Advances in Applied Mathematics 应用数学进展, 2026, 15(4), 483-492
Published Online April 2026 in Hans. https://www.hanspub.org/journal/aam
https://doi.org/10.12677/aam.2026.154176

Hans汉斯

正则M-矩阵平方根的一类不动点迭代法

马琳璐[1]，关晋瑞[1*]，邵荣侠[2]

[1]太原师范学院数学与统计学院，山西 晋中
[2]新疆财经大学统计与数据科学学院，新疆 乌鲁木齐

收稿日期：2026年3月9日；录用日期：2026年4月3日；发布日期：2026年4月13日

摘 要

矩阵平方根在数学的很多领域中都有广泛的应用。为了探讨M-矩阵平方根的数值算法，提出了一类不动点迭代法计算正则M-矩阵的平方根，并对其收敛性进行了分析。该方法的可行性已通过数值实验进行了验证，且在一定条件下优于现有的若干算法。

关键词

正则M-矩阵，平方根，不动点迭代法，收敛性分析，收敛率

A Class of Fixed-Point Iteration Method for the Square Root of Regular M-Matrices

Linlu Ma[1], Jinrui Guan[1*], Rongxia Shao[2]

[1]School of Mathematics and Statistics, Taiyuan Normal University, Jinzhong Shanxi
[2]School of Statistics and Data Science, Xinjiang University of Finance and Economics, Urumqi Xinjiang

Received: March 9, 2026; accepted: April 3, 2026; published: April 13, 2026

Abstract

The square root of matrix has a wide range of applications in many fields of mathematics. In order to explore the numerical algorithms for the square root of an M-matrix, a class of fixed-point iteration method for computing the square root of a regular M-matrix is proposed, and the convergence of these methods is analyzed. The feasibility of this method has been theoretically analyzed and verified through numerical experiments, and under certain conditions, it outperforms several existing algorithms.

*通讯作者。

文章引用: 马琳璐, 关晋瑞, 邵荣侠. 正则 M-矩阵平方根的一类不动点迭代法[J]. 应用数学进展, 2026, 15(4): 483-492.
DOI: 10.12677/aam.2026.154176

Keywords

Regular M-Matrix, Square Root, Fixed-Point Iterative Method, Convergence Analysis, Convergence Rate

1. 引言

设矩阵 $A \in R^{n\times n}$，若存在 $X \in R^{n\times n}$，满足方程

$$X^2 = A, \tag{1.1}$$

则称 X 为矩阵 A 的平方根。矩阵平方根在广义特征值问题、微分方程边值问题、非线性矩阵方程的求解、矩阵对数的计算等方面都有应用[1]-[4]。

对于矩阵平方根的相关内容，很多学者进行了讨论。在理论上，学者们除了对一般矩阵平方根展开了深入研究，对于特殊的矩阵，如 M-矩阵[5]、对角元为正的 H-矩阵[6]、中心对称矩阵[7] [8]、循环矩阵[9]等，其平方根也有研究。在数值算法中，经典的求解矩阵平方根的方法有 Schur 方法[10] [11]，它基于矩阵的 Schur 分解，而 Schur 分解最初是用来研究矩阵特征值和特征向量的。然而，矩阵规模在实际应用中不断扩大，Schur 方法等逐渐出现了弊端。在应用场景中，大规模矩阵计算时间长，难以满足实时性等要求。为解决这些方法的不足，迭代算法应运而生[12]-[16]。这类算法通过多次迭代逐步逼近矩阵的平方根，简化了计算过程，提高了计算效率。人们应用牛顿法计算矩阵的平方根[17]，牛顿法具备了二次收敛的数学性质，然而牛顿法的计算量太大且稳定性较差。文献[18]提出了新的平方根公式，数值稳定性优于传统公式，设计了结构化牛顿迭代，大幅降低了计算成本，但对比方法不够全面。近年来，人们又提出了一些方法，如 DB 算法、CR 算法、梯度网络神经法、Zolotarev 法等[19]-[22]。文献[23]提出一类不精确迭代解法，还衍生出有限步不精确的不动点迭代法，计算时间大幅减少，相对误差仍满足精度要求。文献[24]利用了二次矩阵方程理论，提出了一种不动点迭代法，该算法具有保结构、适定的特点。然而，这类算法也存在着一些问题，比如当矩阵条件数较大时，计算的稳定性会显著下降。尤其是矩阵接近奇异时，计算误差可能会急剧增大，在实际应用中可靠性不足。因此需要人们继续研究新的算法。

M-矩阵是矩阵理论和数值分析中的一类重要矩阵。文献[5]讨论了 M-矩阵平方根的存在性、唯一性和计算方法，这些结果为相关领域的应用提供了新的方法。文献[2] [5]提出了一种称为二项式迭代的简单迭代方法以计算 M-矩阵 $A = s(I - C)$ 的平方根，如下所示

$$P_{k+1} = \frac{1}{2}\left(C + P_k^2\right), P_0 = 0 \tag{1.2}$$

文献[2]中的收敛性分析显示，二项式迭代呈现出单调收敛的特性，能够保持结构不变，并且在大多数情况下展现出线性收敛速率。但是该方法存在一定的局限性，有时候收敛速度较慢，特别是矩阵规模较大时。

为此，本文对于如何计算正则 M-矩阵的平方根，提出了一类不动点迭代法。该方法通过构造迭代格式，利用 M-矩阵的结构特性，旨在实现更快的收敛速度和更高的计算精度。后面的内容将介绍该迭代法的理论基础、收敛性分析、数值实验，为正则 M-矩阵平方根的计算提供更有效的解决方案，推动相关领

域的进一步发展。

2. 预备知识

下面将阐述本研究采用的数学符号、Z-矩阵和M-矩阵定义和性质。

用 $R^{m\times n}$ 表示所有 $m\times n$ 维实数矩阵的集合，符号 $A\otimes B$ 表示两个矩阵 A 和 B 的 Kronecher 积，向量 $vec(A)$ 表示矩阵的 A 列向量形式，矩阵 A 的谱半径表示为 $\rho(A)$，I 表示 n 阶单位矩阵。

设 $A=(a_{ij})\in R^{n\times n}$，若对于所有的 $i\neq j$，都有 $a_{ij}\leq 0$，则称 A 为Z-矩阵。对于Z-矩阵 A，若存在一个非负矩阵 B，满足 $A=sI-B$ 且 $s\geq\rho(B)$，则称 A 为 M-矩阵，其中 $\rho(B)$ 表示矩阵 B 的谱半径。特别地，若 $s>\rho(B)$，则称 A 为非奇异M-矩阵；若 $s=\rho(B)$ 则称 A 为奇异M-矩阵。

引理 2.1 [25] 设 $A\in R^{n\times n}$ 为Z-矩阵，则下面几个结论是等价的：

(1) A 是非奇异M-矩阵；

(2) $A^{-1}\geq 0$；

(3) 存在正向量 $v>0$，使得 $v>0$；

(4) A 的特征值都具有正实部。

定义 2.1 [26]设 $A=(a_{ij})\in R^{n\times n}$ 为M-矩阵，如果存在一个正向量 $u>0$，使得 $Au\geq 0$，则称矩阵 A 为正则M-矩阵。

很容易证明，所有非奇异 M-矩阵和不可约奇异 M-矩阵都是正则 M-矩阵。此外，对于任意 Z-矩阵 A，如果存在 $Au\geq 0$ 对于某个正向量 $u>0$ 成立，则 A 也是正则M-矩阵。

关于正则M-矩阵的平方根，文献[27]中得到如下结论。

引理 2.2 [27]设 $A=(a_{ij})\in R^{n\times n}$ 为正则M-矩阵，则 A 存在平方根，而且该平方根也是一个正则M-矩阵。

3. 一类不动点迭代法

在本部分，主要提出了一类不动点迭代法，并对其收敛性进行了分析。

设矩阵 $A\in R^{n\times n}$ 是正则M-矩阵，令 $X=D-Y$，其中

$$D=\begin{pmatrix} d_1 & & \\ & \ddots & \\ & & d_n \end{pmatrix},\quad d_i>0,$$

代入(1.1)式可得如下方程

$$Y^2-DY-YD+D^2-A=0, \tag{3.1}$$

这里我们选取 $d_i\geq\sqrt{a_{ii}}$，此时 D^2-A 是非负矩阵。

下面研究方程(3.1)最小非负解的存在性。设 $\alpha>0$ 是一个给定的参数，由(3.1)式可得如下方程

$$Y^2+Y(\alpha I-D)+D^2-A=(\alpha I+D)Y,$$

选取参数使得 $\alpha\geq\max D_{ii}$，此时 $\alpha I+D$ 可逆且 D^2-A 是非负矩阵，因此可将上式整理为如下不动点格式

$$Y_{k+1}=(\alpha I+D)^{-1}\left[Y_k^2+Y_k(\alpha I-D)+D^2-A\right],\quad Y_0=0。 \tag{3.2}$$

我们有下面的结论。

引理 3.1： 设 $A=(a_{ij})\in R^{n\times n}$ 为正则M-矩阵，则由迭代式(3.2)得到的序列 $\{Y_k\}$ 是适定的，且单调递增有上界。

证明：显然序列$\{Y_k\}$适定。另外，由数学归纳法我们能够证明

(i) 序列$\{Y_k\}$单调递增，即$0 \le Y_k \le Y_{k+1}$，$k \ge 0$。

当$k=0$时，$Y_0 = 0 \le Y_1 = (\alpha I + D)^{-1}(D^2 - A)$，结论成立。

假设对任意的$k \ge 1$，$Y_k \ge Y_{k-1}$成立，则

$$Y_{k+1} - Y_k = (\alpha I + D)^{-1}\left[\left(Y_k^2 - Y_{k-1}^2\right) + \left(Y_k - Y_{k-1}\right)(\alpha I - D)\right],$$

因为$Y_k \ge Y_{k-1}$及$\alpha I - D \ge 0$，得$Y_{k+1} - Y_k \ge 0$。故$\{Y_k\}$单调递增。

(ii) 序列$\{Y_k\}$存在上界。因为A是正则 M-矩阵，那么一定存在一个向量$u > 0$，且$Au \ge 0$。在下文中，对任意的$k \ge 0$，$Y_k u \le Du$。当$k=0$时，结论显然成立。假设$Y_{k-1}u \le Du$成立，则

$$\begin{aligned} Y_k u &\le (\alpha I + D)^{-1}\left[D^2 u + D(\alpha I - D)u + D^2 u\right] \\ &= (\alpha I + D)^{-1}\left(\alpha D + D^2\right)u \\ &= Du \end{aligned}$$

所以，对任意$k \ge 0$，$Y_k u \le Du$。

定理 3.1：设$A = (a_{ij}) \in R^{n\times n}$为正则 M-矩阵，则方程(3.1)存在最小非负解$Y$，且$X = D - Y$也是正则 M-矩阵。此外，$X = D - Y$是$A$的平方根。

证明：由引理 3.1，迭代式(3.2)生成的序列$\{Y_k\}$适定，且单调递增有上界，因此存在极限$\lim_{k\to\infty} Y_k = Y$。

在迭代式(3.2)两边取极限，可知Y是方程(3.1)的解，且$Y \ge 0$。若存在另一个非负解$Z \ge 0$，通过上述引理可知$Y_k \le Z$对所有k成立，所以$Y \le Z$。因此Y是(3.1)式的最小非负解。

由上述引理知$Y_k u \le Du$，取极限可得$Yu \le Du$，所以

$$(D - Y)u = Du - Yu \ge 0,$$

因此$D - Y$是正则 M-矩阵。又$(D - Y)^2 = A$，即$D - Y$是A的平方根。证毕。

通过上面的分析，当方程(1.1)中的矩阵$A = (a_{ij}) \in R^{n\times n}$是一个正则 M-矩阵时，矩阵$A$存在 M-矩阵平方根，且该平方根也是一个正则 M-矩阵。为了有效地求解方程(3.1)，根据方程的特殊形式，我们将(3.1)进行改写，可得到如下的迭代法

$$DY_{k+1} + Y_{k+1}D = D^2 - A + Y_k^2,\quad Y_0 = 0。\tag{3.3}$$

注：由于方程(3.1)存在最小非负解，根据解的唯一性，我们能发现D并不影响最终结果。结合D的对角元D_{ii}，要求$\alpha \ge \max D_{ii}$，一方面能保证$\alpha I + D$可逆，避免分母为零的奇异问题，另一方面结合$D^2 - A$非负，为迭代式(3.2)的构造提供了理论支撑。而参数α本质是理论推导的辅助工具，用于证明格式的合理性。同时，α的最优取值已隐含在$\max D_{ii}$中，舍弃后不影响算法的收敛性和结果准确性，实际迭代简洁高效。

完整的算法格式如下：

算法 3.1：一类新的不动点迭代法(NFP)

步骤 1. 确定对角矩阵D，满足$d_i \ge \sqrt{a_{ii}}$；

步骤 2. 初始化$Y_0 = 0 \in R^{n\times n}$；

步骤 3. 对于$k = 0,1,2,\cdots$，直至序列$\{Y_k\}$收敛，按照下面的迭代格式进行计算

$$DY_{k+1} + Y_{k+1}D = D^2 - A + Y_k^2;$$

步骤 4. 令$X = D - Y$，其中$\lim_{k\to\infty} Y_k = Y$；

该迭代法的运算量大约为$2n^3$。

引理 3.2：设 $A\in R^{n\times n}$ 是正则 M-矩阵，对任意 $k\ge 0$，由迭代法(3.3)生成的序列 $\{Y_k\}$ 满足下列不等式

$$0\le Y_k\le Y_{k+1}\le Y\text{。}\tag{3.4}$$

证明：下面，采用数学归纳法证明。

当 $k=0$ 时，由(3.3)式，可得

$$DY_1+Y_1D=D^2-A\text{，}\tag{3.5}$$

因为 D^2-A 是非负矩阵，显然 $0=Y_0\le Y_1$，又由于

$$Y^2-DY-YD+D^2-A=0\text{。}\tag{3.6}$$

整理(3.5)式与(3.6)式，可知

$$D\left(Y_1-Y\right)+\left(Y_1-Y\right)D=-Y^2\text{，}$$

上式 $-Y^2\le 0$，因此 $Y_1\le Y$。于是当 $k=0$ 时，结论成立。

假设当 $k=l-1$ 时结论成立，则 $0\le Y_{l-1}\le Y_l\le Y$。

(i) 若 $Y_l\le Y$，由迭代法(3.3)可知

$$DY_{l+1}+Y_{l+1}D=D^2-A+Y_l^2\text{，}\tag{3.7}$$

将(3.6)式代入(3.7)式，可得

$$D\left(Y-Y_{l+1}\right)+\left(Y-Y_{l+1}\right)D=Y^2-Y_l^2\text{，}$$

又因为 $Y^2-Y_l^2\ge 0$，所以 $Y_{l+1}\le Y$。

(ii) 若 $Y_{l-1}\le Y_l$，将(3.7)式代入下列等式

$$DY_l+Y_lD=D^2-A+Y_{l-1}^2\text{，}$$

可得到

$$D\left(Y_{l+1}-Y_l\right)+\left(Y_{l+1}-Y_l\right)D=Y_l^2-Y_{l-1}^2\text{。}$$

又因为 $Y_l^2-Y_{l-1}^2\ge 0$，所以 $Y_l\le Y_{l+1}$。因此，当 $k=l$ 时结论成立。

所以，(3.4)式对任意 $k\in N^*$ 都成立。

定理 3.2：设 $A\in R^{n\times n}$ 是正则 M-矩阵，则由迭代法(3.3)生成的序列是 $\{Y_k\}$ 适定的，且单调递增收敛于 Y。

证明：显然序列 $\{Y_k\}$ 适定。由引理 3.2 可知，序列 $\{Y_k\}$ 单调递增且有上界，从而存在一个非负矩阵 Y^*，使得 $\lim\limits_{k\to\infty}Y_k=Y^*$。由引理 3.2，我们知道 $Y^*\le Y$。另一方面，取(3.3)式极限，Y^* 是(3.1)式的一个解，因此 $Y\le Y^*$，所以 $Y=Y^*$。

定理 3.3：设 $A\in R^{n\times n}$ 是正则 M-矩阵，则迭代法(3.3)生成的序列 $\{Y_k\}$ 的收敛率 R 满足

$$R=\rho\left[\left(I\otimes D+D\otimes I\right)^{-1}\left(I\otimes 2Y\right)\right]$$

其中 $\rho(\bullet)$ 是矩阵的谱半径。

证明：从引理 3.2，可知

$$D\left(Y-Y_k\right)+\left(Y-Y_k\right)D=Y^2-Y_{k-1}^2\le 2Y\left(Y-Y_{k-1}\right)\text{，}$$

对其向量化，可得

$$\left(I\otimes D+D\otimes I\right)vec\left(Y-Y_k\right)\le\left(I\otimes 2Y\right)vec\left(Y-Y_{k-1}\right)\text{，}$$

所以有如下不等式

$$\begin{aligned} vec(Y-Y_k) &\le (I\otimes D+D\otimes I)^{-1}(I\otimes 2Y)vec(Y-Y_{k-1}) \\ &\le \cdots \\ &\le \left[(I\otimes D+D\otimes I)^{-1}(I\otimes 2Y)\right]^k vec(Y-Y_0) \end{aligned},$$

对上述不等式的两边取 k 次方根并利用范数的性质进一步简化，可得

$$\left\|vec(Y-Y_k)\right\|^{1/k} \le \left\|\left[(I\otimes D+D\otimes I)^{-1}(I\otimes 2Y)\right]^k\right\|^{1/k}\left\|vec(Y-Y_0)\right\|^{1/k},$$

对上式两边取极限可得

$$\lim_{k\to\infty}\left\|vec(Y-Y_k)\right\|^{1/k} \le \lim_{k\to\infty}\left\|\left[(I\otimes D+D\otimes I)^{-1}(I\otimes 2Y)\right]^k\right\|^{1/k},$$

则 $R=\rho\left[(I\otimes D+D\otimes I)^{-1}(I\otimes 2Y)\right]$。

推论 3.1：设 $A\in R^{n\times n}$ 是正则 M-矩阵，若 $D_1\le D_2$，则对应的收敛率 R 满足 $R_1\le R_2$，且 D^2-A 是非负矩阵。

证明：因为 $D=diag(d_1,d_2,\cdots,d_n)$，则

$$I\otimes D+D\otimes I=diag\left(d_i+d_j\right)_{i,j=1}^n,$$

其逆为

$$(I\otimes D+D\otimes I)^{-1}=diag\left(\frac{1}{d_i+d_j}\right)_{i,j=1}^n。$$

而 $I\otimes 2Y$ 为

$$I\otimes 2Y=diag(2Y,2Y,\cdots,2Y)。$$

因此

$$(I\otimes D+D\otimes I)^{-1}(I\otimes 2Y)=diag\left(\frac{2Y}{d_i+d_j}\right)_{i,j=1}^n,$$

即

$$R=\max_{i,j}\rho\left(\frac{2Y}{d_i+d_j}\right)。$$

因为 $\frac{2Y}{d_i+d_j}$ 的谱半径严格减小。所以 $D_1\le D_2$ 时，$R_1\le R_2$。

根据推论 3.1，D 的选择对 R 有直接影响，在确保选 D 满足 $d_i\ge\sqrt{a_{ii}}$ 且 d_i 是正数的前提下，选择较小的 D 有助于提高收敛速度，即降低收敛率 R。因此，在实际应用中，如果 A 的对角元素存在零值，可通过给 d_i 加上一个很小的正数 ε，避免影响计算的可行性。

4. 数值实验

在本部分，我们通过一些数值实验来验证新方法的可行性和有效性。我们将新方法称作(NFP)，与文

献[24]中的方法(FP)、文献[2]中的二项式迭代法(BI)、文献[17]中的方法(NM)进行对比性分析。每次实验的数值结果涵盖了迭代次数(IT)，CPU 时间(秒)和残差(Res)，其中残差定义为 $\mathrm{RES}=\frac{\left\|X^2-A\right\|_\infty}{\left\|A\right\|_\infty}$。所有实验均在 MATLAB (R2023b)环境中运行，迭代过程在满足 $\mathrm{RES}<10^{-6}$ 的条件下终止。

例 4.1：本例考虑一个可约奇异的 M-矩阵

$$A=\begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{pmatrix},$$

A 的平方根为(保留小数点后四位)

$$A^{1/2}=\begin{pmatrix} 1.3660 & -0.3660 & -1.0000 \\ -0.3660 & 1.3660 & -1.0000 \\ 0 & 0 & 0 \end{pmatrix},$$

数值结果见表 1。

Table 1. Numerical results
表 1. 数值结果

方法	IT	CPU	RES
FP	1407	0.0034	9.9886e−07
BI	1406	0.0024	9.9886e−07
NM	10	0.0009	4.7684e−07
NFP	12	0.0008	9.1411e−07

例 4.2：本例考虑一个不可约奇异的 M-矩阵

$$A=\begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{pmatrix},$$

A 的平方根为(保留小数点后四位)

$$A^{1/2}=\begin{pmatrix} 0.7893 & -0.5767 & -0.2107 \\ -0.5767 & 1.1553 & -0.5767 \\ -0.2107 & -0.5767 & 0.7893 \end{pmatrix},$$

数值结果见表 2。

Table 2. Numerical results
表 2. 数值结果

方法	IT	CPU	RES
FP	1407	0.0129	9.9886e−07
BI	1406	0.0043	9.9886e−07
NM	10	0.0137	4.7684e−07
NFP	1197	0.0071	9.9952e−07

例 4.3： 本例考虑通过以下方式随机生成阶数为 n 的非奇异 M-矩阵

$$a = rand(n,n)\text{；}\quad A = diag(1:n) + diag(a * ones(n,1)) - a\text{；}$$

在这个例子中，$n = 500,1000,2000,3000$，结果汇总在表 3。

Table 3. Numerical results
表 3. 数值结果

n	方法	IT	CPU	RES
500	FP	21	0.8053	8.8339e−07
	BI	20	0.5981	7.3388e−07
	NM	4	3.1703	9.4232e−08
	NFP	11	0.4602	5.0892e−07
1000	FP	21	2.1244	7.9827e−07
	BI	20	2.0917	7.7423e−07
	NM	4	10.1250	9.8515e−08
	NFP	11	1.6406	5.2192e−07
2000	FP	21	13.2126	7.7021e−07
	BI	20	12.5086	7.7103e−07
	NM	4	62.4396	9.5061e−08
	NFP	11	9.9054	5.1945e−07
3000	FP	21	43.8889	8.0102e−07
	BI	20	38.9251	7.8451e−07
	NM	4	197.3620	9.8864e−08
	NFP	11	33.7662	5.2201e−07

例 4.4： 本例考虑不可约非奇异的 M-矩阵

$$A = \begin{pmatrix} 1 & & & -1 \\ -1 & 2 & & \\ & \ddots & \ddots & \\ & & -1 & n \end{pmatrix},$$

在这个例子中，$n = 500,1000,2000,3000$，结果汇总在表 4。

Table 4. Numerical results
表 4. 数值结果

n	方法	IT	CPU	RES
500	FP	193	3.3364	9.8065e−07
	BI	192	3.3265	9.8065e−07
	NM	7	2.7580	8.4662e−08
	NFP	3	0.0791	3.8452e−07

续表

1000	FP	193	3.2891	9.8065e−07
	BI	192	3.0727	9.8065e−07
	NM	7	2.8716	8.4662e−08
	NFP	3	0.0906	3.8452e−07
2000	FP	254	25.2435	9.9721e−07
	BI	253	25.6135	9.9721e−07
	NM	8	18.4265	3.6988e−10
	NFP	3	0.5578	1.9220e−07
3000	FP	298	86.4473	9.7858e−07
	BI	297	86.0821	9.7858e−07
	NM	8	52.1373	4.8240e−09
	NFP	3	1.5002	1.2812e−07

由表 1~4 可知，对于不同阶数的 n，四种方法均满足精度要求的结果，但新方法的迭代次数和运行时间明显少于文献[2]、文献[17]、文献[24]。通过上述 4 个例子，我们能够得出新方法的可行性和有效性，且在一定条件下优于现有的若干算法。

5. 小结

本文研究了 M-矩阵平方根的数值算法，提出了一类的不动点迭代法用来计算正则 M-矩阵的平方根。在构造迭代格式的基础上，证明了迭代序列的适定性、单调性和收敛性。数值实验结果验证了方法的正确性，并且在一定情况下该方法的收敛速度更快。

基金项目

国家自然科学基金(12001395)；太原师范学院研究生教育创新项目(SYYJSYC-2530)；太原师范学院教改项目(JGLX25078)。

参考文献

[1] 龙建辉. 求解几类非线性矩阵方程的数值算法[D]: [博士学位论文]. 长沙: 湖南大学, 2008.

[2] Higham, N.J. (2008) Functions of Matrices. Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9780898717778

[3] Horn, R.A. and Johnson, C.R. (1991) Topics in Matrix Analysis. Cambridge University Press. https://doi.org/10.1017/cbo9780511840371

[4] Dieci, L., Morini, B. and Papini, A. (1996) Computational Techniques for Real Logarithms of Matrices. *SIAM Journal on Matrix Analysis and Applications*, **17**, 570-593. https://doi.org/10.1137/s0895479894273614

[5] Alefeld, G. and Schneider, N. (1982) On Square Roots of M-Matrices. *Linear Algebra and Its Applications*, **42**, 119-132. https://doi.org/10.1016/0024-3795(82)90143-4

[6] Lin, L. and Liu, Z. (2001) On the Square Root of an H-Matrix with Positive Diagonal Elements. *Annals of Operations Research*, **103**, 339-350. https://doi.org/10.1023/a:1012931928589

[7] Johnson, C.R., Okubo, K. and Reams, R. (2001) Uniqueness of Matrix Square Roots and an Application. *Linear Algebra and its Applications*, **323**, 51-60. https://doi.org/10.1016/s0024-3795(00)00243-3

[8] Liu, Z., Zhang, Y. and Ralha, R. (2007) Computing the Square Roots of Matrices with Central Symmetry. *Applied Mathematics and Computation*, **186**, 715-726. https://doi.org/10.1016/j.amc.2006.08.032

[9] Lu, C. and Gu, C. (2011) The Computation of the Square Roots of Circulant Matrices. *Applied Mathematics and Computation*, **217**, 6819-6829. https://doi.org/10.1016/j.amc.2011.01.018

[10] Björck, A. and Hammarling, S. (1983) A Schur Method for the Square Root of a Matrix. *Linear Algebra and Its Applications*, **52**, 127-140. https://doi.org/10.1016/0024-3795(83)90010-1

[11] Higham, N.J. (1987) Computing Real Square Roots of a Real Matrix. *Linear Algebra and Its Applications*, **88**, 405-430. https://doi.org/10.1016/0024-3795(87)90118-2

[12] 杨壮, 彭振赟, 牟继萍. 一种新的矩阵平方根的迭代算法[J]. 桂林电子科技大学学报, 2014, 34(1): 60-64.

[13] Sharma, P. and Kansal, M. (2024) An Efficient Iterative Method for Matrix Sign Function with Application in Stability Analysis of Control Systems Using Spectrum Splitting. *Mathematical Methods in the Applied Sciences*, **47**, 9450-9468. https://doi.org/10.1002/mma.10077

[14] 张姗梅, 刘雁. 正定矩阵的算术平方根[J]. 高等数学研究, 2012, 15(1): 10-13.

[15] Guan, J., Shao, R. and Zubair, A. (2022) An Iteration Method for the Square Root of M-Matrix. *Mathematica Applicata*, **35**, 386-393.

[16] Song, Y., Sebe, N. and Wang, W. (2025) Fast Differentiable Matrix Square Root and Inverse Square Root. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **45**, 7367-7380. https://doi.org/10.1109/tpami.2022.3216339

[17] Higham, N.J. (1986) Newton's Method for the Matrix Square Root. *Mathematics of Computation*, **46**, 537-549. https://doi.org/10.1090/s0025-5718-1986-0829624-5

[18] Fasi, M., Higham, N.J. and Liu, X. (2025) Computing the Square Root of a Low-Rank Perturbation of the Scaled Identity Matrix. *SIAM Journal on Matrix Analysis and Applications*, **44**, 156-174. https://doi.org/10.1137/22m1471559

[19] Higham, N.J. (1997) Stable Iterations for the Matrix Square Root. *Numerical Algorithms*, **15**, 227-242. https://doi.org/10.1023/a:1019150005407

[20] Meini, B. (2004) The Matrix Square Root from a New Functional Perspective: Theoretical Results and Computational Issues. *SIAM Journal on Matrix Analysis and Applications*, **26**, 362-376. https://doi.org/10.1137/s0895479803426656

[21] 严慧玲, 肖林, 周文辉. 梯度神经网络在求解矩阵平方根中的应用[J]. 计算机技术与发展, 2017, 27(2): 155-157+162.

[22] Gawlik, E.S. (2019) Zolotarev Iterations for the Matrix Square Root. *SIAM Journal on Matrix Analysis and Applications*, **40**, 696-719. https://doi.org/10.1137/18m1178529

[23] 王志欣, 关晋瑞. 非对称代数 Riccati 方程的一类不精确迭代解法[J]. 青海师范大学学报(自然科学版), 2022, 38(4): 5-11.

[24] 关晋瑞, 王志欣, 李宣达. 正则 M-矩阵的平方根[J]. 太原师范学院学报(自然科学版), 2023, 22(3): 1-4+32.

[25] Berman, A. and Plemmons, R.J. (1994) Nonnegative Matrices in the Mathematical Sciences. Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9781611971262

[26] Guo, C. (2013) On Algebraic Riccati Equations Associated with M-Matrices. *Linear Algebra and Its Applications*, **439**, 2800-2814. https://doi.org/10.1016/j.laa.2013.08.018

[27] Wang, Z., Guan, J. and Zubair, A. (2024) A Structure-Preserving Doubling Algorithm for the Square Root of Regular M-Matrix. *Electronic Research Archive*, **32**, 5306-5320. https://doi.org/10.3934/era.2024245