Applied Mathematics and Computation 260 (2015) 242–248

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Construction of a convergent scheme for finding matrix sign function

Eman S. Alaidarous, Malik Zaka Ullah*

Department of Mathematics, Faculty of Sciences, King Abdulaziz University, P.O. Box 80203, Jeddah 21589, Saudi Arabia

ARTICLE INFO

MSC:
65F60

Keywords:
Matrix sign function
Newton–Schulz iteration
Convergence
Iterative methods

ABSTRACT

In this paper, a new method is constructed for finding matrix sign function. It is proven that it possesses the high convergence order nine with global behavior. Numerical experiments are also provided to support the theoretical discussions.

1. Introductory notes

Assume that $A \in \mathbb{C}^{n\times n}$ is a matrix with the spectrum $\sigma(A) = \{\lambda_1, \cdots, \lambda_l\}$. Moreover, let A have the following Jordan normal form

$$J = T^{-1}AT = \text{diag}\,(J_{m_1}(\lambda_1), \cdots, J_{m_l}(\lambda_l)), \tag{1}$$

and the function $f : \mathbb{C} \to \mathbb{C}$ is $m_j - 1$ times differentiable at λ_j for $j = 1, \cdots, l$. Then the matrix function $f(A)$ is defined as

$$f(A) = Tf(J)T^{-1}, \tag{2}$$

where

$$f(J) = \text{diag}\,(f(J_{m_1}(\lambda_1)), \cdots, f(J_{m_l}(\lambda_l))), \tag{3}$$

with

$$f(J_{m_j}(\lambda_j))) = \sum_{\upsilon=0}^{m_j-1} \frac{1}{\upsilon!} f^{(\upsilon)}(\lambda_j) \cdot S_{m_j}^{\upsilon}. \tag{4}$$

Note that m_j is the size of the jth Jordan block associated with λ_j, i.e.,

$$J_{m_j}(\lambda_j) = \begin{pmatrix} \lambda_j & 1 & 0 & \cdots & 0 \\ 0 & \lambda_j & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & 1 \\ 0 & \cdots & \cdots & 0 & \lambda_j \end{pmatrix} =: \lambda_j I + S_{m_j} \in \mathbb{C}^{m_j \times m_j}. \tag{5}$$

* Corresponding author. Tel.: +966 596369791.
E-mail addresses: ealaidarous@kau.edu.sa (E.S. Alaidarous), mzhussain@kau.edu.sa (M.Z. Ullah).

http://dx.doi.org/10.1016/j.amc.2015.03.049
0096-3003/

This definition makes explicit use of the Jordan canonical form and of the associated transformation matrix T. Neither T nor J are unique, but it can be shown that $f(A)$ as introduced in (2) does not depend on the particular choice of T or J [1].

In this paper, we are interested in numerically computing the matrix sign function which is in fact one of the most fundamental tools in the theory of matrix functions

$$S = \mathrm{sign}(A), \tag{6}$$

for a matrix $A \in \mathbb{C}^{n\times n}$ with no eigenvalues lying on the imaginary axis.

We remind that a primary matrix function with a non-primary flavor is the matrix sign function, which for a matrix A is a (generally) non-primary square root of I that depends on A.

A simplified definition of the matrix sign function for Hermitian case (eigenvalues are all real) is

$$\mathrm{sign}(A) = U\mathrm{diag}\,(\mathrm{sign}(\lambda_1), \cdots, \mathrm{sign}(\lambda_n))U^*, \tag{7}$$

where $U^*AU = \mathrm{diag}\,(\lambda_1, \ldots, \lambda_n)$ is a diagonalization of A.

The matrix sign function was first introduced by Roberts [5] for solving algebraic Riccati equations. However, it was soon extended to solving the spectral decomposition problem [2].

Several numerical methods exist for computing (6). Here, we discuss Newton-type methods which are fixed-point-type methods by producing a convergent sequence of matrices via applying a suitable initial matrix.

Clearly, the most fundamental matrix iteration for finding (6) is the method of Newton as follows

$$X_{k+1} = \frac{1}{2}\left(X_k + X_k^{-1}\right). \tag{8}$$

It should be remarked that iterative methods, such as (8), and the Newton–Schultz iteration

$$X_{k+1} = \frac{1}{2}X_k\left(3I - X_k^2\right), \tag{9}$$

or the cubically convergent Halley's method

$$X_{k+1} = [I + 3X_k^2][X_k(3I + X_k^2)]^{-1}, \tag{10}$$

are all special cases of the Padé family proposed originally in [4].

The Newton–Schulz iteration is quadratically convergent (just like (8)), for computing the sign function of a matrix. It is advantageous over other methods for high-performance computing because it is rich in matrix–matrix multiplications. However, its convergence is not guaranteed for all cases of the input matrix A, see for more [3].

The remaining sections of this work are organized in what follows. In Section 2, we discuss how to construct new iterative methods for finding (6). It is also shown that the constructed method of this work is convergent with high order. Numerical experiments in an academical manner are also given to show the higher numerical accuracy for the constructed solver in Section 3. The paper ends in Section 4 with some concluding comments.

2. New method and error analysis

As discussed in [6], new iterative methods for S can be constructed using suitable nonlinear equation solvers. Here, we present the following root-solver without memory which consists of three steps

$$\begin{cases} y_k = x_k - 2^{-1}f'(x_k)^{-1}f(x_k), \\ z_k = x_k - f'(y_k)^{-1}f(x_k), \\ x_{k+1} = z_k - \left(1 + \dfrac{1}{2}\left(\dfrac{L_k}{1 + \dfrac{49}{6}L_k}\right)\right)\dfrac{f(z_k)}{f'(z_k)}, \end{cases} \tag{11}$$

with $L_k = \frac{f''(z_k)f(z_k)}{f'(z_k)^2}$. In what follows, we observe that (11) possesses ninth order of convergence.

Theorem 2.1. *Let $\alpha \in D$ be a simple zero of a sufficiently differentiable function $f : D \subseteq \mathbb{C} \to \mathbb{C}$ for an open interval D, which contains x_0 as an initial approximation of α. Then the iterative expression (11) reads the error equation below*

$$e_{k+1} = \frac{1}{192}\left(55c_2^2 - 3c_3\right)\left(4c_2^2 - c_3\right)^3 e_k^9 + O(e_k^{10}), \tag{12}$$

wherein $c_j = \frac{f^{(j)}(\alpha)}{j!f'(\alpha)}$, $e_k = x_k - \alpha$.

Proof. The proof of this theorem is based on Taylor's series expansion of the iterative method (11) around the solution in the kth iterate. To save the space and not to distract from the main topic, we exclude the proof. □

Applying (11) on the following matrix equation

$$F(X) \equiv X^2 - I = 0, \tag{13}$$