Hindawi Publishing Corporation
Discrete Dynamics in Nature and Society
Volume 2015, Article ID 649423, 11 pages
http://dx.doi.org/10.1155/2015/649423

Research Article

A Novel Iterative Method for Polar Decomposition and Matrix Sign Function

F. Soleymani,[1] Predrag S. Stanimirović,[2] and Igor Stojanović[3]

[1]*Department of Mathematics, Islamic Azad University, Zahedan Branch, Zahedan, Iran*
[2]*Faculty of Sciences and Mathematics, University of Niš, Visegradska 33, 18000 Niš, Serbia*
[3]*Faculty of Computer Science, Goce Delčev University, Goce Delčev 89, 2000 Štip, Macedonia*

Correspondence should be addressed to F. Soleymani; fazlollah.soleymani@gmail.com

Received 3 April 2015; Accepted 21 June 2015

Academic Editor: Gian I. Bischi

We define and investigate a globally convergent iterative method possessing sixth order of convergence which is intended to calculate the polar decomposition and the matrix sign function. Some analysis of stability and computational complexity are brought forward. The behaviors of the proposed algorithms are illustrated by numerical experiments.

1. Introduction

Recently, a large number of papers intended for calculating simple and multiple zeros of the nonlinear equation $f(x) = 0$ have been published. There are several attempts to overcome the difficulties in designing new iterative methods for solving $f(x) = 0$ or in the application of such iterative methods in computing matrix functions (see [1, 2]). Our current work discusses an application of a high order iterative method for solving nonlinear scalar equations in calculating the polar decomposition and the matrix sign function.

We now restate some preliminaries about the polar decomposition and the matrix sign.

Let $\mathbb{C}^{m\times n}$ denote the linear space of all $m\times n$ complex matrices. The polar decomposition of a complex matrix $A \in \mathbb{C}^{m\times n}$ is defined as

$$\begin{aligned} A &= UH, \\ U^*U &= I_r, \\ \operatorname{rank}(U) &= r = \operatorname{rank}(A), \end{aligned} \tag{1}$$

wherein H is a Hermitian positive semidefinite matrix of the order n and $U \in \mathbb{C}^{m\times n}$ is a subunitary matrix (partial isometry). Here, the inequality $m \geq n$ is assumed. A matrix U is subunitary if $\|Ux\|_2 = \|x\|_2$ for any $x \in \mathscr{R}(U^*) = \mathscr{N}(U)^{\perp}$, where $\mathscr{R}(X)$ and $\mathscr{N}(X)$ denote the linear space spanned by columns of the matrix X (range of X) and the null space of X, respectively. Note that if $\operatorname{rank}(A) = n$, then $U^*U = I_n$, and U is an orthonormal Stiefel matrix.

It is assumed that the singular value decomposition (SVD) of A has the following form:

$$A = P\begin{pmatrix}\Sigma\\ 0\end{pmatrix}Q^*, \tag{2}$$

wherein $P \in \mathbb{C}^{m\times m}$ and $Q \in \mathbb{C}^{n\times n}$ are unitary matrices and

$$\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_n), \quad \sigma_1 \geq \cdots \geq \sigma_n \geq 0. \tag{3}$$

The Hermitian factor H is always unique and it can be expressed as $(A^*A)^{1/2}$, and the unitary factor is unique if A is nonsingular. Note that, here, the exponent $1/2$ denotes the principal square root: the one whose eigenvalues lie in the right half-plane.

The polar decomposition may be easily constructed from an SVD of the given matrix A. However, the SVD is a substantial calculation that displays much more about the structure of A than does the polar decomposition. Constructing the polar decomposition from the SVD destroys this extra information and wastes the arithmetic work used to compute it. It is intuitively more appealing to use the polar decomposition as a preliminary step in the computation of the SVD.

On the other hand, a related issue to the polar decomposition is the matrix sign decomposition, which is defined for a matrix $A \in \mathbb{C}^{n\times n}$ having no pure imaginary eigenvalues. The most concise definition of the matrix sign decomposition is given in [3]

$$A = SN = A\left(A^2\right)^{-1/2}\left(A^2\right)^{1/2}. \tag{4}$$

Here, $S = \mathrm{sign}(A)$ is the matrix sign function, introduced by Roberts [4]. We herewith remark that, in this work, whenever we write about the computation of a matrix sign function, we mean a square matrix with no eigenvalues on the imaginary axis.

Now we briefly review some of the most important iterative methods for computing the matrix polar decomposition.

Among many iterations available for computing the polar decomposition, the most practically useful is the scaled Newton iteration [5] as well as the recently proposed dynamically weighted Halley iteration [6]. The method of Newton introduced for computing the polar decomposition (via unitary polar factor) is defined in [5] by the iterative scheme

$$U_{k+1} = \frac{1}{2}\left(U_k + U_k^{-*}\right), \tag{5}$$

for the square nonsingular cases and by the following alternative for general rectangular cases [7]:

$$U_{k+1} = \frac{1}{2}\left(U_k + U_k^{\dagger *}\right). \tag{6}$$

Note that $U^{\dagger}$ stands for the Moore-Penrose generalized inverse, $U_k^{-*} = (U_k^{-1})^*$, and $U_k^{\dagger *} = (U_k^{\dagger})^*$.

The cubically convergent method of Halley

$$U_{k+1} = \left[U_k\left(3I + U_k^* U_k\right)\right]\left[I + 3U_k^* U_k\right]^{-1} \tag{7}$$

has been developed in [8] for computing the unitary polar factor.

The particular formula (7) is also applicable to singular or rectangular matrices. This scheme has further been developed adaptively in [6]. In fact the dynamically weighted Halley method (DWH) for computing unitary polar factor has been introduced as follows:

$$U_{k+1} = \left[U_k\left(a_k I + b_k U_k^* U_k\right)\right]\left[I + c_k U_k^* U_k\right]^{-1}. \tag{8}$$

The parameters a_k, b_k, and c_k are dynamically chosen to accelerate the convergence. They are computed by

$$\begin{aligned} a_k &= h\left(l_k\right), \\ b_k &= \frac{\left(a_k - 1\right)^2}{4}, \\ c_k &= a_k + b_k - 1, \end{aligned} \tag{9}$$

where

$$\begin{aligned} h(l) &= \sqrt{1+\gamma} + \frac{1}{2}\sqrt{8 - 4\gamma + \frac{8\left(2 - l^2\right)}{l^2\sqrt{1+\gamma}}}, \\ \gamma &= \left(\frac{4\left(1 - l^2\right)}{l^4}\right)^{1/3}. \end{aligned} \tag{10}$$

In (9), l_k is a lower bound for the smallest singular value of U_k. Fortunately, once $l_0 \le \sigma_{\min}(U_0)$ is obtained, then effective and sharp bounds can be attained at no cost from the following recurrence:

$$l_k = l_{k-1}\frac{a_{k-1} + b_{k-1}l_{k-1}^2}{1 + c_{k-1}l_{k-1}^2}, \quad k \ge 1. \tag{11}$$

An initial matrix U_0 must be employed in a matrix fixed-point type method so as to arrive at the convergence phase. Such a sharp initial approximation for the unitary factor can be expressed as

$$U_0 = \frac{1}{\alpha}A, \tag{12}$$

whereas $\alpha > 0$ is an estimate of $\|A\|_2$ (a safe choice of which is $\|A\|_F$).

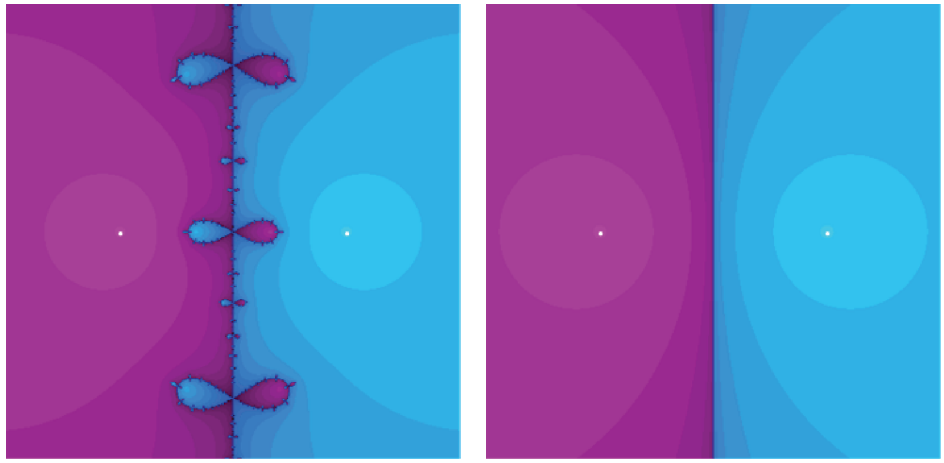
The rest of this paper unfolds the contents in what follows. Section 2 derives a new iterative scheme for solving nonlinear equations. Additionally, by applying the illustration via basins of attraction, we find a scheme with a global convergence. The global convergence of the constructed solver is verified analytically. In Section 3, we generalize the proposed nonlinear equation solver into the iterative method for finding the polar decomposition. Furthermore, we prove that the new scheme is convergent. The computational complexity is discussed in Section 4. In Section 5, we define an extension of the proposed method in numerical computation of the matrix sign function and show its asymptotic stability. Section 6 is devoted to the application of the contributed methods in solving two numerical examples (one in double precision arithmetic and the other in a high precision computing environment). Finally, Section 7 draws a conclusion of this paper.

2. Chebyshev-Halley Type Iteration and Its Extension

Many of the iterative methods for computing matrix functions can be deduced by applying a nonlinear equation solver to a special mapping. For example, applying Newton's method to the mapping

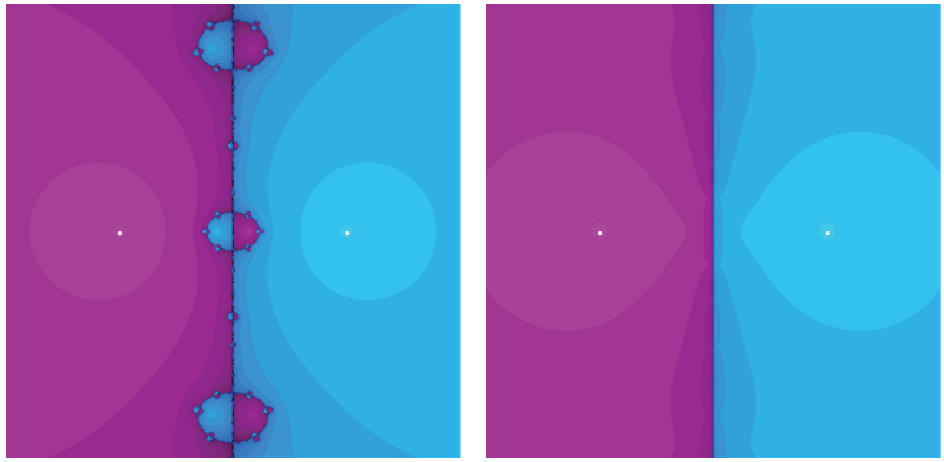
$$F(U) := U^*U - I = 0, \tag{13}$$

in which I is the identity matrix of the appropriate size, could result in iterates (5) (note that the equivalent form of (13) for the matrix sign is $U^2 - I = 0$). This reveals a close relation between matrix functions and iterative root-finding methods [3].



(a) $\eta = 0$ (b) $\eta = 1/2$

Figure 1: Basins of attraction for different methods of family (14) for $f(x) = x^2 - 1$ shaded according to the number of iterations.



(a) $\eta = 1/3$ (b) $\eta = 3/4$

Figure 2: Basins of attraction for different methods of family (14) for $f(x) = x^2 - 1$ shaded according to the number of iterations.

Gutiérrez and Hernández in [9] developed a Chebyshev-Halley type scheme in Banach space for finding simple zeros of $f(x) = 0$, which can be written in what follows:

$$x_{k+1} = x_k - \left(1 + \frac{1}{2}\left(\frac{L(x_k)}{1 - \eta L(x_k)}\right)\right)\frac{f(x_k)}{f'(x_k)}, \qquad (14)$$

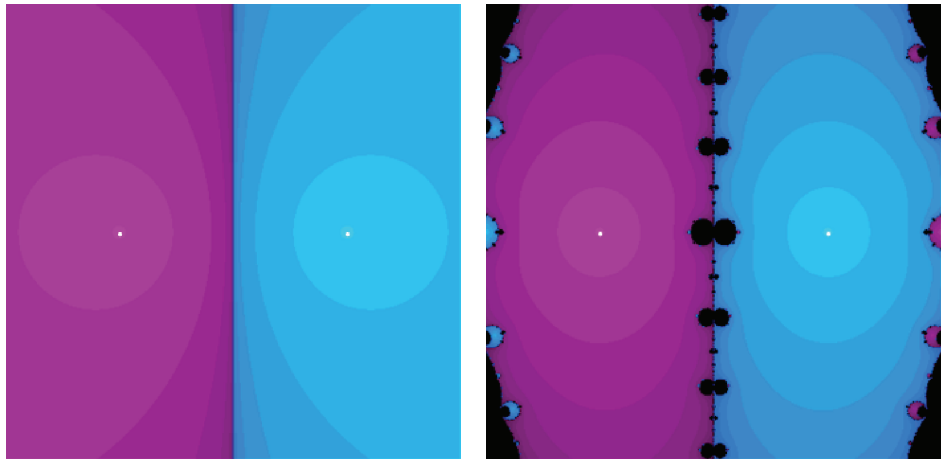
wherein $\eta \in \mathbb{R}$, $L(x_k) = f''(x_k)f(x_k)/f'(x_k)^2$, and the convergence order is cubic.

If we decide to apply (14) for solving (13), we will obtain a family of at least cubically convergent schemes in finding the polar decomposition. But an important barrier occurred and it would be the nonglobal convergence of some of the schemes.

On the other hand, Iannazzo in [10] showed that the matrix convergence is governed by the scalar convergence *for the so-called pure matrix iterations*. We remark that this is true for the matrix sign function when the scalars are the eigenvalues, while for polar decomposition it is true when scalars are singular values. This is also well illustrated in the textbook [3].

Hence, we should find a member from (14), so that the derived method is new and also possesses the global convergence. Toward this goal, we employ the theory of basins of attraction for (14) so as to solve the quadratic polynomial $x^2 - 1 = 0$ in a square $[-2, 2] \times [-2, 2]$ of the complex plane whereas the maximal number of iterations is fixed to 100 and the stopping criterion is $|f(x_k)| \leq 10^{-4}$.

This is done in Figures 1–3 for different values of η. In Figures 1(a) and 2(a), the convergence in the process of solving $x^2 - 1 = 0$ is local, and therefore convergence of the matrix iterations could happen only for very sharp initial matrices. Moreover, the method in Figure 3(b) behaves chaotically and



(a) $\eta = 3/2$ (b) $\eta = 5/2$

FIGURE 3: Basins of attraction for different methods of family (14) for $f(x) = x^2 - 1$ shaded according to the number of iterations.

also includes divergence points. The divergence points are included in the black areas and, accordingly, should be outside of interest.

We remark that η has been chosen randomly and experimentally. Clearly, the investigation of the other cases can be used in future studies.

Among the six tested methods extracted by different particular values of η and illustrated in Figures 1–3, only the methods illustrated in Figures 1(b), 2(b), and 3(a) have a global convergence. A deeper verification reveals that the methods used in Figures 1(b) and 3(a) are in fact Halley's method and its reciprocal for $x^2 - 1 = 0$, respectively. Since those are not new, the only novel and useful method which has global convergence and remains from the six considered methods of (14) is the iterative method used in Figure 2(b). It can be written as follows:

$$x_{k+1} = \frac{1 + 12x_k^2 + 3x_k^4}{6x_k + 10x_k^3}. \tag{15}$$

To increase the order of convergence, let us now compose the method of Newton with iterations (14) corresponding to $\eta = 3/4$ in order to derive a useful high order globally convergent iterative scheme for solving $f(x) = 0$. This intention leads to

$$\begin{aligned} y_k &= x_k - \frac{f(x_k)}{f'(x_k)}, \\ x_{k+1} &= y_k - \left(1 + \frac{1}{2}\left(\frac{L(y_k)}{1 - (3/4)L(y_k)}\right)\right)\frac{f(y_k)}{f'(y_k)}, \end{aligned} \tag{16}$$

wherein $L(y_k) = f''(y_k)f(y_k)/f'(y_k)^2$.

Theorem 1. *Let $\alpha \in D$ be a simple zero of a sufficiently differentiable function $f : D \subseteq \mathbb{R} \to \mathbb{R}$ for an open interval D, which contains x_0 as an initial approximation of α. Then the iterative expression (16) without memory has a sixth order of convergence.*

Proof. To save the space and in order not to deviate from the main topic, we here exclude the proof. In addition, the steps of the proof are similar to those taken in [11]. □

The iterative method (16) reaches the sixth-order convergence using five functional evaluations and thus achieves the efficiency index $6^{1/5} \approx 1.430$, which is higher than that of Newton; that is, $2^{1/2} \approx 1.414$. We remark that the cost of one function evaluation and its first and second derivatives are assumed to be unity. Furthermore, the application of (16) in solving the polynomial equation $g(x) \equiv x^2 - 1 = 0$ possesses the global convergence (except for the points lying on the imaginary axis). This is illustrated analytically in what follows.

In terms of the fractal theory, it is necessary to find the global basins of attraction for a zero z^*:

$$S(z^*) := \{z \in C : \Upsilon_k(z) \longrightarrow z^*, \text{ as } k \longrightarrow \infty\}, \tag{17}$$

where $\Upsilon_k(x) = \Upsilon(\Upsilon(\cdots(\Upsilon(x))))$ is the k-fold composition $\Upsilon \circ \cdots \circ \Upsilon$ of the iteration function Υ. Here using (16), we have (in its reciprocal form)

$$x_{k+1} = I(x_k) = \frac{20x_k + 108x_k^3 + 108x_k^5 + 20x_k^7}{3 + 60x_k^2 + 130x_k^4 + 60x_k^6 + 3x_k^8}, \quad k = 0, 1, \ldots. \tag{18}$$

To check the global convergence of (16) in the case of the quadratic polynomial $g(z) = z^2 - 1$, with the zeros ± 1, we start from

$$B(z) = \frac{20z + 108z^3 + 108z^5 + 20z^7}{3 + 60z^2 + 130z^4 + 60z^6 + 3z^8} \tag{19}$$

and find

$$\frac{B(z)+1}{B(z)-1} = \lambda \left(\frac{U+1}{U-1}\right)^6, \tag{20}$$

wherein $\lambda = -(3 + U(2 + 3U))/(3 + U(-2 + 3U))$.

Let ∂S denote the boundary of the set S. One of basic notions in the fractal theory connected to iterative processes and convergence of an iterative function f is a Julia set for the proposed operator Υ. Thus, when $k \to \infty$, we obtain the following:

(1) If $|(z+1)/(z-1)| < 1$, then $|(B_k(z)+1)/(B_k(z)-1)| \to 0$, and $B_k(z) \to -1$.

(2) If $|(z+1)/(z-1)| > 1$, then $|(B_k(z)+1)/(B_k(z)-1)| \to 0$, and $B_k(z) \to +1$.

Furthermore, the basins of attraction $S(-1)$ and $S(1)$ in the case of the operator B are the half-planes on either side in relation to the line $z = 0$ (the imaginary axis). Since ± 1 are attractive fixed points of B, the Julia set $J(B)$ is the boundary of the basins of attraction $S(-1)$ and $S(1)$; that is,

$$J(B) = \partial S(-1) = \partial S(1) = \{\gamma i : \gamma \in \mathbb{R}\}. \tag{21}$$

The Julia set $J(B)$ is just the line $z = 0$ for (19), and thus the new sixth-order method (18) is globally convergent. Therefore, the presented method has global behavior, even outside the square $[-2, 2] \times [-2, 2]$ which is considered in Figure 1.

In addition, we remark that a globally convergent sixth-order method can be easily constructed by composing standard Halley and Newton methods.

The main contribution of this work lies in the next two sections at which we extend (16) into the iterative methods for computing the polar decomposition and the matrix sign function.

3. Extension to Polar Decomposition

This section is devoted to the extension of (16) to solve the matrix equation (13). This application enables us to obtain a fast sixth-order iterative matrix method for constructing the polar decomposition. Note that the improvements in hardware and software have been ultimately indispensable, since higher order methods produce approximations of great accuracy and require complicated convergence analysis, which is feasible only by symbolic computation. Subsequently, in this paper, we use Mathematica [12] to illustrate the speed of convergence.

To this goal, an application of (16) on (13) in conjunction with further simplifications produces the following reciprocal form as the rational iteration:

$$\begin{aligned} U_{k+1} &= U_k \left[20I + 108Y_k + 108Z_k + 20W_k\right] \\ &\quad \cdot \left[3I + 60Y_k + 130Z_k + 60W_k + 3T_k\right]^{-1}, \end{aligned} \tag{22}$$

where

$$\begin{aligned} Y_k &= U_k^* U_k, \\ Z_k &= Y_k Y_k, \\ W_k &= Z_k Y_k, \\ T_k &= W_k Y_k, \end{aligned} \tag{23}$$

and U_0 is given by (12).

Now, we have a novel iterative fixed-point type method for finding the polar decomposition via calculating the unitary matrix U.

Theorem 2. *Assume that $A \in \mathbb{C}^{m\times n}$ is an arbitrary matrix. Then, the matrix iterates $\{U_k\}_{k=0}^{k=\infty}$ of (22) converge to U using $U_0 = A$.*

Proof. To prove the statement, we make use of the SVD of A in the form $A = P\Sigma Q^*$, where $\Sigma = \left(\begin{smallmatrix} \Sigma_r & 0 \\ 0 & 0 \end{smallmatrix}\right)$, $r = \operatorname{rank}(A)$ and the zero blocks in Σ may be absent. Note that the steps of the proof are similar to those which recently have been taken in [2]. Define

$$D_k = P^* U_k Q. \tag{24}$$

Subsequently, from (22), we have

$$\begin{aligned} D_0 &= \Sigma, \\ D_{k+1} &= \left[20D_k + 108D_k^3 + 108D_k^5 + 20D_k^7\right] \\ &\quad \cdot \left[3I + 60D_k^2 + 130D_k^4 + 60D_k^6 + 3D_k^8\right]^{-1}. \end{aligned} \tag{25}$$

Since $D_0 \in \mathbb{R}^{m\times n}$ has diagonal and zero elements, it follows by the induction that the sequence $\{D_k\}_{k=0}^{\infty}$ is defined by

$$D_k = \begin{pmatrix} \operatorname{diag}\left(d_i^{(k)}\right) & 0 \\ 0 & 0 \end{pmatrix}, \quad d_i^{(k)} > 0. \tag{26}$$

Accordingly, (25) represents r uncoupled scalar iterations

$$\begin{aligned} d_i^{(0)} &= \sigma_i, \quad 1 \le i \le r, \\ d_i^{(k+1)} &= \left[20d_i^{(k)} + 108{d_i^{(k)}}^3 + 108{d_i^{(k)}}^5 + 20{d_i^{(k)}}^7\right] \\ &\quad \cdot \left[3 + 60{d_i^{(k)}}^2 + 130{d_i^{(k)}}^4 + 60{d_i^{(k)}}^6 + 3{d_i^{(k)}}^8\right]^{-1}. \end{aligned} \tag{27}$$

Simple manipulations yield the relation

$$\frac{d_i^{(k+1)}-1}{d_i^{(k+1)}+1}=\frac{-3+20d_i^{(k)}-60{d_i^{(k)}}^2+108{d_i^{(k)}}^3-130{d_i^{(k)}}^4+108{d_i^{(k)}}^5-60{d_i^{(k)}}^6+20{d_i^{(k)}}^7-3{d_i^{(k)}}^8}{3+20d_i^{(k)}+60{d_i^{(k)}}^2+108{d_i^{(k)}}^3+130{d_i^{(k)}}^4+108{d_i^{(k)}}^5+60{d_i^{(k)}}^6+20{d_i^{(k)}}^7+3{d_i^{(k)}}^8}. \tag{28}$$

Since σ_i is positive, (28) holds for each i. It follows that $|(d_i^{(k+1)}-1)/(d_i^{(k+1)}+1)|\to 0$ as $k\to\infty$; that is to say,

$$D_k\longrightarrow\begin{pmatrix} I_r & 0\\ 0 & 0\end{pmatrix}. \tag{29}$$

Therefore, as $k\to\infty$, $U_k\to U_rV_r^*=U$. The proof is complete. □

Theorem 3. *Let $A\in\mathbb{C}^{m\times n}$ be an arbitrary matrix. Then, method (22) has sixth order of convergence to find the unitary polar factor of A.*

Proof. The proposed scheme (22) transforms the singular values of U_k according to

$$\begin{aligned}\sigma_i^{(k+1)}&=\left[20\sigma_i^{(k)}+108{\sigma_i^{(k)}}^3+108{\sigma_i^{(k)}}^5+20{\sigma_i^{(k)}}^7\right]\\&\cdot\left[3+60{\sigma_i^{(k)}}^2+130{\sigma_i^{(k)}}^4+60{\sigma_i^{(k)}}^6+3{\sigma_i^{(k)}}^8\right]^{-1},\\&\qquad 1\le i\le r,\end{aligned} \tag{30}$$

and leaves the singular vectors invariant. From (30), it is enough to show that the convergence of the singular values to unity has sixth order of convergence for $k\ge 1$:

$$\frac{\sigma_i^{(k+1)}-1}{\sigma_i^{(k+1)}+1}=-\frac{\left(-1+\sigma_i^{(k)}\right)^6\left(3+\sigma_i^{(k)}\left(-2+3\sigma_i^{(k)}\right)\right)}{\left(1+\sigma_i^{(k)}\right)^6\left(3+\sigma_i^{(k)}\left(2+3\sigma_i^{(k)}\right)\right)}. \tag{31}$$

Now, we attain

$$\left|\frac{\sigma_i^{(k+1)}-1}{\sigma_i^{(k+1)}+1}\right|\le\left(\left|\frac{3+\sigma_i^{(k)}\left(-2+3\sigma_i^{(k)}\right)}{3+\sigma_i^{(k)}\left(2+3\sigma_i^{(k)}\right)}\right|\right)\left|\frac{\sigma_i^{(k)}-1}{\sigma_i^{(k)}+1}\right|^6. \tag{32}$$

This reveals the sixth order of convergence for the new method (22). The proof is ended. □

Note that, in 1991, Kenney and Laub [13] have proposed a family of rational iterative methods for the matrix sign, based on Padé approximation. Their principal Padé iterations are convergent globally. Thus, we have convergent methods of arbitrary orders for the matrix sign (subsequently for the polar decomposition). However, here we tried to propose another new and useful method for this purpose.

We now end this section by recalling an important approach for speeding up the convergence speed of (22). The sixth-order convergence for iteration (22) ensures rapid convergence in the final stages of the iterates. The speed of convergence can be slow at the beginning of the process, so it is necessary to scale the matrix U_k before each cycle. An important scaling approach was derived in [14] in the Frobenius norm as comes next:

$$\theta_k=\left(\frac{\left\|U_k^\dagger\right\|_F}{\left\|U_k\right\|_F}\right)^{1/2}. \tag{33}$$

See [15] for more details. We remark that numerical experiments show improvements in the speed of convergence applying (33).

Remark 4. The new scheme can be expressed in the following accelerated form:

$$\begin{aligned}&\text{Compute } \theta_k \text{ by } (33),\quad k=0,1,\dots,\\&M_k=\left[3I+60\theta_k^2Y_k+130\theta_k^4Z_k+60\theta_k^6W_k+3\theta_k^8T_k\right],\\&U_{k+1}=\theta_kU_k\left[20I+108\theta_k^2Y_k+108\theta_k^4Z_k+20\theta_k^6W_k\right]\\&\quad\cdot M_k^{-1}.\end{aligned} \tag{34}$$

4. Computational Complexity

In this section, we evaluate the computational efficiency of (22). To compare the behavior of different matrix methods for finding U, we recall the definition of efficiency index:

$$\mathrm{EI}=\mathfrak{p}^{1/\mathscr{C}}, \tag{35}$$

wherein $\mathscr{C}$ and $\mathfrak{p}$ denote the computational cost and the convergence order per cycle. Here, in order to have a fair comparison and since there are matrix-matrix multiplications (denoted by mmm) and matrix inversion(s) per computing cycles of (6), (7), and (22), we extend (35) as follows:

$$\mathrm{CEI}=\mathfrak{p}^{1/s(m+c)}, \tag{36}$$

so as to be able to incorporate all the existing factors of an algorithm into the definition of the computational efficiency index. In (36), s, m, and c denote the whole number of iterations required by a method to converge to U, the number of mmm per cycle, and the considered cost of matrix inversion per cycle, respectively.

On the other hand, it is assumed that the cost of mmm is 1 unit and subsequently the cost of one matrix inversion is 1.2

(1) U_0 is given
(2) use (22) until $\|U_{k+1} - U_k\|_\infty / \|U_k\|_\infty \leq \zeta$
(3) If the final stop termination $\|U_{k+1} - U_k\|_\infty / \|U_k\|_\infty \leq \epsilon$ has already been occurred, go to (5), else
(4) do (6) to fulfill the stop termination $\|U_{k+1} - U_k\|_\infty / \|U_k\|_\infty \leq \epsilon$
(5) end for

Algorithm 1: The hybrid algorithm for computing polar decomposition.

and the cost for computing the Moore-Penrose inverse is 1.4. We considered these weights empirically after testing many random matrices on a typical PC. We also neglect the cost of additions, subtractions, and so forth, because their costs are negligible in contrast to the cost of mmm and matrix inversion.

Now, the *approximated* CEI for the studied methods would be

$$\begin{aligned} \mathrm{CEI}_{(6)} &\approx 2^{1/s_1(0+1(1.4))}, \\ \mathrm{CEI}_{(7)} &\approx 3^{1/s_2(3+1(1.2))}, \\ \mathrm{CEI}_{(22)} &\approx 6^{1/s_3(6+1(1.2))}, \end{aligned} \tag{37}$$

wherein s_1, s_2, and s_3 are the whole number of iterations required by (6), (7), and (22) to converge, respectively, in the same environment.

Finally, we assume that $s_3 = s$ and thus the number of iterations for (6) and (7) would *roughly* be $(5/2)s - 1$ and $(5/3)s - 1$, respectively, since they have the second and the third orders of convergence in contrast to the sixth order for (22). We have obtained these factors empirically via solving many numerical examples.

The results of comparisons have now been drawn and illustrated in the bar chart of Figure 4. We here remark that NMP, HMP, and PMP stand for Newton's method for polar decomposition, Halley's method for polar decomposition, and proposed method for polar decomposition. Such naming will be used from now on. In this figure, one may easily observe that when a higher number of iterations are required (occasionally for large scale matrices), then the new method performs much better, because it requires smaller number of matrix inversions. In Figure 4, for example, $s = 4$ means that the number of steps required for the convergence of (22) is 4 and subsequently around $s_1 = 9$ and $s_2 = 6$ steps are required for (6) and (7) to converge. Note that we plotted the logarithm of CEI in Figure 4 to show the distinctions obviously.

The only difficulty in our high order method is that if, in one iteration, it produces results of a lower accuracy (a low tolerance, e.g., $\zeta = 10^{-1}$), then an expensive further cycle should be carried out to reach the tolerance in double precision (e.g., 10^{-8}). In our proposed algorithm (ALP, i.e., Algorithm for polar decomposition) hereby, we first apply (22) to arrive at the convergence phase much rapidly and accelerate the beginning of the process. And next, we apply the simple method (6) for the last step only. This switching idea has been illustrated in Algorithm 1.

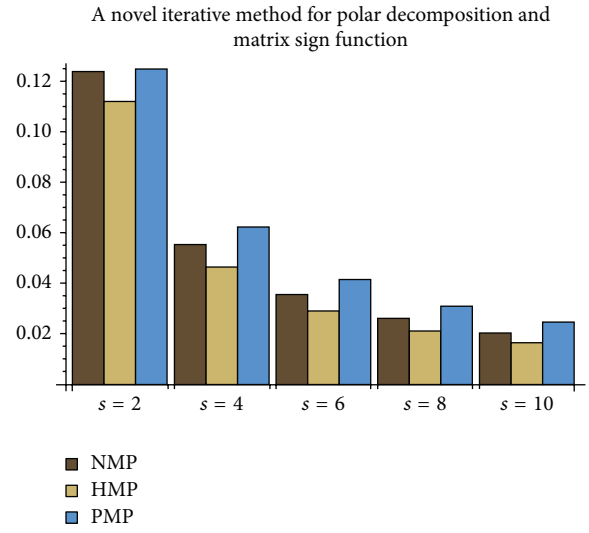


Figure 4: Comparison of computational efficiency indices for different matrix methods in finding the unitary polar factor.

5. Extension for Matrix Sign

Herein, we show an application of the proposed iterative method (16) in finding the matrix sign function. In fact, there is a tight relationship between the matrix sign and the polar decomposition. As introduced in Section 1, the sign of a nonsingular square matrix is an important matrix function with potential applications in different branches of Mathematics; see [16, 17].

The iterative rule

$$X_{k+1} = \frac{1}{2}\left(X_k + X_k^{-1}\right), \tag{38}$$

which is also known as the Newton method, is the most common and well-known way for finding the sign of a square nonsingular matrix. It converges quadratically when $X_0 = A$ has been chosen as the initial matrix.

A general family of matrix iterations for finding the matrix sign function S was discussed thoroughly in [13]. An example from this class of methods is the method of Halley which is a modification of (7) and is defined by

$$X_{k+1} = \left[X_k\left(3I + X_k^2\right)\right]\left[I + 3X_k^2\right]^{-1}. \tag{39}$$

(1) X_0 is given
(2) Use (40) until $\|X_{k+1} - X_k\|_\infty / \|X_k\|_\infty \leq 10^{-1}$
(3) If the final stop termination $\|X_{k+1} - X_k\|_\infty / \|X_k\|_\infty \leq \epsilon$ has already been occurred, go to (5), else
(4) do (38) to fulfill the stop termination $\|X_{k+1} - X_k\|_\infty / \|X_k\|_\infty \leq \epsilon$
(5) end for

Algorithm 2: The hybrid algorithm for computing matrix sign function.

Accordingly, the new scheme (22) can simply be used for finding the matrix sign function S with a little modification as comes next:

$$X_{k+1} = X_k \left[20I + 108X_k^2 + 108X_k^4 + 20X_k^6\right] \cdot \left[3I + 60X_k^2 + 130X_k^4 + 60X_k^6 + 3X_k^8\right]^{-1}. \tag{40}$$

Iteration (40) is rational. We investigate the stability of (40) for finding S in a neighborhood of the solution. In fact, we analyze how a small perturbation at the kth iterate is amplified or damped along the iterates.

Lemma 5. *Under the same conditions as in Theorem 3, the sequence $\{X_k\}_{k=0}^{k=\infty}$ generated by (40) is asymptotically stable.*

Proof. If X_0 is a function of A, then the iterates from (40) are all functions of A and hence commute with A. Let Δ_k be a numerical perturbation introduced at the kth iterate of (40). Next, one has

$$\widetilde{X}_k = X_k + \Delta_k. \tag{41}$$

Here, we perform a first-order error analysis, that is, formally using approximations $(\Delta_k)^i \approx 0$, since $(\Delta_k)^i$, $i \geq 2$, is close to zero (matrix). This formal approximation is meaningful if Δ_k is sufficiently small. We have

$$\begin{aligned}
\widetilde{X}_{k+1} &= \left(20\widetilde{X}_k + 108\widetilde{X}_k^3 + 108\widetilde{X}_k^5 + 20\widetilde{X}_k^7\right)\left[3I + 60\widetilde{X}_k^2 + 130\widetilde{X}_k^4 + 60\widetilde{X}_k^6 + 3\widetilde{X}_k^8\right]^{-1} \\
&= \left(20\left(X_k + \Delta_k\right) + 108\left(X_k + \Delta_k\right)^3 + 108\left(X_k + \Delta_k\right)^5 + 20\left(X_k + \Delta_k\right)^7\right) \times \left[3I + 60\left(X_k + \Delta_k\right)^2 + 130\left(X_k + \Delta_k\right)^4 + 60\left(X_k + \Delta_k\right)^6 + 3\left(X_k + \Delta_k\right)^8\right]^{-1} \\
&\simeq \left(256S + 640\Delta_k + 384S\Delta_k S\right)\left(256I + 512S\Delta_k + 512\Delta_k S\right)^{-1} \\
&\simeq \left(S + \frac{5}{2}\Delta_k + \frac{3}{2}S\Delta_k S\right)\left(I - 2S\Delta_k - 2\Delta_k S\right) \\
&\simeq \left(S + \frac{1}{2}\Delta_k - \frac{1}{2}S\Delta_k S\right),
\end{aligned} \tag{42}$$

wherein the identities

$$(B + C)^{-1} \simeq B^{-1} - B^{-1}CB^{-1}, \tag{43}$$

were used (for any nonsingular matrix B and the matrix C). Note that, after some algebraic manipulations and the approximation $\Delta_{k+1} = \widetilde{X}_{k+1} - X_{k+1} = \widetilde{X}_{k+1} - S$, one can verify (assuming $X_k \simeq \operatorname{sign}(A) = S$ for enough large k)

$$\Delta_{k+1} \simeq \frac{1}{2}\Delta_k - \frac{1}{2}S\Delta_k S. \tag{44}$$

We also used the equalities $S^2 = I$ and $S^{-1} = S$, relative to the matrix sign function. We can now conclude that the perturbation at the iterate $k + 1$ is bounded; that is,

$$\left\|\Delta_{k+1}\right\| \simeq \frac{1}{2^{k+1}}\left\|\Delta_0 - S\Delta_0 S\right\|. \tag{45}$$

Therefore, the sequence $\{X_k\}_{k=0}^{k=\infty}$ generated by (40) is asymptotically stable. This ends the proof. □

It is now not difficult to also show that (40) has a sixth order of convergence and reads in the following error inequality:

$$\begin{aligned}
&\left\|X_{k+1} - S\right\| \\
&\quad \leq \left(\left\|M_k^{-1}\right\|\left\|S^{-1}\right\|\left\|3I + SX_k\left(-2I + 3SX_k\right)\right\|\right) \\
&\quad \cdot \left\|X_k - S\right\|^6,
\end{aligned} \tag{46}$$

where $M_k = 3I + 60X_k^2 + 130X_k^4 + 60X_k^6 + 3X_k^8$.

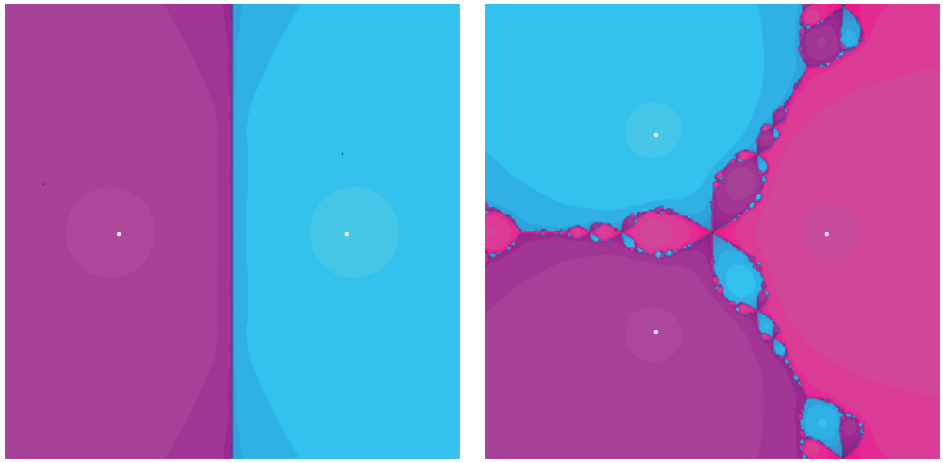
Here we make a comment that we have not given any discussions on backward stability for the polar decomposition in the sense of measuring $\|A - UH\|_F / \|A\|_F$ (this is a much stronger notion than the stability).

An acceleration via scaling, similar to that of (34), is applicable to (40). See [18] for an interesting choice of the scaling parameter. In Figure 5, we have drawn the basins of attractions for (16) in order to solve the complex polynomial of different orders. Although Figure 5(a) shows a global convergence of our method for the matrix sign function (also theoretically shown at the end of Section 2), the application of the proposed method in computation of the matrix sector function needs a deeper care.

A similar implementation for (40) is advised and could be found in Algorithm 2 (we denote it by ALS, i.e., Algorithm for Sign).

6. Numerical Experiments

We have tested the contributed method (22), denoted by PMP, and (40), denoted by PMS, using the programming package



(a) $x^2 - 1 = 0$ (b) $x^3 - 1 = 0$

Figure 5: Basins of attraction (16) shaded according to the number of iterations.

Table 1: Results of comparisons in Example 1 with $U_0 = A$ for polar decomposition.

	Methods			
	NMP	HMP	PMP	ALP
IT	9	6	4	3 + 1
R_{k+1}	2.93539×10^{-9}	7.6991×10^{-9}	7.63809×10^{-10}	7.63809×10^{-10}
Time	2.7812500	1.1250000	1.4218750	1.2750000
F_{k+1}	3.60456×10^{-14}	1.05716×10^{-14}	8.2024×10^{-15}	3.52843×10^{-14}

Mathematica 8 with the machine precision $\epsilon = 2.22045 \times 10^{-16}$. Apart from this scheme, several iterative methods such as (6), denoted by NMP, and (7), denoted by HMP, which require one (pseudo)inverse per cycle, have been tested. We also use the algorithms ALP, ALS, NMS, and HMS. Note that a sixth-order method from the Padé family [13] for the matrix sign is defined as follows:

$$X_{k+1} = X_k \left[6I + 20X_k^2 + 6X_k^4\right] \cdot \left[I + 15X_k^2 + 15X_k^4 + X_k^6\right]^{-1}. \quad (47)$$

We denote this iteration by SMS (sixth-order method for sign), that is, [2/3]-Padé approximant of the Padé family. Although the computational complexity of this method is lower than that of PMS, the PMS produces results of higher accuracies in high precision computing as will be observed in Example 2.

The considered stopping termination in performed numerical experiments is

$$R_{k+1} = \frac{\|U_{k+1} - U_k\|_\infty}{\|U_k\|_\infty} \leq \epsilon, \quad (48)$$

wherein ϵ is the tolerance. There is a similar stopping termination once we used it to find the matrix sign function.

Example 1. In this experiment, we study the behavior of different methods for finding the unitary polar factor of the complex rectangular 400 × 200 matrix which is randomly generated with the uniform distribution using the code

```
SeedRandom[1234]; m = 400; n = 200;
A = RandomComplex[{-1 - I, 1 + I},
{m, n}];
```

The results of the comparison are arranged together in Table 1 applying the tolerance $\epsilon = 10^{-6}$. It could easily be observed that a clear reduction in the number of iterations and the CPU time for finding the polar decomposition is obtained using ALP.

A considerable increase of the accuracy of approximations produced by the proposed method could be observed from numerical results. To check the accuracy of the numerical results, we have computed and reported $F_k = \|U_k^* U_k - I\|_F$ for the last iteration in Table 1.

An important application of higher order methods is in high precision computing. In the next test we considered an academical example, with 128-digit floating point. We do this consideration purposely so as to check the convergence behavior of different methods by the following definition for computational order of convergence (COC) using (48). The COC can be approximated by

$$\text{COC} = \frac{\ln\left(R_{k+1}/R_k\right)}{\ln\left(R_k/R_{k-1}\right)}, \quad (49)$$

Table 2: Results of comparisons in Example 2 for the matrix sign.

	Methods			
	NMS	HMS	SMS	PMS
IT	13	9	6	6
R_{k+1}	2.13346×10^{-36}	3.9478×10^{-58}	3.83821×10^{-69}	2.63194×10^{-95}
COC	2.0	3.0	6.03717	6.0543
E_{k+1}	1.65001×10^{-72}	0	0	0

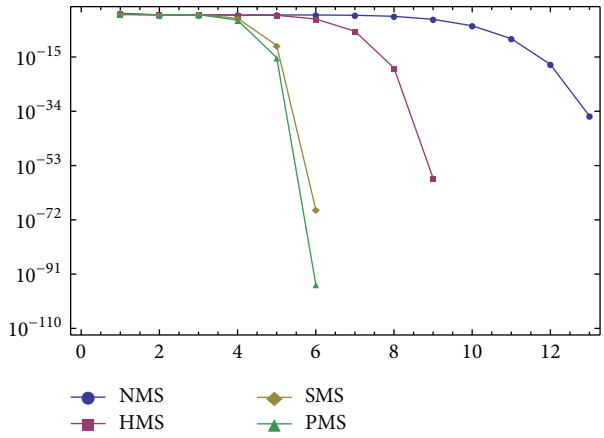


Figure 6: Convergence history of different methods using `logplot` in Example 2 with high precision computing.

at which the last three approximations for the polar decomposition or the matrix sign function are used.

Example 2 (academical test). In this experiment, we study the behavior of different methods for finding the matrix sign function of the following 5×5 matrix:

$$A = \begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix}. \quad (50)$$

The results of comparison are carried out in Figure 6 and Table 2 applying the tolerance $\epsilon = 10^{-20}$. It could easily be observed that there is a clear reduction in the number of iterations ensured by the application of PMS. To check the accuracy of the results we have computed and reported $E_k = \|X_k^2 - I\|_\infty / \|X_k\|_\infty^2$.

Example 3. In this experiment, we compare the behavior of different accelerated methods via scaling defined by (33) for finding the unitary polar factors. We now compare the Newton method (NMP), accelerated Newton method (ANMP), PMP, and the accelerated proposed method for polar decomposition (34) denoted by APMP. We used the following six

Table 3: Results of comparisons for Example 3 in terms of number of iterations.

Matrices	Methods			
	NMP	PMP	ANMP	APMP
#1	11	5	9	4
#2	11	5	9	4
#3	11	5	9	4
#4	11	5	9	4
#5	11	5	9	4
#6	12	5	9	4

Table 4: Results of comparisons for Example 3 in terms of elapsed time (s).

Matrices	Methods			
	NMP	PMP	ANMP	APMP
#1	5.1718750	3.0781250	4.6875000	4.2031250
#2	5.2031250	3.1093750	4.6718750	4.2031250
#3	5.1718750	3.0781250	4.6718750	4.1875000
#4	5.1718750	3.0781250	4.7031250	4.1718750
#5	5.1562500	3.0781250	4.6718750	4.2031250
#6	5.5937500	3.0781250	4.6875000	4.1718750

complex rectangular 310×300 matrices (with uniform distribution):

```
m = 310; n = 300; number = 6; SeedRandom
[345];
Table[A[l] = RandomComplex [{−10−10 I,
10+10 I}, {m, n}];, {l, number}];
```

The results of comparison are carried out in Tables 3 and 4 applying the tolerance $\epsilon = 10^{-10}$ in double precision. It could easily be observed that there is a clear reduction in the number of iterations, ensured by the acceleration via scaling (33).

It can be observed that taking into account every presented method (e.g., NMP) the number of iterations for all of matrices is the same for that method. It would be more interesting to make a comment that there is an upper bound for the maximal number of iterations for each method in computing the matrix sign and the polar decomposition in double precision arithmetic, as discussed fully in [3].

Example 4. And finally to report the number of iterations when the input matrix is ill-conditioned, we take into consideration the Hilbert matrix of dimension 10 whose condition number is 3.53534×10^{13} (in l_∞) [18]. The unitary polar factor is known to be the unit matrix I in this case. Applying the tolerance 10^{-10}, we found that NMP, HMP, and PMP require 49, 31, and 19 cycles, respectively. This again shows the superiority of the proposed approach.

7. Concluding Remarks

The polar decomposition is an important theoretical and computational tool, known because of its approximative properties, its sensitivity to perturbations, and its computation.

In this paper, we have developed a high order method for solving nonlinear scalar equations and then extend it to the iterative method applicable in the computation of the matrix polar decomposition. It has been shown that the convergence of the method is global and its convergence rate is six.

It has further been discussed how the proposed method could be applied in finding the matrix sign function. The presented scheme possesses asymptotic stability and it is very useful in a high precision computing environment. Some numerical tests have been provided to show the performance of the new methods.

Conflict of Interests

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

Acknowledgments

The second author gratefully acknowledges support from the Research Project 174013 of the Serbian Ministry of Science. The second and third authors gratefully acknowledge support from the Project "Applying Direct Methods for Digital Image Restoring" of the Goce Delčev University.

References

[1] W. Gander, "On Halley's iteration method," *The American Mathematical Monthly*, vol. 92, no. 2, pp. 131–134, 1985.

[2] F. K. Haghani and F. Soleymani, "On a fourth-order matrix method for computing polar decomposition," *Computational & Applied Mathematics*, vol. 34, no. 1, pp. 389–399, 2015.

[3] N. J. Higham, *Functions of Matrices: Theory and Computation*, SIAM, Philadelphia, Pa, USA, 2008.

[4] J. D. Roberts, "Linear model reduction and solution of the algebraic Riccati equation by use of the sign function," *International Journal of Control*, vol. 32, no. 4, pp. 677–687, 1980.

[5] N. J. Higham, "Computing the polar decomposition-with applications," *SIAM Journal on Scientific and Statistical Computing*, vol. 7, pp. 1160–1174, 1986.

[6] Y. Nakatsukasa, Z. Bai, and F. Gygi, "Optimizing Halley's iteration for computing the matrix polar decomposition," *SIAM Journal on Matrix Analysis and Applications*, vol. 31, no. 5, pp. 2700–2720, 2010.

[7] K. Du, "The iterative methods for computing the polar decomposition of rank-deficient matrix," *Applied Mathematics and Computation*, vol. 162, no. 1, pp. 95–102, 2005.

[8] W. Gander, "Algorithms for the polar decomposition," *SIAM Journal on Scientific and Statistical Computing*, vol. 11, no. 6, pp. 1102–1115, 1990.

[9] J. M. Gutiérrez and M. A. Hernández, "A family of Chebyshev-Halley type methods in Banach spaces," *Bulletin of the Australian Mathematical Society*, vol. 55, no. 1, pp. 113–130, 1997.

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

[11] F. Soleymani, S. Shateyi, and H. Salmani, "Computing simple roots by an optimal sixteenth-order class," *Journal of Applied Mathematics*, vol. 2012, Article ID 958020, 13 pages, 2012.

[12] J. Hoste, *Mathematica Demystified*, McGraw-Hill, New York, NY, USA, 2009.

[13] 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.

[14] A. A. Dubrulle, *Frobenius Iteration for the Matrix Polar Decomposition*, Hewlett-Packard Company, Palo Alto, Calif, USA, 1994.

[15] N. J. Higham, "The matrix sign decomposition and its relation to the polar decomposition," *Linear Algebra and Its Applications*, vol. 212-213, pp. 3–20, 1994.

[16] A. R. Soheili, F. Toutounian, and F. Soleymani, "A fast convergent numerical method for matrix sign function with application in SDEs," *Journal of Computational and Applied Mathematics*, vol. 282, pp. 167–178, 2015.

[17] F. Soleymani, P. S. Stanimirović, S. Shateyi, and F. K. Haghani, "Approximating the matrix sign function using a novel iterative method," *Abstract and Applied Analysis*, vol. 2014, Article ID 105301, 9 pages, 2014.

[18] R. Byers and H. Xu, "A new scaling for Newton's iteration for the polar decomposition and its backward stability," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 2, pp. 822–843, 2008.