AIMS Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 8(8): 19264–19274.
DOI:10.3934/math.2023982
Received: 11 July 2022
Revised: 19 October 2022
Accepted: 07 November 2022
Published: 07 June 2023

Research article

A fast and efficient Newton-type iterative scheme to find the sign of a matrix

Malik Zaka Ullah[1], Sultan Muaysh Alaslani[1], Fouad Othman Mallawi[1], Fayyaz Ahmad[2], Stanford Shateyi[3,*] and Mir Asma[4,*]

[1] Mathematical Modeling and Applied Computation (MMAC) Research Group, Department of Mathematics, King Abdulaziz University, Jeddah 21589, Saudi Arabia

[2] Department of Mathematics, School of Science, National Textile University, Faisalabad, Pakistan

[3] Department of Mathematics and Applied Mathematics, School of Mathematical and Natural Sciences, University of Venda, P. Bag X5050, Thohoyandou 0950, South Africa

[4] Institute of Mathematical Sciences, Faculty of Science, University of Malaya, Kuala Lumpur 50603, Malaysia

* **Correspondence:** Email: stanford.shateyi@univen.ac.za, miraszaka@gmail.com.

Abstract: This work proposes a new scheme under the umbrella of iteration methods to compute the sign of an invertible matrix. To this target, a review of the exiting solvers of the same type is given and then a new scheme is derived based on a multi-step Newton-type nonlinear equation solver. It is shown that the new method and its reciprocal converge globally with wider convergence radii in contrast to their competitors of the same order from the general Padé schemes. After investigation on the theoretical parts, numerical experiments based on complex matrices of various sizes are furnished to reveal the superiority of the proposed solver in terms of elapsed CPU time.

Keywords: matrix sign function; iterative methods; fourth order; stability; basins of attraction
Mathematics Subject Classification: 65F30, 65F60

1. Introduction

The author in [10] first proposed a matrix function for the sign to be used and applied in algebraic Riccati equations. The sign function of a matrix can be defined using the following Cauchy integral definition:

$$S = \operatorname{sign}(A) = \frac{2}{\pi}\int_0^\infty (t^2 I + A^2)^{-1} dt. \tag{1.1}$$

Assume that $A \in \mathbb{C}^{n\times n}$ is given, and g is a scalar function. Then, we define $g(A)$ as a function of a matrix with the same size as the input matrix. As long as g is given on the spectrum of the input matrix [5, Chapter 5], we can find the following features for $g(A)$:

- If Y commutes with A, then Y commutes with $g(A)$,
- The eigenvalues of $g(A)$ are $g(\lambda_i)$, where λ_i stand for the eigenvalues of A,
- $g(YAY^{-1}) = Yg(A)Y^{-1}$,
- $g(A^T) = g(A)^T$,
- $g(A)$ commutes with A.

Note that both Jordan decomposition and Schur decomposition with block Parlett recursion can be used under some conditions to compute matrix functions. The matrix sign function can give us the ability to decompose an appropriate matrix into two components whose spectra lie on opposite sides of the axis of imaginary. For more information, see [3].

Practically speaking, we are motivated to derive a new Newton-type fast iteration method to find (1.1), which is efficient. That is, due to its higher order and wider convergence radii, fewer iterations will be required. This results in lower computational cost of the whole algorithm since fewer number of stopping terminations should be computed per cycle for the iterative structure. This is useful especially when the entries of the input matrix are changing a bit in a loop and each time, it is needed to find (1.1). The goal of this research is to boost convergence speed, resulting in high computational efficiency.

Here, we concentrate on functions of matrices and especially the sign function. Toward this aim, we first derive an iterative scheme for finding the matrix sign function. We derive a novel numerical procedure to find (1.1) for which the convergence rate is high via the number of matrix-matrix products while only one matrix inversion is needed per full stage.

For this research study, the remaining sections of this article are unfolded as comes next. In Section 2, iteration methods for finding (1.1) are discussed. Section 3 investigates a higher order method, which is under the umbrella of iteration methods to compute the sign of an invertible matrix. The convergence of the scheme and its asymptotical stability are brought forward. To test the correctness, efficiency, and stability, we run numerical tests on various problems in Section 4. Finally, based on the numerical results obtained, the suggested technique is determined to be efficient. Some remarks are furnished in Section 5.

2. Iteration methods in the literature

A typical procedure for finding iteration methods to calculate matrix functions is via employing a suitable nonlinear solver and selecting an initial value so as to guarantee the (local) convergence. It is here requisite to remind that the results originated from scalar nonlinear solvers do not generalize necessarily to the matrix environment. As an instance, standard convergence conditions given via the derivatives of g at a fixed point in the scalar case do not translate explicitly to similar conditions on the Frechét derivatives in the matrix environment.

The second-order Newton's method (NM) has the following structure to calculate the sign of a square invertible matrix:

$$W_{m+1} = \frac{1}{2}\left(W_m + W_m^{-1}\right), \tag{2.1}$$

where

$$W_0 = A, \tag{2.2}$$

is initial value. Authors in [7] proposed an important family of iterative schemes for finding (1.1) employing the following Padé approximants:

$$g(\zeta) = (1-\zeta)^{-1/2}. \tag{2.3}$$

Let us consider that the (m_1, m_2)-Padé approximate to $g(\zeta)$ is defined by:

$$\frac{P_{m_1}(\zeta)}{Q_{m_2}(\zeta)}, \tag{2.4}$$

where P, Q are polynomials of the approximate degree in the Padé approximation and $m_1 + m_2 \geq 1$. Then, [7] showed that the iterative scheme

$$w_{m+1} = \frac{w_m P_{m_1}(1-w_m^2)}{Q_{m_2}(1-w_m^2)} := \psi_{2m_1+1,2m_2}, \tag{2.5}$$

converges with convergence speed $m_1 + m_2 + 1$ to ±1.

By considering (2.5), the well-known locally convergent inversion-free Newton-Schulz method

$$W_{m+1} = \frac{1}{2} W_m (3I - W_m^2), \tag{2.6}$$

and Halley's iteration scheme

$$W_{m+1} = [I + W_m^2][W_m(3I + W_m^2)]^{-1}, \tag{2.7}$$

belong to this family. It is noted that Newton's method (2.1) is a member of the reciprocal of (2.5), [4].

Authors in the work [2] investigated a fourth-order family of methods to find (1.1) with the following structure:

$$\begin{aligned} W_m = {} & W_m \left((1-6v)I + 2(-7+2v)W_m^2 + (-3+2v)W_m^4 \right) \\ & \times \left[(1-2v)I - 2(3+2v)W_m^2 + (-11+6v)W_m^4 \right]^{-1}, \end{aligned} \tag{2.8}$$

wherein v is a free real parameter.

Two other fourth-order methods from (2.5) having global convergence behavior are given by:

$$W_{m+1} = [I + 6W_m^2 + W_m^4][4W_m(I + W_m^2)]^{-1}, \qquad \text{Padé [1,2]}, \tag{2.9}$$

$$W_{m+1} = [4W_m(I + W_m^2)][I + 6W_m^2 + W_m^4]^{-1}, \qquad \text{Reciprocal of Padé [1,2]}. \tag{2.10}$$

For a recent work on this topic, an interested reader may refer to [12].

3. A novel fourth-order method

3.1. A Newton-type nonlinear equation solver

Recalling that iteration methods can be constructed for matrix sign function if they can solve the matrix equation below (see, e.g., [11]):

$$G(X) := -I + X^2 = 0, \tag{3.1}$$

wherein I is the unit matrix. In the scalar case, we consider $g(x) = x^2 - 1 = 0$. To get a better background on nonlinear equation solvers and their recent findings, one may refer to [1].

Without loss of generality, consider the scalar case. Now, we propose the following multi-step Newton-type iteration scheme for solving (3.1):

$$\begin{cases} y_m = w_m - \dfrac{g(w_m)}{g'(w_m)}, \qquad m = 0, 1, \ldots, \\ h_m = w_m - \dfrac{g(w_m) - (5/4)g(y_m)}{g(w_m) - (9/4)g(y_m)} \dfrac{g(w_m)}{g'(w_m)}, \\ w_{m+1} = h_m - \dfrac{g(h_m)(h_m - w_m)}{g(h_m) - g(w_m)}. \end{cases} \tag{3.2}$$

Theorem 3.1. *Let $\xi \in D$ be a simple solution (root) of $g : D \subseteq \mathbb{C} \to \mathbb{C}$, which is a sufficiently smooth nonlinear function. Also assume the initial value w_0 is close enough to the solution. Then, the sequence generated by (3.2) has at least fourth-order convergence and satisfies the following equation of error:*

$$e_{m+1} = -\frac{k_2^3}{4} e_m^4 + \left(\frac{31}{16} k_2^4 - \frac{9}{4} k_2^2 k_3 \right) e_m^5 + O(e_m^6), \tag{3.3}$$

where $k_j = \frac{g^{(j)}(\xi)}{j! g'(\xi)}$ and $e_m = w_m - \xi$.

Proof. The proof is via Taylor expansions. See e.g., [13, 14]. Hence, it is omitted. □

3.2. Extension to matrix sign computation

By employing (3.2) to solve the scalar version of (3.1), we attain the following iterative method in a unique way:

$$W_{m+1} = W_m \left(23I + 38W_m^2 + 3W_m^4 \right) \left[5I + 42W_m^2 + 17W_m^4 \right]^{-1}, \tag{3.4}$$

having the initial value (2.2). Similarly, one may obtain the reciprocal version of (3.4) as follows:

$$W_{m+1} = \left(5I + 42W_m^2 + 17W_m^4 \right) \left[W_m \left(23I + 38W_m^2 + 3W_m^4 \right) \right]^{-1}. \tag{3.5}$$

3.3. Convergence

Now, the convergence properties of (3.5) are investigated. Recall that for any H such that $A + H$ is positive definite, we have

$$\operatorname{sign} \begin{pmatrix} 0 & A + H \\ I & 0 \end{pmatrix}, \tag{3.6}$$

as a fixed point of (3.5).

Theorem 3.2. *In computing the sign of the matrix A without any eigenvalues on the imaginary axis, consider that W_0 is sufficiently close to S or is chosen by (2.2) and commutes with A. Then, the scheme (3.5) (or equivalently (3.4)) converges to the sign matrix S, and the convergence rate is four.*

Proof. Let us use the Jordan block matrix J and decompose A using an invertible matrix T of the same size as follows:

$$A = TJT^{-1}. \tag{3.7}$$

Employing this along with the iterative method yields an iteration scheme similar to the original matrix iterative method but on the eigenvalues from the step m to the step $m+1$ as follows:

$$\lambda_{m+1}^{i} = \left(5 + 42{\lambda_m^i}^2 + 17{\lambda_m^i}^4\right)\left[\lambda_m^i\left(23 + 38{\lambda_m^i}^2 + 3{\lambda_m^i}^4\right)\right]^{-1}, \quad 1 \le i \le n, \tag{3.8}$$

where

$$s_i = \operatorname{sign}(\lambda_m^i) = \pm 1. \tag{3.9}$$

In general and after some mathematical simplifications, the expression (3.8) shows that the eigenvalues tend to $s_i = \pm 1$, that is:

$$\lim_{m\to\infty}\left|\frac{\lambda_{m+1}^i - s_i}{\lambda_{m+1}^i + s_i}\right| = 0. \tag{3.10}$$

This shows the convergence and states that the eigenvalues tends to ±1 after each iteration, and there would be eigenvalue clustering using (3.5). After studying the convergence, it is requisite to compute the convergence speed. Toward this, it is considered that:

$$\Delta_m = W_m(23I + 38W_m^2 + 3W_m^4). \tag{3.11}$$

Using (3.11), we can write:

$$\begin{aligned} W_{m+1} - S &= (5I + 42W_m^2 + 17W_m^4)\Delta_m^{-1} - S \\ &= [5I + 42W_m^2 + 17W_m^4 - S\Delta_m]\Delta_m^{-1} \\ &= [5I + 42W_m^2 + 17W_m^4 - 23W_mS - 38W_m^3S - 3W_m^5S]\Delta_m^{-1} \\ &= [-5(W_m - S)^4 + 3W_mS(W^4 - 4W_m^3S \\ &\qquad +6W_m^2S^2 - 4W_mS^3 + I)]\Delta_m^{-1} \\ &= [-5(W_m - S)^4 + 3W_mS(W_m - S)^4]\Delta_m^{-1} \\ &= (W_m - S)^4[-5I + 3W_mS]\Delta_m^{-1}. \end{aligned} \tag{3.12}$$

Using (3.12), it is possible to get that:

$$\|W_{m+1} - S\| \le \left(\| - 5I + 3W_mS\|\|\Delta_m^{-1}\|\right)\|W_m - S\|^4, \tag{3.13}$$

which shows the convergence rate of four. This completes the proof. The analysis of error for (3.4) might be similarly performed. The proof is complete now. □

We remark that $W_0 = A$ is the best choice as the initial approximation since no eigenvalues are located on the axis of imaginary and could lead to convergence. However, another sufficiently close initial approximation based on μA where μ is an appropriate non-zero parameter can be chosen as long as no eigenvalues are located on the axis of imaginary.

The presented method needs computation of matrix multiplications, as do the other competitors. On the other hand, most of the convergence theory for the presented method is via the calculation of the eigenvalues [8] from an iterate to the next one. Nonetheless, we have showed the fourth order is obtained only if the initial approximation is chosen properly.

The cost of an algorithm is important for employing it in practical problems. The convergence rate is not the only factor, and the method is useful only if it can compete with the most efficient existing solvers of the same type. When we compare (3.5) to (2.9)–(2.10), it is observed that all possess per cycle four matrix products and only one matrix inversion. It is now seen and checked that the proposed methods have wider convergence radii.

To check the global convergence of the presented scheme in contrast to the existing solvers, we may draw attraction basins of the iterative methods when solving the nonlinear equation $g(x) = x^2 - 1$ on the complex domain $[-2, 2] \times [-2, 2]$. In fact, we divide the domain into a refined mesh and test to what root, each of the mesh points converge. Results of comparisons are brought forward in Figures 1 and 2 by employing the stopping condition

$$|g(x_m)| \leq 10^{-3}. \tag{3.14}$$

The results are shaded via the number of iterates required to reach the convergence. They too show that (3.4) and (3.5), have wider convergence radii than their competitors of the same order from (2.5).

Here, although the iterative methods (2.1), and (2.9) are globally convergent, the basins for (3.4) and (3.5) contain lighter areas which indicate wider convergence radii.

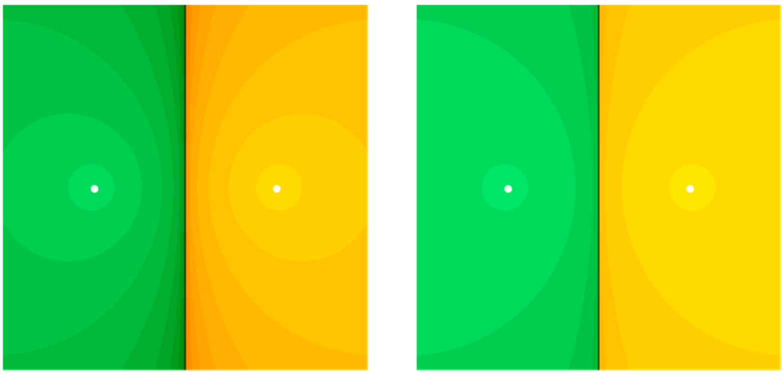

Figure 1. Attractors and convergence radii for (2.1) on left, and (2.9) on right.

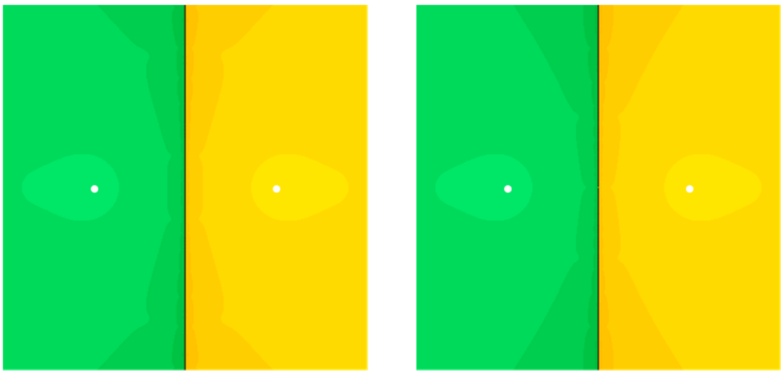

Figure 2. Attractors and convergence radii for (3.4) on left, and (3.5) on right.

The larger attraction basins given for the proposed solvers are mainly due to the appropriate selection of the parameters furnished in the second step of the nonlinear equation solver (3.2). In fact, the parameters are selected as limiting values, in which the convergence order of the solver improves to higher order, but it does not select the exact limiting value since it costs further matrix products. In this way, by keeping the number of matrix products fixed, we can derive an iterative method for finding the sign of a nonsingular matrix with larger attraction basins than the competitors from the Padé scheme.

3.4. Stability

The stability for (3.4) or similarly (3.5) are presented now. Theorem 3.3 follows from [6] on the stability of "pure matrix iterations."

Theorem 3.3. *Using (3.5) and having a nonsingular matrix A, $\{W_m\}_{m=0}^{\infty}$ with $W_0 = A$ is stable asymptotically.*

Proof. Assume that δ_m is a numerical perturbation which is produced in the m^{th} iteration. So, we can define

$$\widetilde{W}_m = W_m + \delta_m. \tag{3.15}$$

In the rest of the proof, it is assumed that $(\delta_m)^i \approx 0,\ i \geq 2$, is used, which is valid based on first order error analysis. This is rational if δ_m is sufficiently small. Now, we obtain

$$\begin{aligned}\widetilde{W}_{m+1} =&[5I + 42\widetilde{W}_m^2 + 17\widetilde{W}_m^4]\\ &\times [\widetilde{W}_m(23I + 38\widetilde{W}_m^2 + 3\widetilde{W}_m^4)]^{-1}.\end{aligned} \tag{3.16}$$

For sufficiently large m, viz., at the phase of convergence, it is assumed that

$$W_m \approx \text{sign}(A) = S, \tag{3.17}$$

where the following facts are employed (for any the matrix E and any invertible matrix H) [15, page 188]:

$$(H + E)^{-1} \simeq H^{-1} - H^{-1}EH^{-1}. \tag{3.18}$$

We used

$$S^2 = I, \tag{3.19}$$

and

$$S^{-1} = S, \tag{3.20}$$

to get that

$$\widetilde{W}_{m+1} \approx \left(S + \frac{1}{2}\delta_m - \frac{1}{2}S\delta_m S\right). \tag{3.21}$$

By further simplifications and using $\delta_{m+1} = \widetilde{W}_{m+1} - W_{m+1}$, we can find

$$\delta_{m+1} \approx \frac{1}{2}\delta_m - \frac{1}{2}S\delta_m S. \tag{3.22}$$

This can lead to the point that the method at the stage $m + 1$ is bounded, i.e., we have:

$$\|\delta_{m+1}\| \leq \frac{1}{2}\|\delta_0 - S\delta_0 S\|. \tag{3.23}$$

Therefore, the sequence $\{W_m\}_{m=0}^{\infty}$ produced via (3.4) has asymptotical stability. The proof ends now. □

4. Numerical aspects

Here, the entire implementations were performed by Mathematica 12.0 in machine precision [16, Chapter 1.4]. The iteration methods (2.1), (2.7), (2.9), (3.4) and (3.5) (denoted by NM, HM, Padé, PM1, and PM2, respectively) were compared based on the efficiency and stability on complex test matrices of various sizes. We compared the methods using the following condition:

$$E_{m+1} = \|W_{m+1}^2 - I\|_2 \leq 10^{-4}. \tag{4.1}$$

It is recalled that for (2.1), the process can be accelerated, via calculating an extra parameter per iterate and replacing W_m with $\mu_m W_m$ as follows:

$$\mu_m = \begin{cases} |\det(W_m)|^{\frac{-1}{n}}, & \text{(determinantal scaling)}, \\ \sqrt{\dfrac{\rho(W_m^{-1})}{\rho(W_m)}}, & \text{(spectral scaling)}, \\ \sqrt{\dfrac{\|W_m^{-1}\|}{\|W_m\|}}, & \text{(norm scaling)}. \end{cases} \tag{4.2}$$

Further acceleration and optimizing of (2.7) were studied in [9]. Similar investigations can be performed for the new derived iterative method in order to accelerate the scheme. Anyhow, none of such accelerations will be performed in this section for any of the compared methods in order to have a fair comparison.

Experiment 4.1. *Various complex matrices of different sizes are employed here to compare the efficiency of the existing solvers based on both the number iterations as well as the CPU time in seconds. Ten matrices are produced as follows:*

```
SeedRandom[123];
number = 10;
Table[A[l] = RandomComplex[{-5 - 5 I,
      5 + 5 I}, {100 l, 100 l}];, {l, number}];
tolerance = 10^-4;
```

Results are provided in Tables 1 and 2 showing that PM1 has the best performance against its competitors. Here, it is important to note that the convergence of the discussed iteration methods relies on an appropriate choice of initial matrices. It is observed PM1 beats all other competitors by converging to the sign as quickly as possible.

This part of the work is ended by stating that after drawing attraction basins of the new methods, such as PM1, in Figures 3 and 4 for higher degree polynomials, it can be stated that like all the other solvers, when one wants to extend NM, HM, Padé or PM1 to find the matrix sector function, then similar conditions can be imposed since they have local convergence (like the Newton's solver) to find the matrix sector function, but PM1 or PM2 enjoy higher rates of convergence, which will reduce the CPU time in practice.

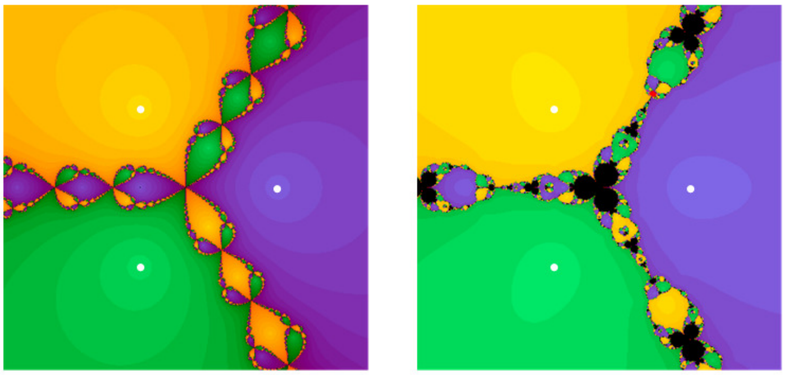

Figure 3. Attractors and convergence radii for (2.1) on left, and (3.2) on right for the equation $x^3 - 1 = 0$.

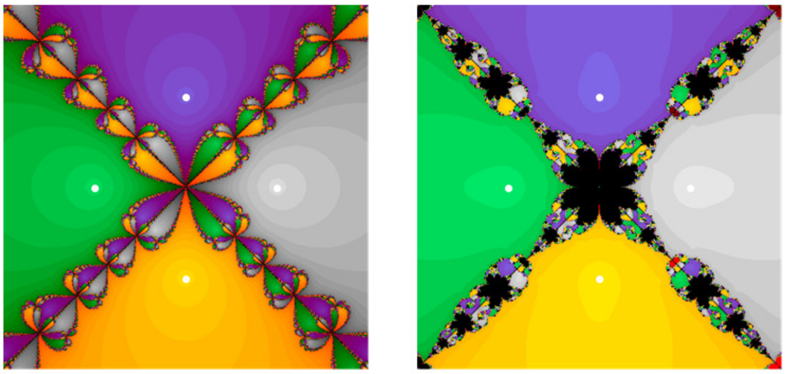

Figure 4. Attractors and convergence radii for (2.1) on left, and (3.2) on right for the equation $x^4 - 1 = 0$.

Table 1. Comparisons for the number of iterates needed to find the signs of the matrices in Experiment 4.1.

Matrices	Size	NM	HM	Padé	PM1	PM2
#1	100×100	16	11	8	7	7
#2	200×200	18	12	9	8	8
#3	300×300	16	11	8	7	7
#4	400×400	18	12	9	8	8
#5	500×500	19	12	10	9	9
#6	600×600	18	11	9	8	8
#7	700×700	18	12	9	8	8
#8	800×800	21	13	11	9	9
#9	900×900	18	12	9	8	8
#10	1000×1000	20	13	10	9	9
Mean		18.2	11.9	9.2	8.1	8.1

Table 2. Comparison of CPU times in seconds required to find the signs of the matrices in Experiment 4.1.

Matrices	Size	NM	HM	Padé	PM1	PM2
#1	100×100	0.03	0.02	0.03	0.01	0.01
#2	200×200	0.12	0.10	0.10	0.09	0.09
#3	300×300	0.28	0.29	0.25	0.27	0.26
#4	400×400	0.62	0.62	0.54	0.54	0.55
#5	500×500	1.15	1.05	1.07	1.03	1.08
#6	600×600	2.07	1.56	1.48	1.49	1.45
#7	700×700	3.13	2.41	2.23	2.08	2.16
#8	800×800	5.15	3.75	3.84	3.40	3.54
#9	900×900	5.23	4.96	4.36	4.20	4.19
#10	1000×1000	8.45	7.57	6.76	6.25	6.50
Mean		2.62	2.23	2.06	1.94	1.98

5. Concluding remarks

This paper has discussed the existing iteration methods to find the matrix sign functions. The most challenging ones arise from (2.5) with arbitrary order of convergence. Such methods are called optimal in some sense in the literature. However, in this article, we have proposed a novel Newton-type solver possessing fourth convergence speed with global convergence behavior to calculate the sign of a matrix. The new method competes with corresponding methods from the Padé family of methods.

A clear application of the sign matrix is in finding the matrix geometric mean of two Hermitian positive definite matrices, which is required mainly in the context of solving a special class of nonlinear matrix equations. For the proposed solvers in this work, (3.4) and (3.5), parallel implementations are simple, employing the available optimized programs for matrix-matrix products. Computational tests in Section 4 have been performed to show the effectiveness of the new iterative methods for a variety of complex matrices of different sizes. The mean CPU times for (3.4) and (3.5) showed better performance than the others.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research work was funded by Institutional Fund Projects under grant no. (IFPIP: 581-130-1443). The authors gratefully acknowledge technical and financial support provided by the Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

Conflict of interest

The writers state that there is no conflict of interests in this paper.

References

1. G. Candelario, A. Cordero, J. R. Torregrosa, M. P. Vassileva, An optimal and low computational cost fractional Newton-type method for solving nonlinear equations, *Appl. Math. Lett.*, **124** (2022), 107650, http://doi.org/10.1016/j.aml.2021.107650
2. A. Cordero, F. Soleymani, J. R. Torregrosa, M. Zaka Ullah, Numerically stable improved Chebyshev-Halley type schemes for matrix sign function, *J. Comput. Appl. Math.*, **318** (2017), 189–198, http://doi.org/10.1016/j.cam.2016.10.025
3. E. D. Denman, A. N. Beavers, The matrix sign function and computations in systems, *Appl. Math. Comput.*, **2** (1976), 63–94, http://doi.org/10.1016/0096-3003(76)90020-5
4. O. Gomilko, F. Greco, K. Ziętak, A Padé family of iterations for the matrix sign function and related problems, *Numer. Linear. Algebr. Appl.*, **19** (2012), 585–605, http://doi.org/10.1002/nla.786
5. N. J. Higham, *Functions of Matrices: Theory and Computation*, Philadelphia: SIAM, 2008.
6. B. Iannazzo, *Numerical Solution of Certain Nonlinear Matrix Equations*, PhD thesis, Universita degli studi di Pisa, 2007.
7. C. S. Kenney, A. J. Laub, Rational iterative methods for the matrix sign function, *SIAM J. Matrix Anal. Appl.*, **12** (1991), 273–291, http://doi.org/10.1137/0612020
8. E. M. Maralani, F. D. Saei, A. A. J. Akbarfam, K. Ghanbari, Computation of eigenvalues of fractional Sturm-Liouville problems, *Iran. J. Numer. Anal. Optim.* **11** (2021), 117–133. http://doi.org/10.22067/IJNAO.2020.11305.0
9. Y. Nakatsukasa, Z. Bai, F. Gygi, Optimizing Halley's iteration for computing the matrix polar decomposition, *SIAM J. Matrix Anal. Appl.*, **31** (2010), 2700–2720. http://doi.org/10.1137/090774999
10. J. D. Roberts, Linear model reduction and solution of the algebraic Riccati equation by use of the sign function, *Int. J. Control*, **32** (1980), 677–687. http://doi.org/10.1080/00207178008922881
11. A. R. Soheili, F. Toutounian, F. Soleymani, A fast convergent numerical method for matrix sign function with application in SDEs, *J. Comput. Appl. Math.*, **282** (2015), 167–178. http://doi.org/10.1016/j.cam.2014.12.041
12. F. Soleymani, F. W. Khdhr, R. K. Saeed, J. Golzarpoor, A family of high order iterations for calculating the sign of a matrix, *Math. Meth. Appl. Sci.*, **43** (2020), 8192–8203. http://doi.org/10.1002/mma.6471
13. F. Soleymani, Some efficient seventh-order derivative-free families in root-finding, *Opuscula Math.*, **33** (2013), 163–173. http://doi.org/10.7494/OpMath.2013.33.1.163
14. F. Soleymani, A three-step iterative method for nonlinear systems with sixth order of convergence, *Int. J. Comput. Sci. Math.*, **4** (2013), 363–373. https://doi.org/10.1504/IJCSM.2013.058057
15. G. W. Stewart, *Introduction to Matrix Computations*, New York: Academic Press, 1973.
16. M. Trott, *The Mathematica Guide-Book for Numerics*, New York: Springer, 2006.

AIMS Press