2 - 1 = 0$ (shaded by the number of iterations to obtain the solution) and the distribution of the eigenvalues of $A[1]$ for Example 1.

Table 1: Results of comparisons for Example 1.

Methods	Newton	Halley	PM1	PM2
Number of iterations	21	14	18	9
Time	6.68	7.53	14.12	7.02

Table 2: Results of comparisons for Example 3.

Methods	Newton	Halley	M4	PM2
Number of iterations	12	8	7	6
COC	1.99999	2.99561	4.04145	4.03896

```
n = 120; number = 15; SeedRandom[1];
Table[A[l] = RandomReal[{-5, 5},
{n, n}];, {l, number}].
```

For this test, the prescribed tolerance is $\|X_{k+1} - X_k\|_2 \leq 10^{-12}$ and the maximum number of iterations is set to 100.

The results of comparisons are reported in Figure 4. New method (23) beats its competitors in terms of the number of iterations, while both (23) and (8) are the best iterations in terms of the computational time. Note that, in the last two examples, we have used double precision arithmetic in our calculations.

Although this example showed the robustness of new method (23), there is an approach to observe the order of convergence of different iteration methods numerically. To be more precise, the computational order of convergence for matrix iterations in finding the matrix sign function can be estimated by

$$\mathrm{COC} = \frac{\log\left(\left\|X_{k+1}^2 - I\right\| / \left\|X_k^2 - I\right\|\right)}{\log\left(\left\|X_k^2 - I\right\| / \left\|X_{k-1}^2 - I\right\|\right)}, \tag{38}$$

wherein X_{k-1}, X_k, X_{k+1} are the last three approximations for finding S in the convergence phase.

Example 3 (an academical test). Let us find the computational order of convergence for different methods when finding the matrix sign for the well-known Wilson matrix as follows:

$$A = \begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix}. \tag{39}$$

In order to find the COC, we herein apply 64-digit fixed point arithmetic in our calculations.

The convergence history along the COCs (in the infinity norm) using formula (38) for different methods is illustrated in Figure 5 and Table 2, applying the stopping termination $\|X_k^2 - I\|_\infty \leq 10^{-16}$. Results show that new method (23) is quite fast and its computational order of convergence for academical tests in high precision computing environment is around 4.

4.1. Scaling. Main proposed iteration (23) is quite fast and reliable due to the discussions in Sections 3 and 4. However, a way is open for speeding up its initial phase of convergence.

An effective way to enhance the initial speed of convergence is to scale the iterates prior to each iteration; that is, X_k is replaced by $\mu_k X_k$. Such an approach can simply be done in what follows:

$$\begin{aligned} X_0 &= A, \\ \mu_k &= \sqrt{\frac{\left\|X_k^{-1}\right\|}{\|X_k\|}}, \\ X_{k+1} &= \left(I + 18\mu_k^2 Y_k + 13\mu_k^4 Z_k\right) \\ &\quad \times \left[\mu_k X_k \left(7I + \mu_k^2 Y_k\right)\left(I + 3\mu_k^2 Y_k\right)\right]^{-1}, \end{aligned} \tag{40}$$

where $\lim_{k\to\infty}\mu_k = 1$ and $\lim_{k\to\infty} X_k = S$.

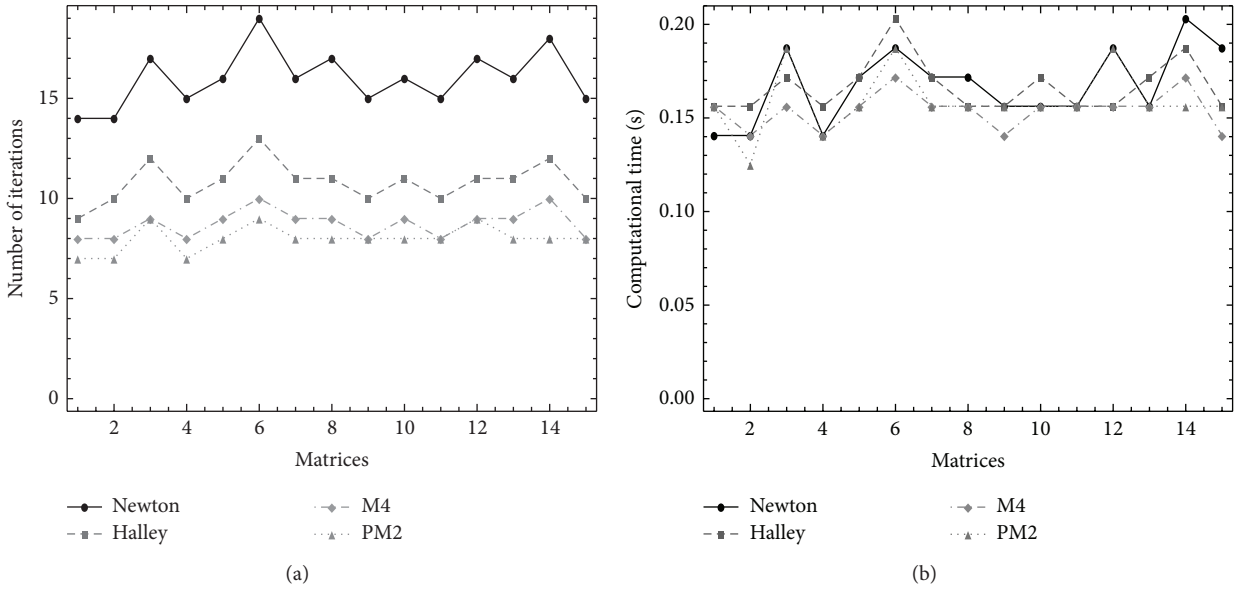


Figure 4: The comparisons of different matrix iterations in terms of the number of iterations and the computational time for 15 different test matrices in Example 2.

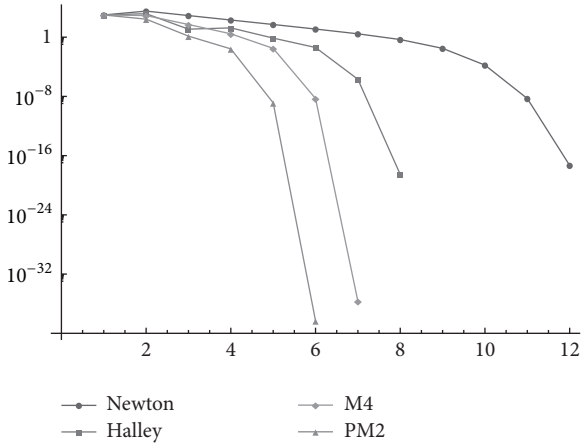


Figure 5: Convergence history based on the logarithm of the residuals $\|X_k^2 - I\|_\infty$ in Example 3.

5. Discussion

A function of a matrix can be defined and computed in several ways, such as Cauchy integral, polynomial interpolation, and Jordan canonical form. However, another approach is to use iteration methods for such computations. Several matrix functions can be computed by iteration $X_{k+1} = g(X_k)$, with an appropriate initial matrix X_0 where, for reasons of computational cost, g is usually a polynomial or a rational function.

Under this motivation, in this paper we have introduced and demonstrated some fourth-order matrix methods for finding the matrix sign function. The proposed methods consist of one matrix inversion per cycle and are asymptotically stable. The consistency and efficiency of the contributed methods have also been tested numerically for finding the matrix sign functions to support the theoretical parts.

Conflict of Interests

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

Acknowledgment

The authors are grateful to the anonymous referees for their valuable comments and suggestions for improving the paper.

References

[1] S. Barrachina, P. Benner, and E. S. Quintana-Ortí, "Efficient algorithms for generalized algebraic Bernoulli equations based on the matrix sign function," *Numerical Algorithms*, vol. 46, no. 4, pp. 351–368, 2007.

[2] F. Filbir, "Computation of the structured stability radius via matrix sign function," *Systems and Control Letters*, vol. 22, no. 5, pp. 341–349, 1994.

[3] N. Kyurkchiev and A. Iliev, "A refinement of some overrelaxation algorithms for solving a system of linear equations," *Serdica Journal of Computing*, vol. 7, no. 3, pp. 245–256, 2013.

[4] M. S. Misrikhanov and V. N. Ryabchenko, "A matrix sign function in problems of the analysis and design of linear systems," *Automation and Remote Control*, vol. 69, pp. 198–222, 2008.

[5] S. Chountasis, V. N. Katsikis, and D. Pappas, "Applications of the Moore-Penrose inverse in digital image restoration," *Mathematical Problems in Engineering*, vol. 2009, Article ID 170724, 12 pages, 2009.

[6] N. J. Higham, *Functions of Matrices: Theory and Computation*, Society for Industrial and Applied Mathematics, Philadelphia, Pa, USA, 2008.

[7] J. L. Howland, "The sign matrix and the separation of matrix eigenvalues," *Linear Algebra and its Applications*, vol. 49, pp. 221–232, 1983.

[8] D. A. Bini, N. J. Higham, and B. Meini, "Algorithms for the matrix *p*th root," *Numerical Algorithms*, vol. 39, no. 4, pp. 349–378, 2005.

[9] O. Gomilko, F. Greco, and K. Zietak, "A Padé family of iterations for the matrix sign function and related problems," *Numerical Linear Algebra with Applications*, vol. 19, no. 3, pp. 585–605, 2012.

[10] C. 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.

[11] F. Greco, B. Iannazzo, and F. Poloni, "The Padé iterations for the matrix sign function and their reciprocals are optimal," *Linear Algebra and Its Applications*, vol. 436, no. 3, pp. 472–477, 2012.

[12] C. S. Kenney and A. J. Laub, "The matrix sign function," *Institute of Electrical and Electronics Engineers: Transactions on Automatic Control*, vol. 40, no. 8, pp. 1330–1348, 1995.

[13] M. Monsalve, "A secant method for the matrix sign function," in *Lecturas en Ciencias de la Computacin*, RT 2009-03, Venezuela, 2009.

[14] J. M. McNamee and V. Y. Pan, "Efficient polynomial root-refiners: a survey and new record efficiency estimates," *Computers & Mathematics with Applications*, vol. 63, no. 1, pp. 239–254, 2012.

[15] F. Soleimani, F. Soleymani, and S. Shateyi, "Some iterative methods free from derivatives and their basins of attraction for nonlinear equations," *Discrete Dynamics in Nature and Society*, vol. 2013, Article ID 301718, 10 pages, 2013.

[16] F. K. Haghani and F. Soleymani, "On a fourth-order matrix method for computing polar decomposition," *Computational and Applied Mathematics*, 2014.

[17] M. Monsalve and M. Raydan, "A new inversion-free method for a rational matrix equation," *Linear Algebra and Its Applications*, vol. 433, no. 1, pp. 64–71, 2010.

[18] F. Soleymani, S. K. Vanani, and A. Afghani, "A general three-step class of optimal iterations for nonlinear equations," *Mathematical Problems in Engineering*, vol. 2011, Article ID 469512, 10 pages, 2011.

[19] B. Iannazzo, *Numerical solution of certain nonlinear matrix equations [Ph.D. thesis]*, Dipartimento di Matematica, Università di Pisa, 2007.

[20] B. Iannazzo, "A family of rational iterations and its application to the computation of the matrix *p*th root," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 4, pp. 1445–1462, 2008.

[21] http://reference.wolfram.com/mathematica/tutorial/LinearAlgebraInMathematicaOverview.html.

[22] M. Trott, *The Mathematica Guide-Book for Numerics*, Springer, New York, NY, USA, 2006.

[23] H. D. Noble and R. A. Chipman, "Mueller matrix roots algorithm and computational considerations," *Optics Express*, vol. 20, no. 1, pp. 17–31, 2012.