mathematics

MDPI

Article

On the Calculation of the Moore–Penrose and Drazin Inverses: Application to Fractional Calculus

Khosro Sayevand [1], Ahmad Pourdarvish [2], José A. Tenreiro Machado [3,*] and Raziye Erfanifar [1]

[1] Faculty of Mathematical Sciences, Malayer University, Malayer P.O. Box 16846-13114, Iran; ksayehvand@malayeru.ac.ir (K.S.); raziye.erfanifar@stu.malayeru.ac.ir (R.E.)
[2] Faculty of Mathematical Sciences, Department of Statistics, Mazandaran University, Mazandaran P.O. Box 47416-135534, Iran; a.pourdarvish@umz.ac.ir
[3] Department of Electrical Engineering, Institute of Engineering, Polytechnic of Porto, 4249-015 Porto, Portugal
* Correspondence: jtm@isep.ipp.pt

Abstract: This paper presents a third order iterative method for obtaining the Moore–Penrose and Drazin inverses with a computational cost of $\mathcal{O}(n^3)$, where $n \in \mathbb{N}$. The performance of the new approach is compared with other methods discussed in the literature. The results show that the algorithm is remarkably efficient and accurate. Furthermore, sufficient criteria in the fractional sense are presented, both for smooth and non-smooth solutions. The fractional elliptic Poisson and fractional sub-diffusion equations in the Caputo sense are considered as prototype examples. The results can be extended to other scientific areas involving numerical linear algebra.

Keywords: Caputo sense; convergence order; Drazin inverse; fractional calculus; iterative method; Moore–Penrose inverse; non-smooth solution

Citation: Sayevand, K.; Pourdarvish, A.; Machado, J.A.T.; Erfanifar, R. On the Calculation of the Moore–Penrose and Drazin Inverses: Application to Fractional Calculus. *Mathematics* **2021**, *9*, 2501. https://doi.org/10.3390/math9192501

Academic Editor: Christopher Goodrich

Received: 12 August 2021
Accepted: 30 September 2021
Published: 6 October 2021

1. Introduction

Computing the inverse matrix, particularly for a high dimension, has been a time consuming task. Therefore, numerical methods are important for the calculation of the inverse of a matrix, and numerical iterative algorithms have a special role among the available techniques.

The Moore–Penrose inverse of a matrix $A \in \mathbb{C}^{m\times n}$, denoted by $A^\dagger \in \mathbb{C}^{n\times m}$, is the unique matrix X that obeys the four conditions [1]

$$AXA = A, \quad XAX = X, \quad (AX)^* = AX, \quad (XA)^* = XA, \tag{1}$$

where A^* is the conjugate transpose of A. If $rank(A) = \min\{m, n\}$, then

$$A^\dagger = \begin{cases} (A^*A)^{-1}A^*, & m > n, \\ A^{-1}, & m = n, \\ A^*(AA^*)^{-1}, & m < n. \end{cases} \tag{2}$$

We find in the published literature a number of different iterative methods for computing the Moore–Penrose inverse. The most common approach for the approximate inverse, A^{-1}, is the Newton's iterative method (NM):

$$V_{r+1} = V_r(2I - AV_r), \quad r = 0, 1, 2, \cdots, \tag{3}$$

where I is the identity matrix. For more details, interested readers can see [2,3].

Li et al. [4] investigated the following third-order method, known as Chebyshev's iterative method:

$$\begin{cases} W_r = AV_r, \\ V_{r+1} = V_r\big(3I - W_r(3I - W_r)\big). \end{cases} \tag{4}$$

In addition, Toutounian and Soleymani [5] proposed another iterative method to find A^{-1} of the fourth order given by

$$\begin{cases} W_r = AV_r, \\ V_{r+1} = 0.5V_r\Big(9I - W_r\big(16I - W_r(14I - W_r(6I - W_r))\big)\Big), \end{cases} \tag{5}$$

Pan et al. [6] investigated the following eighteenth-order scheme:

$$\begin{cases} W_r = AV_r, \quad Z_r = I - W_r, \quad P_r = Z_r^2, \quad U_r = P_r^2, \\ M_r = (I + c_1P_r + U_r)(I + c_2P_r + U_r), \\ T_r = M_r + c_3P_r, \quad S_r = M_r + d_1P_r + d_2U_r, \\ V_{r+1} = V_r\Big((I + Z_r)(T_rS_r + \mu P_r + \psi U_r)\Big), \end{cases} \tag{6}$$

where

$$\begin{cases} c_1 = \frac{1}{4}(\sqrt{27 - 2\sqrt{93}} - 1), \quad c_2 = \frac{1}{2}(1 - \sqrt{27 - \sqrt{93}}), \\ c_3 = \frac{1}{496}(5\sqrt{93} - 93), \quad d_1 = \frac{1}{496}(-93 - 5\sqrt{93}), \\ d_2 = -\frac{93}{4}, \quad \mu = \frac{3}{8}, \quad \psi = \frac{321}{1984}. \end{cases} \tag{7}$$

Esmaieli et al. [7] proposed the second-order method

$$\begin{cases} W_r = AV_r, \\ V_{r+1} = V_r\big(5.5I - W_r(8I - 3.5W_r)\big), \end{cases} \tag{8}$$

which is superior in terms of computational efficiency.

To initialize these algorithms, an initial matrix V_0 was introduced by Pan et al. [8]

$$V_0 = \alpha A^*, \;\text{ where }\; \alpha = \frac{1}{\|A\|_1\|A\|_\infty}. \tag{9}$$

In 1958, a different kind of generalized inverse was introduced by Drazin [9]. This definition does not have flexibility in the rings and semi-groups of associations but commutes with the element. The importance of this type of inverse and its calculation was later discussed by Wilkinson [10], and several researchers proposed direct or iterative methods for calculating the solution of this problem [11–14]. In this paper, a characterization of the Drazin inverse in the scope of fractional calculus is investigated.

The paper is organized as follows. Section 2 introduces the essential concepts, fundamental definitions, and properties of fractional calculus. Sections 3 and 4 analyse the performance of a novel iterative method for obtaining the Moore–Penrose and Drazin inverses. Section 5 introduces the error measurement. Section 6 compares the numerical results of the proposed approach with other available schemes, especially for high dimensional values. Section 7 highlights several applications of the new method and provides a numerical assessment of their effectiveness in a fractional sense. Finally, Section 8 presents the main conclusions.

Table 1 lists the abbreviations and acronyms used in the follow-up.

Table 1. List of abbreviations and acronyms used in the paper.

The Abbreviations	Description
NM	Newton's method (3)
CH	Chebyshev method (4)
TS	Method (5)
$E1$	Method (6)
$E2$	Method (8)
$E3$	Method (22)
CO	Convergence order of method
MM	Number of operations for $Q_{n\times n}.R_{n\times n}$
PM	Number of operations for $Q_{n\times n}+R_{n\times n}$
SM	Number of operations for $\beta Q_{n\times n}$
IM	Number of operations for $\gamma I_{n\times n}+Q_{n\times n}$
CPU	CPU time spent
PDE	Partial differential equation
FDE	Fractional differential equation

2. Fractional Calculus

In recent decades, fractional calculus and fractional differential equations (FDE) have had a significant impact in science, with particular emphases in system dynamical modelling [15–24]. This mathematical tool generalises the standard calculus, and several definitions of fractional derivatives and integrals have been proposed.

Let $\xi(x)$ be defined as a function on the interval $[a,b]$. The Riemann–Liouville integral is defined by

$$I^{\gamma}=\frac{1}{\Gamma(\gamma)}\int_a^x f(t)(x-t)^{\gamma-1}dt, \tag{10}$$

where Γ is the gamma function, and a is an arbitrary but fixed base point. The γth order $(m-1<\gamma<m)$ left and right sided Riemann–Liouville fractional derivatives of $\xi(x)$ are defined as

$${}_a^R D_x^{\gamma}\xi(x)=\frac{1}{\Gamma(m-\gamma)}\frac{d^m}{dx^m}\int_a^x\frac{\xi(\tau)}{(x-\tau)^{\gamma-m+1}}d\tau, \tag{11}$$

$${}_x^R D_b^{\gamma}\xi(x)=\frac{(-1)^m}{\Gamma(n-\gamma)}\frac{d^m}{dx^m}\int_x^b\frac{\xi(\tau)}{(x-\tau)^{\gamma-m+1}}d\tau. \tag{12}$$

The Riemann–Liouville fractional derivative and integral played an important role in the development of theoretical problems of fractional calculus. However, since the solution of FDE requires initial conditions with fractional derivatives, they pose difficulties in their application. In 1967, the Caputo fractional derivative was formulated [17]:

$$\frac{\partial^{\gamma}\xi(x,t)}{\partial t^{\gamma}}=\frac{1}{\Gamma(m-\gamma)}\int_0^t\frac{1}{(t-\tau)^{\gamma-m+1}}\frac{\partial^m\xi(x,\tau)}{\partial\tau^m}d\tau,\ \gamma\in(m-1,m),\ m\in\mathbb{N}. \tag{13}$$

This definition simplifies the initial condition problem. The relationship between the Riemann–Liouville and Caputo fractional derivatives is as follows:

$${}_a^{RL}D_x^{\gamma}\xi(x)={}_a^{C}D_x^{\gamma}\xi(x)+\sum_{k=0}^{m-1}\frac{f^{(k)(a)(t-a)^{k-\gamma}}}{\Gamma(k+1-\gamma)},\quad m-1\le\gamma<m. \tag{14}$$

Nonetheless, during the discretization of the FDE, we obtain a matrix, and therefore, a problem of linear algebra has a relationship with the solution of FDE.

3. New Iterative Method

Iterative methods for solving nonlinear equations are pervasive in applied mathematics, and many researchers have studied a variety of algorithms [25–28], keeping in mind that the efficiency of the method is of key importance. The existence of a function derivative in the algorithm often poses constraints and increases the computational cost. Therefore, the use of function high derivatives is usually avoided. We suppose that the function $F(\chi)$ has a simple root at a and that χ_0 is an initial guess sufficiently close to a. To solve the equation $F(\chi) = 0$, we consider the iterative algorithm

$$\begin{aligned}\chi_{r+1} = \chi_r \quad &- \quad \frac{F(\chi_r)}{F'(\chi_r)}\Big(1 + \frac{F''(\chi_r)F(\chi_r)}{2F'(\chi_r)^2}\Big) - \frac{F(\chi_r)^3}{4F'(\chi_r)^4}\Big(\frac{F''(\chi_r)^2}{2F'(\chi_r)} \\ &+ \quad \frac{F''''(\chi_r)F(\chi_r)}{F'(\chi_r)} - \frac{F'''(\chi_r)}{6}\Big), \quad r = 0, 1, \cdots. \end{aligned} \tag{15}$$

In the following, the convergence analysis of this method is investigated.

Theorem 1. *Suppose that $F : D \subseteq \mathbb{R} \to \mathbb{R}$ is sufficiently differentiable in a neighbourhood of $a \in D$ and that a is a simple zero of $F(\chi) = 0$. The iterative method (15) converges to a with convergence order three. The error equation is given by*

$$\epsilon_{r+1} = \frac{3}{4}\Big(2J_2^2 - J_3\Big)(\epsilon_r)^3 + \mathcal{O}(\epsilon_r^4), \tag{16}$$

where $J_i = \dfrac{1}{i!}\dfrac{F^{(i)}(a)}{F'(a)}$ *for* $i \geq 2$.

Proof. Based on the Taylor expansion for F about a, we can write

$$\begin{aligned} F(\chi_r) \quad &= F'(a)[\epsilon_r + J_2\epsilon_r^2 + J_3\epsilon_r^3 + J_4\epsilon_r^4 + J_5\epsilon_r^5 + J_6\epsilon_r^6 + \mathcal{O}(\epsilon_r^7)], \\ F'(\chi_r) \quad &= F'(a)[1 + 2J_2\epsilon_r + 3J_3\epsilon_r^2 + 4J_4\epsilon_r^3 + 5J_5\epsilon_r^4 + 6J_6\epsilon_r^5 + \mathcal{O}(\epsilon_r^6)], \\ &\vdots \\ F''''(\chi_r) \quad &= F'(a)[24J_4 + 120J_5\epsilon_r + 360J_6\epsilon_r^2 + \mathcal{O}(\epsilon_r^3)]. \end{aligned} \tag{17}$$

From the above relations, we have

$$\begin{aligned} \frac{F(\chi_r)}{F'(\chi_r)} \quad &= \epsilon_r - J_2\epsilon_r^2 + 2(J_2^2 - J_3)\epsilon_r^3 + \mathcal{O}(\epsilon_r^4), \\ \frac{F''(\chi_r)F(\chi_r)}{2F(\chi_r)^2} \quad &= J_2\epsilon_r + (-3J_2^2 + 3J_3)\epsilon_r^2 + \mathcal{O}(\epsilon_r^3)], \end{aligned} \tag{18}$$

$$\begin{aligned} \frac{F(\chi_r)^3}{4F'(\chi_r)^4} \quad &= \frac{1}{4F'(a)}[\epsilon_r^3 - 5J_2\epsilon_r^4 + \mathcal{O}(\epsilon_r^5)], \\ \frac{F''(\chi_r)^2}{2F'(\chi_r)} \quad &= F'(a)[2J_2^2 - 4(J_2^3 - 3J_2J_3)\epsilon_r + \mathcal{O}(\epsilon_r^2)], \\ \frac{F''''(\chi_r)F(\chi_r)}{F'(\chi_r)} \quad &= 24F'(a)[J_4\epsilon_r - (J_2J_4 - 5J_5)\epsilon_r^2 + \mathcal{O}(\epsilon_r^3)]. \end{aligned} \tag{19}$$

Then, according to (15), we can write

$$\epsilon_{r+1} = \frac{3}{4}\Big(2J_2^2 - J_3\Big)\epsilon_r^3 + \mathcal{O}(\epsilon_r^4). \tag{20}$$

Let $F(V) = V^{-1} - A$. Then, the following iterative method is obtained from (15):

$$V_{r+1} = \frac{1}{4}V_r\Big(37I - 111AV_r + 151(AV_r)^2 - 97(AV_r)^3 + 24(AV_r)^4\Big), \tag{21}$$

or

$$\begin{cases} \vartheta_r = AV_r, \\ \xi_r = \vartheta_r^2, \\ V_{r+1} = \frac{1}{4}V_r\Big(37I - 111\vartheta_r + \xi_r\big(151I - 97\vartheta_r + 24\xi_r\big)\Big). \end{cases} \tag{22}$$

Note: It is pointed out that in the body of this paper, methods (6), (8) and (22) are represented by the acronyms $E1$, $E2$, and $E3$, respectively. □

Theorem 2. *Suppose that $A \in \mathbb{C}^{n\times n}$ is a nonsingular matrix and that the initial approximation V_0 satisfies*

$$\|I - AV_0\| < 1. \tag{23}$$

Then, the iterative method (22) converges to A^{-1} with third order.

Proof. The proof is similar to that of Theorem 2.1 in [29]. □

Now, consider $t_r = \|A\epsilon_r\|$ and $s_r = \|E_r\|$. In the following, we show the convergence properties of the iterative method, namely the behaviour of the sequences t_r and s_r.

Corollary 1. *Assume that the conditions of (9) hold. If $\lim_{r\to\infty} t_r = 0$ and $\lim_{r\to\infty} s_r = 0$, then, for the iterative method (22), it yields*

$$\lim_{r\to\infty} \frac{t_{r+1}}{t_r^3} = \lim_{r\to\infty} \frac{s_{r+1}}{s_r^3} = \frac{3}{4}. \tag{24}$$

Proof. From Theorem 2, we have

$$A\epsilon_{r+1} = \frac{3}{4}(A\epsilon_r)^3 - \frac{23}{4}(A\epsilon_r)^4 + 6(A\epsilon_r)^5. \tag{25}$$

Consequently,

$$t_{r+1} = \|A\epsilon_{r+1}\| \geq \frac{3}{4}\|A\epsilon_r\|^3 - \frac{23}{4}\|A\epsilon_r\|^4 + 6\|A\epsilon_r\|^5 = t_r^3\left(\frac{3}{4} - \frac{23}{4}t_r + 6t_r^2\right), \tag{26}$$

or

$$t_{r+1} = \|A\epsilon_{r+1}\| \leq \frac{3}{4}\|A\epsilon_r\|^3 + \frac{23}{4}\|A\epsilon_r\|^4 + 6\|A\epsilon_r\|^5 = t_r^3\left(\frac{3}{4} + \frac{23}{4}t_r + 6t_r^2\right), \tag{27}$$

which implies that

$$\frac{3}{4} - \frac{23}{4}t_r + 6t_r^2 \leq \frac{t_{r+1}}{t_r^3} \leq \frac{3}{4} + \frac{23}{4}t_r + 6t_r^2, \text{ or } \lim_{r\to\infty} \frac{t_{r+1}}{t_r^3} = \frac{3}{4}. \tag{28}$$

Again, by an argument similar to Theorem 2, we have

$$\frac{3}{4} - \frac{23}{4}s_r + 6s_r^2 \leq \frac{s_{r+1}}{s_r^3} \leq \frac{3}{4} + \frac{23}{4}s_r + 6s_r^2, \text{ or } \lim_{r\to\infty} \frac{s_{r+1}}{s_r^3} = \frac{3}{4}. \tag{29}$$

□

Theorem 3. *Suppose that A is a nonsingular matrix. If $AV_0 = V_0A$, then for the sequence (22), we have*

$$AV_i = V_iA, \quad i = 1, 2, \cdots. \tag{30}$$

Proof. The proof is similar to the proofs presented in [30]. □

Lemma 1. *For the sequence* $\{V_k\}_{k=0}^{k=\infty}$*, generated by the iterative method (22), it holds that*

$$(V_kA)^* = V_kA, \quad (AV_k)^* = AV_k, \; V_kAA^\dagger = V_k, \; A^\dagger AV_k = V_k. \tag{31}$$

Proof. The proof is similar to Lemma 2.1 in [31]. □

Theorem 4. *According to the same assumptions as in Theorem 2, the iterative method (22) is asymptotically stable.*

Proof. This theorem is similar to those adopted for a general family of methods in [32]. Thus, the proof is omitted. □

Lemma 2. *[33] For* $M \in \mathbb{C}^{n\times n}$ *and any given* $\xi > 0$*, there is at least one matrix norm* $\|\cdot\|$ *such that*

$$\rho(M) \leq \|M\| \leq \rho(M) + \xi, \tag{32}$$

where $\rho(M) = max|\lambda_i|$ *and* λ_i *are eigenvalues of matrix* M*.*

Lemma 3 ([34])**.** *For* $P, S \in \mathbb{C}^{n\times n}$*, such that* $P = P^2$ *and* $PS = SP$*, it holds that*

$$\rho(PS) \leq \rho(S). \tag{33}$$

Theorem 5. *Let* $A \in \mathbb{C}_r^{m\times n}$ *and let us consider that* $\sigma_1 > \sigma_2 > \cdots > \sigma_r > 0$ *are the singular values of* A*. Then, (22) converges to the Moore–Penrose inverse* $A^\dagger$ *in the third order, provided that* $V_0 = \frac{A^*}{C}$*, where* $C > \sigma_1^2$ *is a constant.*

Proof. According to Lemma 1, we have

$$\|V_{r+1} - A^\dagger\| = \|V_{r+1}AA^\dagger - A^\dagger AA^\dagger\| \leq \|V_{r+1}A - A^\dagger A\|\|A^\dagger\|, \tag{34}$$

and if $E_r = V_r - A^\dagger$, then $A^\dagger AE_rA = E_rA$. From the conditions of the Moore–Penrose inverse, E_r, and from (22), we have

$$(I - A^\dagger A)^t = I - A^\dagger A, \;\; t = 2,3, \;\; (I - A^\dagger A)E_rA = 0, \;\; E_rA(I - A^\dagger A) = 0, \tag{35}$$

and

$$\begin{aligned} E_{r+1}A &= \Big[\frac{1}{4}V_r\Big(37I - 111AV_r + 151(AV_r)^2 - 97(AV_r)^3 + 24(AV_r)^4\Big) - A^\dagger\Big]A \\ &= -(I - V_rA)^3\Big(\frac{3}{4}I - \frac{23}{4}(I - V_rA) + 6(I - V_rA)^2\Big) + I - A^\dagger A \\ &= -(I - A^\dagger A - E_rA)^3\Big(\frac{3}{4}I - \frac{23}{4}(I - A^\dagger A - E_rA) + 6(I - A^\dagger A - E_rA)^2\Big) \\ &\quad + I - A^\dagger A \\ &= -\Big((I - A^\dagger A) - 3(I - A^\dagger A)E_rA + 3(I - A^\dagger A)(E_rA) - (E_rA)^3\Big) \\ &\qquad \times \Big(\frac{3}{4}I - \frac{23}{4}(I - A^\dagger A) + \frac{23}{4}(E_rA) + 6(I - A^\dagger A) \\ &\qquad - 12(I - A^\dagger A)(E_rA) + 6(E_rA)^2\Big) + I - A^\dagger A. \end{aligned} \tag{36}$$

So, it is proved that

$$E_{r+1}A = \Big(\frac{3}{4}I + \frac{23}{4}(E_rA) + 6(E_rA)^2\Big)(E_rA)^3. \tag{37}$$

Now, consider $P = A^\dagger A$ and $S = V_0A - I$, so that $P^2 = P$ and

$$\begin{aligned} PS &= A^\dagger A(V_0A - I) = A^\dagger AV_0A - A^\dagger A = (A^\dagger A)^*V_0A - A^\dagger A \\ &= V_0A - A^\dagger A = V_0AA^\dagger A - A^\dagger A = (V_0A - I)A^\dagger A = SP. \end{aligned} \tag{38}$$

Therefore, according to Lemma 3

$$\rho\big((V_0 - A^{\dagger})A\big) = \rho\Big(\big(\frac{A^*}{C} - A^{\dagger}\big)A\Big) \leq \rho\Big(\frac{A^*}{C}A - I\Big) = \max_{1\leq i\leq r}\Big|1 - \lambda_i\Big(\frac{A^*}{C}A\Big)\Big|. \tag{39}$$

Since $C > \sigma_1^2$, we have

$$\max_{1\leq i\leq r}\Big|1 - \lambda_i\Big(\frac{A^*}{C}A\Big)\Big| < 1, \tag{40}$$

and from Lemma 2,

$$\Big\|(V_0 - A^{\dagger})A\Big\| \leq \rho\Big((V_0 - A^{\dagger})A\Big) + \xi < 1. \tag{41}$$

Consequently, according to (34) and (37), we obtain $\lim_{k\to\infty}\|V_k - A^{\dagger}\| = 0$ with the third order. □

Theorem 6. *The sequence V_r produced by (22) and with (9) satisfies*

$$\mathcal{R}(V_r) = \mathcal{R}(A^*), \quad \mathcal{N}(V_r) = \mathcal{N}(A^*), \tag{42}$$

for $r \geq 0$, where $\mathcal{R}(\cdot)$ and $\mathcal{N}(\cdot)$ denote the range and the null space of the matrix, respectively.

Proof. Since $V_0 = \alpha A^*$, the theorem obviously holds for $r = 0$. Suppose that $y \in \mathcal{N}(V_r)$ is an arbitrary vector. According to the method (22), we have

$$V_{r+1}y = \frac{1}{4}\Big(37V_r y - 111V_rAV_r y + 151V_r(AV_r)^2 y - 97(V_rAV_r)^3 y + 24V_r(AV_r)^4 y\Big) = 0. \tag{43}$$

As we know $y \in \mathcal{N}(V_{r+1})$, we can conclude that $\mathcal{N}(V_r) \subseteq \mathcal{N}(V_{r+1})$. Similarly we have $\mathcal{R}(V_r) \supseteq \mathcal{R}(V_{r+1})$. Therefore, by mathematical induction we can write

$$\mathcal{N}(V_r) \supseteq \mathcal{N}(V_0) = \mathcal{N}(\mathcal{A}^*), \quad \mathcal{R}(V_r) \subseteq \mathcal{R}(V_0) = \mathcal{R}(A^*). \tag{44}$$

To prove the equality, let

$$\mathcal{N} = \bigcup_{r\in\mathbb{N}_0}\mathcal{N}(V_r). \tag{45}$$

Suppose that $y \in \mathcal{N}$. Then, $y \in \mathcal{N}(V_{r_0})$ for $r_0 \in \mathbb{N}_0$. Since $y \in \mathcal{N}(V_r)$ for every $r \geq r_0$, then $V_r y = 0$, and according to Theorem 2,

$$Vy = \lim_{r\to+\infty} V_r y = 0. \tag{46}$$

Finally, $y \in \mathcal{N}(V) = \mathcal{N}(A^*)$ and $\mathcal{N} \subseteq \mathcal{N}(A^*)$. On the other hand, it comes to be that

$$\mathcal{N}(A^*) \subseteq \mathcal{N}(V_r) \subseteq \mathcal{N} \subseteq \mathcal{N}(A^*), \tag{47}$$

and so $\mathcal{N}(V_r) = \mathcal{N}(A^*)$.

Now, according to the relation

$$\dim\ \mathcal{R}(V_r) = m - \dim\ \mathcal{N}(V_r) = m - \dim\ \mathcal{N}(A^*) = \dim\ \mathcal{R}(A^*), \tag{48}$$

and $\mathcal{R}(V_r) \subseteq \mathcal{R}(A^*)$, we conclude that $\mathcal{R}(V_r) = \mathcal{R}(A^*)$. □

Theorem 7. *Let $\{V_k\}_{k=0}^{\infty}$ generated by method (22), for all $\widehat{V}_k$, such that*

$$\widehat{V}_k = V_k + \Delta_k, \tag{49}$$

where Δ_k is a numerical perturbation of the k-th exact iterate V_k having a sufficiently small norm, we can ignore quadratic and higher order terms in $\mathcal{O}(\Delta_k^2)$, and one has

$$\|\Delta_k\| < \|\Delta_k\|\|A\|\|V_k\| + \mathcal{O}(\|\Delta_k\|). \tag{50}$$

Proof. Let $\widehat{E}_k = I - A\widehat{V}_k$. Then,

$$\|\widehat{E}_k^j\| = \|(E_k - A\Delta_k)^j\| \leq \|E_k - A\Delta_k\|^j \leq (\|E_k\| + \|A\Delta_k\|)^j = C_0^j, \quad j = 1,2,3, \tag{51}$$

where $C_0 = \|E_k\| + \|A\Delta_k\| = \|E_k\| + \mathcal{O}(\|\Delta_k\|)$. Furthermore,

$$\begin{aligned} \|\widehat{E}_k^j - E_k^j\| &= \|(E_k - A\Delta_k)^j - E_k^j\| \leq (\|E_k\| + \|A\Delta_k\|)^j - \|E_k\|^j \\ &= \|A\Delta_k\| \left(\sum_{i=0}^{j-1} \binom{j}{j-1-i} \|A\Delta_k\|^i \|E_k\|^{j-1-i} \right), \end{aligned} \tag{52}$$

meaning

$$\|\widehat{E}_k^j - E_k^j\| \leq T_j \|A\Delta_k\|, \tag{53}$$

where

$$T_j = \sum_{i=0}^{j-1} \binom{j}{j-1-i} \|A\Delta_k\|^i \|E_k\|^{j-1-i} = j\|E_k\|^{j-1} + \mathcal{O}(\|\Delta_k\|), \tag{54}$$

and we have

$$\begin{aligned} \Delta_{k+1} &= \widehat{V}_{k+1} - V_{k+1} \\ &= \frac{1}{4}\widehat{V}_k \left(37I - 111A\widehat{V}_k + 151(A\widehat{V}_k)^2 - 97(A\widehat{V}_k)^3 + 24(A\widehat{V}_k)^4 \right) \\ &\quad - \frac{1}{4}V_k \left(37I - 111AV_k + 151(AV_k)^2 - 97(AV_k)^3 + 24(AV_k)^4 \right) \\ &= \frac{1}{4}\widehat{V}_k \left(4I + 4\widehat{E}_k + 4\widehat{E}_k^2 + \widehat{E}_k^3 + 24\widehat{E}_k^4 \right) \\ &- \frac{1}{4}V_k \left(4I + 4E_k + 4E_k^2 + E_k^3 + 24E_k^4 \right) \\ &= \Delta_k (I + \widehat{E}_k + \widehat{E}_k^2 + \frac{1}{4}\widehat{E}_k^3 + 6\widehat{E}_k^4) \\ &\quad + V_k \left(I + (\widehat{E}_k - E_k) + (\widehat{E}_k^2 - E_k^2) + \frac{1}{4}(\widehat{E}_k^3 - E_k^3) + 6(\widehat{E}_k^6 - E_k^6) \right). \end{aligned} \tag{55}$$

Therefore,

$$\begin{aligned} \|\Delta_{k+1}\| &\leq \|\Delta_k\|(1 + \|\widehat{E}_k\| + \|\widehat{E}_k^2\| + \frac{1}{4}\|\widehat{E}_k^3\| + 6\|\widehat{E}_k^4\|) \\ &\quad + \|V_k\| \left(1 + \|\widehat{E}_k - E_k\| + \|\widehat{E}_k^2 - E_k^2\| + \frac{1}{4}\|\widehat{E}_k^3 - E_k^3\| + 6\|\widehat{E}_k^4 - E_k^4\| \right) \\ &= \|\Delta_k\|(1 + C_0 + C_0^2 + \frac{1}{4}C_0^3 + 6C_0^6) \\ &+ \|A\Delta_k\|\|V_k\| \left(1 + T_1 + T_2 + T_3 + 6T_4 \right) \\ &< \|\Delta_k\|\|A\|\|V_k\| + \mathcal{O}(\|\Delta_k\|). \end{aligned} \tag{56}$$

This expression yields the claimed estimates (50) for the numerical perturbation at iteration loop $k + 1$. □

Proof. The proof is straightforward. □

Corollary 2. *The computational cost of the iterative method (22) is $\mathcal{O}(n^3)$.*

Proof. To calculate the computational cost of the suggested method, the following facts hold. Suppose that $Q_{n\times n}$ and $R_{n\times n}$ are given matrices. Then, we verify that n^3 operations are needed to compute $Q_{n\times n}.R_{n\times n}$, n^2 operations are needed for $Q_{n\times n} + R_{n\times n}$ and $\beta Q_{n\times n}$, and n operations for $\gamma I_{n\times n} + Q_{n\times n}$. Consequently, the sum of all required operations in (22) is $4n^3 + 6n^2 + 2n$, and we have $\mathcal{O}(n^3)$. □

Table 2 lists the convergence order (CO), the number of operations for $Q_{n\times n}R_{n\times n}$ (MM), $Q_{n\times n} + R_{n\times n}$ (PM), $\beta Q_{n\times n}$ (SM), $\gamma I_{n\times n} + Q_{n\times n}$ (IM), and the computational cost (CC) in every iteration of the methods $E1$ and $E3$.

Table 2. Computational cost for every iteration of the methods.

Method	CO	MM	PM	SM	IM	CC
NM	2	2	0	0	1	$2n^3 + 2n$
CH	3	3	0	0	2	$3n^3 + 2n$
TS	4	5	0	1	4	$5n^3 + n^2 + 4n$
$E1$	18	7	7	7	4	$7n^3 + 14n^2 + 4n$
$E2$	2	3	0	1	2	$3n^3 + n^2 + 2n$
$E3$	3	4	2	4	2	$4n^3 + 6n^2 + 2n$

4. Application in Finding the Drazin Inverse

Drazin inverses were first introduced and used by Drazin himself in the study of abstract ring theory in finite dimensional algebra. Later, the definition of Drazin inverses was generalized to bounded linear operators in Banach spaces and was used to study linear abstract differential equations in Banach spaces [9,35]. Some of the most important applications of the Drazin inverse are Markov chains, control theory, singular differential and difference equations, and iterative methods in numerical linear algebra [36–38].

Definition 1. *The smallest non-negative integer $k = ind(\cdot)$ that holds*

$$rank(A^{k+1}) = rank(A^k), \tag{57}$$

is called the index of matrix A.

Definition 2. *Suppose that $A \in \mathbb{C}^{n\times n}$. Then, the Drazin inverse of A, denoted by A^D, is the matrix V, which holds in the following equations:*

$$A^k VA = A^k, \quad VAV = V, \quad AV = VA, \tag{58}$$

where $k = ind(A)$.

Li and Wei [39] proved that NM can be used to find the Drazin inverse of square matrices, and they proposed the initial matrix

$$V_0 = W_0 = \beta A^l, \quad l \geq ind(A) = k, \tag{59}$$

where β must satisfy the condition $\|I - AV_0\| < 1$.

We now consider the iterative method (22) for finding the Drazin inverse, with the initial matrix

$$V_0 = W_0 = \frac{2}{tr(A^{k+1})} A^k, \tag{60}$$

where $tr(\cdot)$ stands for the trace of the matrix.

Proposition 1 ([40]). *Let $P_{L,M}$ be the projector on a space L along a space M. Then,*

(i) $P_{L,M}Q = Q \Leftrightarrow \mathcal{R}(Q) \subseteq L$,

(ii) $QP_{L,M}Q = Q \Leftrightarrow \mathcal{N}(Q) \supseteq M$.

Theorem 8. *Suppose that A is a singular square matrix. Additionally, let the initial matrix be chosen by (60). Then, for W_r generated by the iterative method (22), the following asymptotic error estimate holds to find the Drazin inverse*

$$\|A^D - W_r\| \leq \mathcal{O}(\|A^D\|\|F_0\|^{3^r}), \tag{61}$$

where $F_0 = I - AW_0$.

Proof. Let $F_0 = I - AW_0$. Then, $F_r = I - AW_r$. Thus, we have

$$\begin{aligned} F_{r+1} &= I - AW_{r+1} = (I - AW_r)^3\left(\frac{3}{4}I - \frac{23}{4}(I - AW_r) + 6(I - AW_r)^2\right) \\ &= \frac{3}{4}F_r^3 - \frac{23}{4}F_r^4 + 6F_r^5. \end{aligned} \tag{62}$$

Using an arbitrary matrix norm of (62) results in

$$\|F_{r+1}\| \leq \frac{3}{4}\|F_r\|^3 + \frac{23}{4}\|F_r\|^4 + 6\|F_r\|^5. \tag{63}$$

Here, since $\|F_0\| < 1$, from relation (63), we have

$$\|F_1\| \leq \frac{3}{4}\|F_0\|^3 + \frac{23}{4}\|F_0\|^4 + 6\|F_0\|^5 \leq \mathcal{O}(\|F_0\|^3). \tag{64}$$

By continuing this process, we arrive at

$$\|F_{r+1}\| \leq \frac{3}{4}\|F_r\|^3 + \frac{23}{4}\|F_r\|^4 + 6\|F_r\|^5 \leq \mathcal{O}(\|F_r\|^3). \tag{65}$$

Thus, $\|F_{r+1}\| \leq \mathcal{O}(\|F_r)\|$ for every $r \geq 0$. Therefore, we obtain

$$\|F_r\|^3 \leq \mathcal{O}(\|F_0\|^{3^r}), \quad r \geq 0. \tag{66}$$

According to relation (59), we have $\mathcal{R}(W_0) \subseteq \mathcal{R}(A^k)$. In addition, the use of this result together with (21) implies that $\mathcal{R}(W_r) \subseteq \mathcal{R}(W_{r-1})$, and we can write

$$\mathcal{R}(W_r) \subseteq \mathcal{R}(A^k), \quad r \geq 0. \tag{67}$$

On the other hand, we have

$$W_{r+1} = \frac{1}{4}W_r\Big(37I - 111AW_r + 151(AW_r)^2 - 97(AW_r)^3 + 24(AW_r)^4\Big). \tag{68}$$

It is straightforward to verify that

$$\mathcal{N}(W_r) \supseteq \mathcal{N}(A^k), \quad r \geq 0. \tag{69}$$

According to Ben-Israel et al. [41], one can readily show that

$$AA^D = A^D A = P_{\mathcal{R}(A^k),\mathcal{N}(A^k)}, \tag{70}$$

and from Proposition 1 and expressions (67) and (69), we have

$$W_r AA^D = W_r = A^D AW_r, \quad r \geq 0. \tag{71}$$

Therefore, if the error matrix is $\delta_r = A^D - W_r$, then it follows that

$$\delta_r = A^D - W_r = A^D - A^D AW_r = A^D(I - AW_r) = A^D F_r, \tag{72}$$

and from (72) and (66), we have

$$\|\delta_r\| = \|A^D\|\|F_r\| \leq \mathcal{O}(\|A^D\|\|F_0^{3^r}\|), \tag{73}$$

which completes the proof. □

Corollary 3. *Assume that the condition of Theorem 8 and the following stabilization condition*

$$\|F_0\| \leq \|I - AW_0\| < 1 \tag{74}$$

are satisfied. Then, expression (22) converges to A^D.

Theorem 9. *(Stability) Suppose the same assumptions as in Theorem 8 hold. Then, the iterative method (22) has asymptotic stability for finding the Drazin inverse.*

Proof. The proof of asymptotic stability of the iterative method (22) is similar to that in [32]. Thus, the proof is omitted. □

5. Error Measurement

If the quantity $\overline{V}$ is viewed as an approximation to V, then the absolute (e_A) and relative (e_R) errors in the approximation are defined as

$$e_A = |V - \overline{V}|, \tag{75}$$

and

$$e_R = \frac{|V - \overline{V}|}{|V|}, \tag{76}$$

respectively. However, the absolute error is not useful for large sets, and the relative error can sometimes be misleading when $|V|$ is small. To avoid the need to choose between the absolute and relative errors, the following mixed error measure is often used in practice:

$$e = \frac{|V - \overline{V}|}{1 + |V|}. \tag{77}$$

The value of e in (77) is similar to the absolute error e_A when $|V| \ll 1$ and to the relative error e_R when $|V| \gg 1$ [42].

6. Numerical Results

In this section, we compare the results of the proposed approach with other schemes available in the literature. Since the comparison of the method $E2$ reported in [7] shows that it has better performance than others, we only compare our proposed method $E3$ with $E2$, NM, CH, TS, and $E1$. According to (77), the stop criterion is

$$\frac{\|V_{r+1} - V_r\|_\infty}{1 + \|V_r\|_\infty} < 10^{-10}. \tag{78}$$

We denote by CPU the required calculation time using Mathematica and by MM the number of matrix-matrix products. Furthermore for computing the inverse of a matrix and based on Method (22), the producers is presented in Algorithm 1.

Algorithm 1 Method (22) for computing the inverse of a matrix

Step 1: Input matrix $A \in \mathbb{C}^{n\times n}$.

Step 2: Take the initial matrix $V_0 = \dfrac{1}{\|A\|_1\|A\|_\infty}A^*$ and the tolerance $\varepsilon \geq 0$. Set $r := 0$.

Step 3: Let

$$\begin{aligned}\vartheta_r &= AV_r,\\ \xi_r &= \vartheta_r^2,\\ V_{r+1} &= \frac{1}{4}V_r\Big(37I - 111\vartheta_r + \xi_r\big(151I - 97\vartheta_r + 24\xi_r\big)\Big).\end{aligned} \tag{79}$$

Step 4: Stop if $\dfrac{\|V_{r+1}-V_r\|_\infty}{1+\|V_r\|_\infty} \leq \varepsilon$. Otherwise, $r := r+1$, and go to Step 3.

Example 1. *Consider a real-valued tri-diagonal matrix with dimension* 1000×1000*, where the diagonals are as follows:*

$$(1,360) = -2.35, \quad (1,1) = -2.35, \quad (700,1) = 1.85. \tag{80}$$

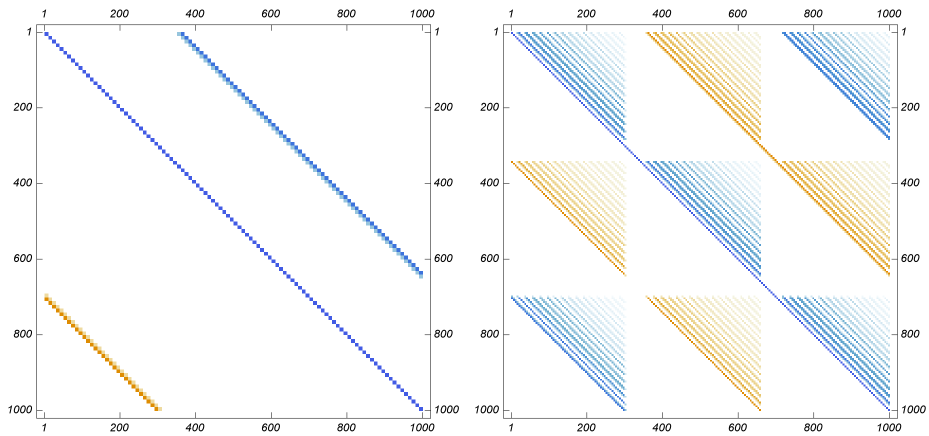
Example 2. *Consider the complex-valued tri-diagonal matrix with dimension* 1000×1000*, where the diagonals are as follows:*

$$(1,280) = 0.9 - 0.45i, \quad (1,1) = -1.25 + 0.14i, \quad (850,1) = -2.25 + 0.6i. \tag{81}$$

Example 3. *Consider the complex-valued tri-diagonal matrix with dimension* 2000×2000*, where the diagonals are as follows:*

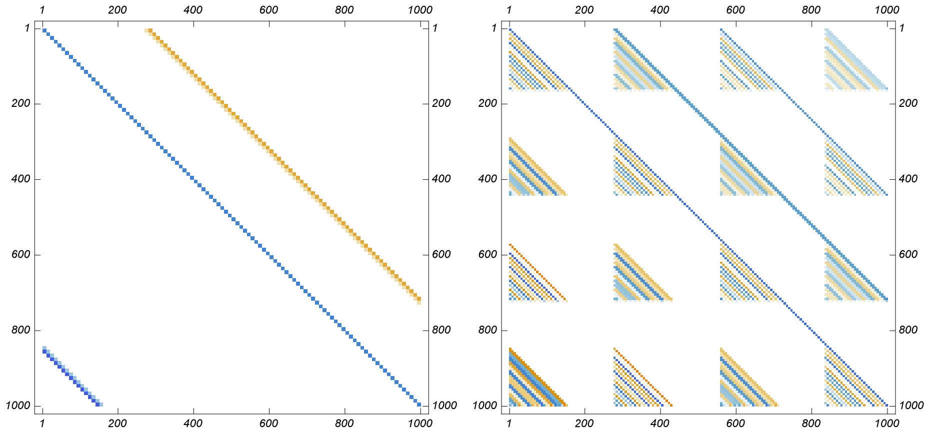
$$(1,420) = -6.5 + 0.25i, \quad (1,1) = -1.5 + 2.25i, \quad (1650,1) = 2.5 - 2i. \tag{82}$$

The results of Examples 1–3 are presented in Tables 3–5 and Figures 1–3.



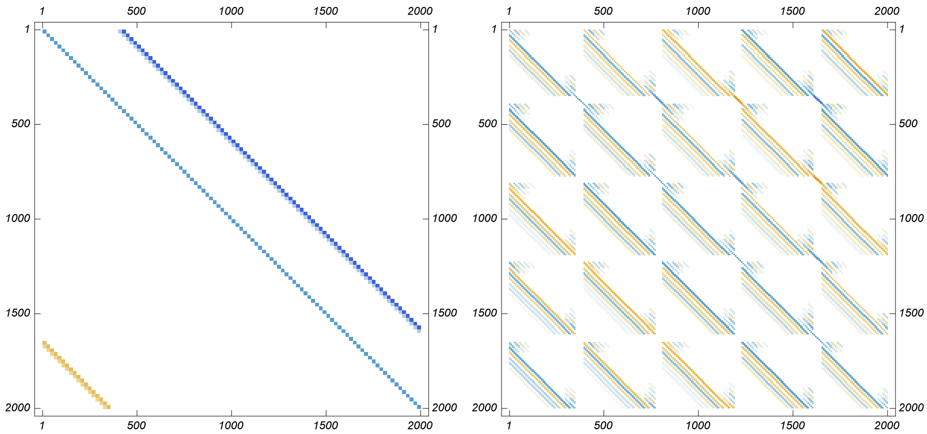
(**a**) (**b**)

Figure 1. Representation of the (**a**) matrix and (**b**) inverse matrix for Example (1).



(**a**) (**b**)

Figure 2. Representation of the (**a**) matrix and (**b**) inverse matrix for Example (2), respectively.



(**a**) (**b**)

Figure 3. Representation of the (**a**) matrix and (**b**) inverse matrix for Example (3), respectively.

Table 3. Results of Example (1).

Method	NM	CH	TS	$E1$	$E2$	$E3$
MM	32	33	40	35	33	28
CPU	0.1089	0.1102	0.1590	0.1423	0.1124	0.0936

Table 4. Results of Example (2).

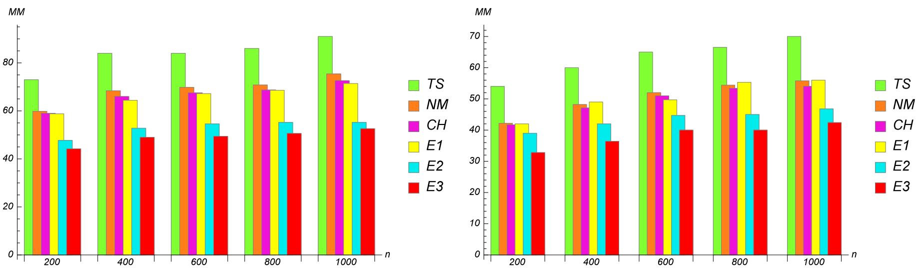
Method	NM	CH	TS	$E1$	$E2$	$E3$
MM	24	24	30	28	27	20
CPU	0.6092	0.6105	0.6701	0.6421	0.6122	0.5436

Table 5. Results of Example (3).

Method	NM	CH	TS	$E1$	$E2$	$E3$
MM	36	33	40	35	30	28
CPU	0.9082	1.1301	1.1625	1.1421	1.0122	0.8336

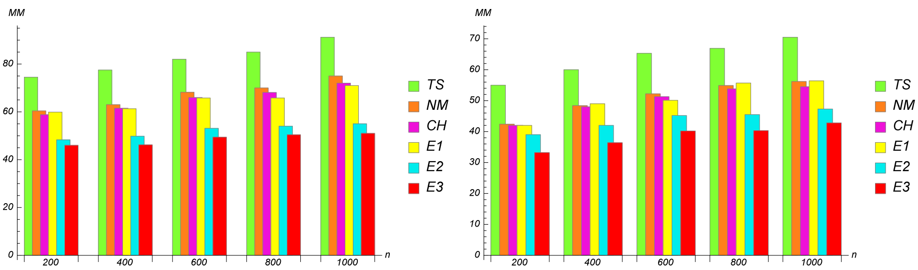
Example 4. *To evaluate the efficiency of the proposed method, we consider several real and complex random matrices with different dimensions. For each of sizes* $n \times n$ *and* $n \times (n+20)$*,* $n =$

$\{200, 400, 600, 800, 1000\}$, *we perform* 5 *random tests and compare average values of matrix multiplications. The results are presented in Figures 4 and 5.*



(**a**) (**b**)

Figure 4. The average MM for computing the Moore–Penrose inverse of real matrices (**a**) $(n \times n)$ and (**b**) $(n \times n+20)$ by the methods $\{TS, NM, CH, E1, E2, E3\}$.



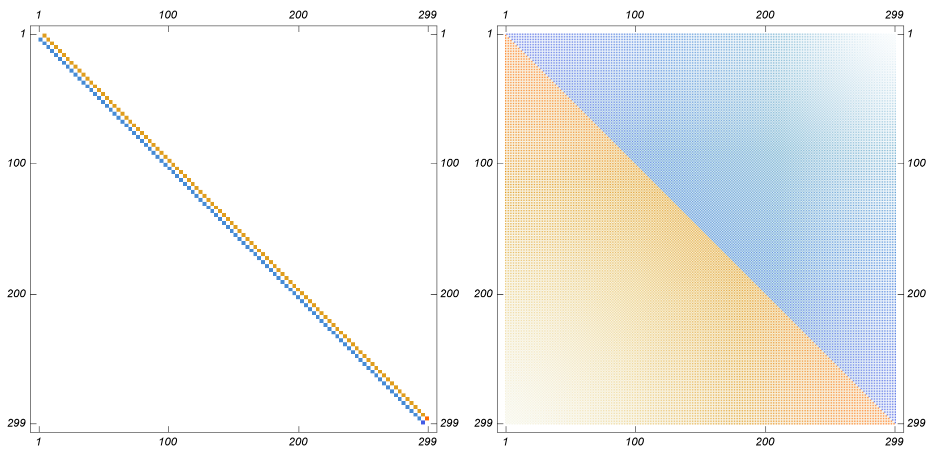
(**a**) (**b**)

Figure 5. The average MM for computing the Moore–Penrose inverse of complex matrices (**a**) $(n \times n)$ and (**b**) $(n \times n+20)$ by the methods.

Example 5 ([43])**.** *Consider tri-diagonal matrices, where the diagonals are as follows:*

$$(1,2) = 1, \quad (1,1) = 0, \quad (2,1) = -1. \tag{83}$$

The dimension of the matrices is an odd number, and the matrices are singular with $ind(A) = 1$. *The results for computing the Drazin inverse matrices for* $n = 109, 299, 499$ *are presented in Table 6 and Figure 6.*



(**a**) (**b**)

Figure 6. Representation of the (**a**) matrix and (**b**) Drazin inverse matrix for Example (5) with $n = 299$, respectively.

Table 6. Results of Example (5).

n	Method	NM	CH	TS	$E1$	$E2$	$E3$
109	MM	42	39	50	42	39	32
	CPU	1.2149	1.8952	1.3025	1.2235	1.1988	1.0102
299	MM	50	48	60	49	42	40
	CPU	3.1258	3.1181	4.102	3.1021	2.7022	2.5789
499	MM	54	54	65	56	42	40
	CPU	6.9082	7.012	8.1541	7.1421	5.81228	5.3552

7. Application

The proposed method can be used to compute the approximate inverse (i.e., the iterative algorithm (22)) when dealing with large sparse matrices arising from the discretization of linear partial differential equations (PDE) or FDE. Therefore, we consider the following PDE and FDE discussed previously in [44,45], using the iterative method (22). The computational performance of the suggested iterative method confirms the applicability and validity of the proposed strategy.

Example 6 ([46])**.** *Consider the fractional elliptic Poisson equation*

$$\begin{aligned} &\frac{\partial^\gamma \xi(x,y)}{\partial x^\gamma} + \frac{\partial^\gamma \xi(x,y)}{\partial y^\gamma} = g(x,y), \\ &\xi(0,y) = \phi_1(y), \quad \xi(1,y) = \phi_2(y), \\ &\xi(x,0) = \psi_1(x), \quad \xi(x,1) = \psi_2(x), \end{aligned} \tag{84}$$

where $0 \le x, y \le 1$*, and two cases:*

(a) $g(x,y) = \Gamma(\gamma+1)(x^\gamma + y^\gamma)$*, for* $0 < \gamma \le 2$ *and*

$$\phi_1(y) = \psi_1(x) = 0, \quad \phi_2(y) = y^\gamma, \;\; \psi_2(x) = x^\gamma, \tag{85}$$

(b) $g(x,y) = \sin(\pi x)\cos(\pi y)$*, for* $\gamma = 2$*, and*

$$\phi_1(y) = \phi_2(y) = \psi_1(x) = \psi_2(x) = 0, \tag{86}$$

where the fractional derivative $\frac{\partial^\gamma \xi(x,y)}{\partial y^\gamma}$ *of order* γ *is formulated in the Caputo sense. For solving Equation (84), we use the centre finite difference for* $\dfrac{\partial^\gamma \xi(x,y)}{\partial x^\gamma}$ *and* $\dfrac{\partial^\gamma \xi(x,y)}{\partial y^\gamma}$*. Therefore, for case (a) we have*

$$\frac{\partial^\gamma \xi(x,y)}{\partial x^\gamma}\Big|_{(x_i,y_j)} \approx \frac{\gamma!(\xi_{i+1,j} - \xi_{i-1,j})}{2h^\gamma}, \quad \frac{\partial^\gamma \xi(x,y)}{\partial y^\gamma}\Big|_{(x_i,y_j)} \approx \frac{\gamma!(\xi_{i,j+1} - \xi_{i,j-1})}{2k^\gamma}. \tag{87}$$

For case (b), we have

$$\frac{\partial^2 \xi(x,y)}{\partial x^2}\Big|_{(x_i,y_j)} \approx \frac{\xi_{i-1,j} - 2\xi_{i,j} + \xi_{i+1,j}}{h^2}, \quad \frac{\partial^2 \xi(x,y)}{\partial y^2}\Big|_{(x_i,y_j)} \approx \frac{\xi_{i,j-1} - 2\xi_{i,j} + \xi_{i,j+1}}{k^2}. \tag{88}$$

Furthermore, the values $h = \frac{1}{p}$ *and* $k = \frac{1}{q}$ *are adopted for the step size along the space x and y coordinates, respectively.*

The results of Example 6 are presented in Tables 7 and 8 and Figures 7 and 8.

Table 7. Results of Example (6) case (a).

γ, p	**Method**	NM	CH	TS	$E1$	$E2$	$E3$
1.2, 12	MM	30	33	40	35	30	28
	CPU	9.0082	10.1201	11.2511	10.1421	9.9122	7.1336
1.8, 17	MM	34	36	45	35	36	28
	CPU	11.0082	13.2589	19.9812	12.1421	11.0422	9.8336

Table 8. Results of Example (6) case (b).

Method	NM	CH	TS	$E1$	$E2$	$E3$
MM	42	42	55	42	39	32
CPU	19.9082	20.1589	31.2589	22.1421	16.0122	14.8336

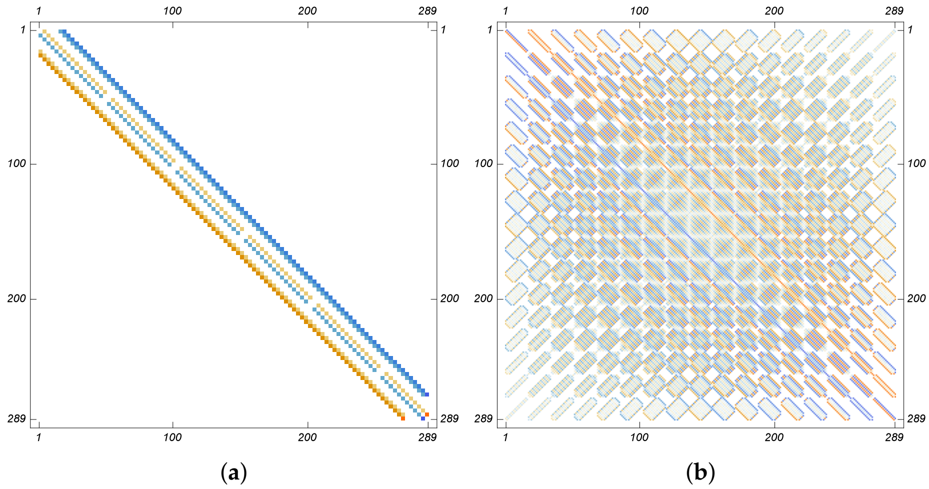


Figure 7. Representation of the (**a**) matrix and (**b**) inverse matrix for Example (6) case (**a**) when $p = q = 17$ and $\gamma = 1.8$.

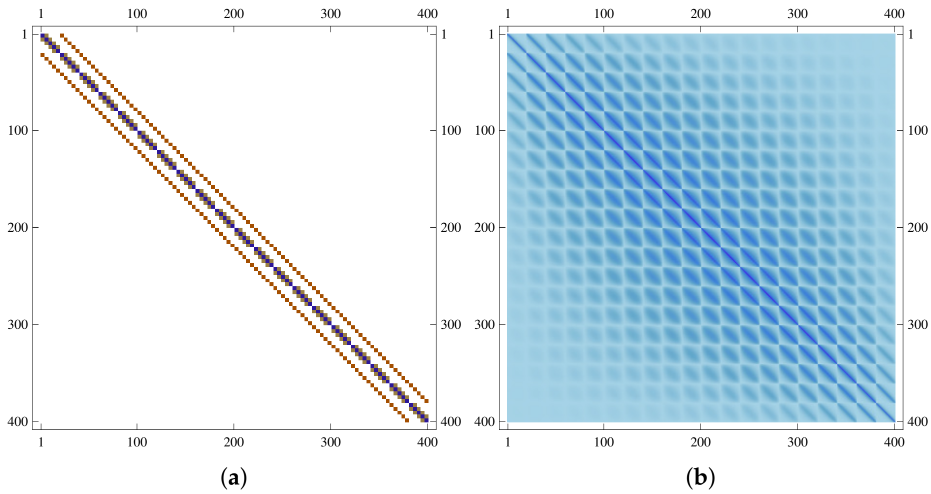


Figure 8. Representation of the (**a**) matrix and (**b**) inverse matrix for Example (6) (**b**) when $p = q = 20$.

Example 7 ([47]). *Consider the fractional sub-diffusion equation*

$$\begin{aligned}
&\frac{\partial^\gamma \xi(x,t)}{\partial t^\gamma} + \frac{\partial^2 \xi(x,t)}{\partial x^2} = f(x,t), \\
&\xi(0,t) = \xi(1,t) = 0, \quad 0 < t \le 1, \quad 0 < \gamma \le 1, \\
&\xi(x,0) = 0, \quad 0 < x < 1,
\end{aligned} \tag{89}$$

where the fractional derivative $\frac{\partial^\gamma \xi(x,t)}{\partial t^\gamma}$ *of order* γ *is formulated in the Caputo sense. According to [47], we use the finite difference for approximating the derivatives* $\dfrac{\partial^\gamma \xi(x,t)}{\partial t^\gamma}$ *and* $\dfrac{\partial^2 \xi(x,t)}{\partial x^2}$*, so that*

$$\frac{\partial^\gamma \xi(x,t)}{\partial t^\gamma}\bigg|_{(x_i,t_j)} \approx \frac{\gamma!(\xi_{i,j+1} - \xi_{i,j})}{k^\gamma}, \tag{90}$$

and

$$\frac{\gamma!(\xi_i^{j+1} - \xi_i^j)}{k^\gamma} = \theta\frac{\xi_{i+1}^{j+1} - 2\xi_i^{j+1} + \xi_{i-1}^{j+1}}{h^2} + (1-\theta)\frac{\xi_{i+1}^{j} - 2\xi_i^{j} + \xi_{i-1}^{j}}{h^2} + \theta f_i^{j+1} + (1-\theta) f_i^j, \tag{91}$$

where $\xi_i^j = \xi(x_i, t_j)$ *and* $f_i^j = f(x_i, t_j)$. $k = \frac{1}{p}$ *and* $h = \frac{1}{q}$ *are the step sizes along time t and space x, respectively. In this example, we examine two cases:*

(a) $f(x,t) = \dfrac{2}{\Gamma(3-\gamma)} t^{2-\gamma} \sin(2\pi x) + 4\pi^2 t^2 \sin(2\pi x)$,

(b) $f(x,t) = \dfrac{3}{4}\Gamma(\frac{1}{2}) t x^4 (x-1) - 4x^2(5x-3)^{\frac{3}{2}}$ *with non-smooth solution at* $t = 0$ *for* $\gamma = 0.5$.

The results of Example 7 are presented in Tables 9 and 10 and Figures 9 and 10.

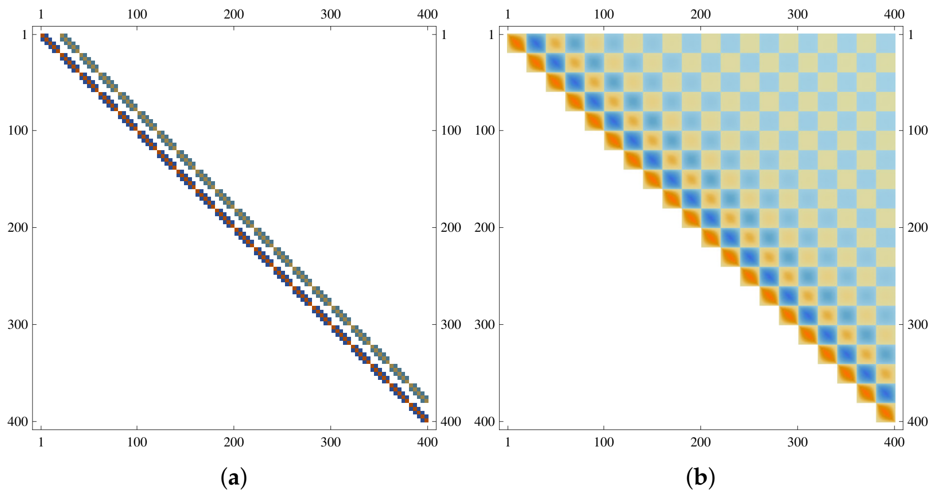


Figure 9. Representation of the (**a**) matrix and (**b**) inverse matrix for Example (7) case (**a**), when $p = q = 20$, respectively.

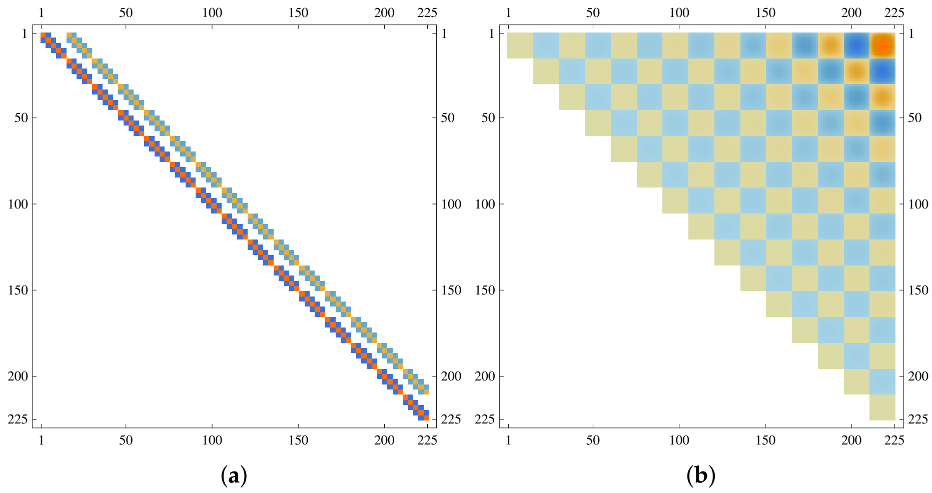


Figure 10. Representation of the (**a**) matrix and (**b**) inverse matrix for Example (7) case (**b**), when $p = q = 15$, respectively.

Table 9. Results of Example (7) (a) for $\gamma = \theta = 0.2$.

Method	NM	CH	TS	$E1$	$E2$	$E3$
MM	50	51	65	49	45	40
CPU	16.9082	17.1562	28.1256	16.1421	14.2122	13.0336

Table 10. Results of Example (7) case (b) for $\gamma = 0.5$ and $\theta = 0.3$.

Method	NM	CH	TS	$E1$	$E2$	$E3$
MM	100	75	95	91	75	68
CPU	21.9182	17.1158	20.1589	19.1821	17.2122	16.0036

As we know, the fractional-order derivatives in the partial differential equations are non-local. This means that the discretized matrix of approximating the spatial fractional-order derivatives should be dense and often Toeplitz-like [48–51]. In Examples 6 and 7, we adopted a very simple numerical approximation to the fractional operators, and the discretized matrices are sparse. In the follow-up, we give an example using a more elaborated approximation that leads to the dense matrix. The obtained results imply an elegant superiority of our proposed iterative scheme.

Example 8. *Consider the Riesz fractional diffusion equation [50]*

$$\begin{aligned} &\frac{\partial \xi(x,t)}{\partial t} = \kappa_\gamma \frac{\partial^\gamma \xi(x,t)}{\partial |x|^\gamma} + f(x,t), && (x,t) \in (0,1) \times (0,1], \ \kappa_\gamma > 0 \\ &\xi(x,0) = 15(1+\frac{\gamma}{4})x^3(1-x)^3, && x \in [0,1], \\ &\xi(0,t) = \xi(1,t) = 0, && t \in [0,1]. \end{aligned} \tag{92}$$

The Riesz fractional derivative $\frac{\partial^\gamma \xi(x,t)}{\partial |x|^\gamma}$ *is defined by [52]*

$$\begin{aligned}\frac{\partial^\gamma \xi(x,t)}{\partial |x|^\gamma} &= -\frac{1}{2\cos(\frac{\pi\gamma}{2})}\cdot\frac{1}{\Gamma(2-\gamma)}\cdot\frac{\partial^2}{\partial x^2}\int_a^b \frac{\xi(\zeta,t)}{|x-\zeta|^{\gamma-1}}d\zeta \\ &= -\frac{1}{2\cos(\frac{\pi\gamma}{2})}\Big({}_aD_x^\gamma \xi(x,t) + {}_xD_b^\gamma \xi(x,t)\Big), \quad \gamma \in (1,2),\end{aligned} \tag{93}$$

in which ${}_aD_x^\gamma$ *and* ${}_xD_b^\gamma$ *are (11) and (12) for* $m = 2$.

According to [50], the first-order time derivative at the point $t = t_j$ *is approximated by the second-order backward difference formula:*

$$\frac{\partial \xi(x,t)}{\partial t}\Big|_{(x_i,t_j)} = \begin{cases} \frac{\xi_i^{j+1} - \xi_i^j}{k}, & j = 1, \\ \frac{\xi_i^j - 4\xi_i^{j-1} + 3\xi_i^{j-2}}{2k}, & j \geq 2, \end{cases} \tag{94}$$

and also for any function $\xi(x) \in L^1(\mathbb{R})$, *we have*

$$\Delta_h^\gamma \xi(x) = -\frac{1}{h^\gamma}\sum_{l=-[\frac{b-x}{h}]}^{\frac{(x-a)}{h}} \omega_l^{(\gamma)} \xi(x - lh), \quad x \in \mathbb{R}, \tag{95}$$

where the γ*-dependent weight coefficient is defined as*

$$\omega_l^\gamma = \frac{(-1)^l \Gamma(1+\gamma)}{\Gamma(1+\frac{\gamma}{2}-l)\Gamma(1+\frac{\gamma}{2}+l)}, \quad l \in \mathbb{Z}. \tag{96}$$

Then, for a fixed h, the fractional centred difference operator in (95) holds:

$$\frac{\partial^\gamma \xi(x)}{\partial |x|^\gamma} = \Delta_h^\gamma \xi(x) + \mathcal{O}(h^2), \tag{97}$$

where $\kappa_\gamma \Delta_h^\gamma \xi_i^j$ *can be written into the matrix-vector product form* $\mathcal{A}\xi^j$ *as*

$$\mathcal{A} = -\kappa_\gamma T_x = -\frac{\kappa_\gamma}{h^\gamma}\begin{pmatrix} \omega_0^{(\gamma)} & \omega_{-1}^{(\gamma)} & \omega_{-2}^{(\gamma)} & \cdots & \omega_{3-N}^{(\gamma)} & \omega_{2-N}^{(\gamma)} \\ \omega_1^{(\gamma)} & \omega_0^{(\gamma)} & \omega_{-1}^{(\gamma)} & \cdots & \omega_{4-N}^{(\gamma)} & \omega_{3-N}^{(\gamma)} \\ \omega_2^{(\gamma)} & \omega_1^{(\gamma)} & \omega_0^{(\gamma)} & \cdots & \omega_{5-N}^{(\gamma)} & \omega_{4-N}^{(\gamma)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \omega_{N-3}^{(\gamma)} & \omega_{N-4}^{(\gamma)} & \omega_{N-5}^{(\gamma)} & \cdots & \omega_0^{(\gamma)} & \omega_{-1}^{(\gamma)} \\ \omega_{N-2}^{(\gamma)} & \omega_{N-3}^{(\gamma)} & \omega_{N-4}^{(\gamma)} & \cdots & \omega_1^{(\gamma)} & \omega_0^{(\gamma)} \end{pmatrix}. \tag{98}$$

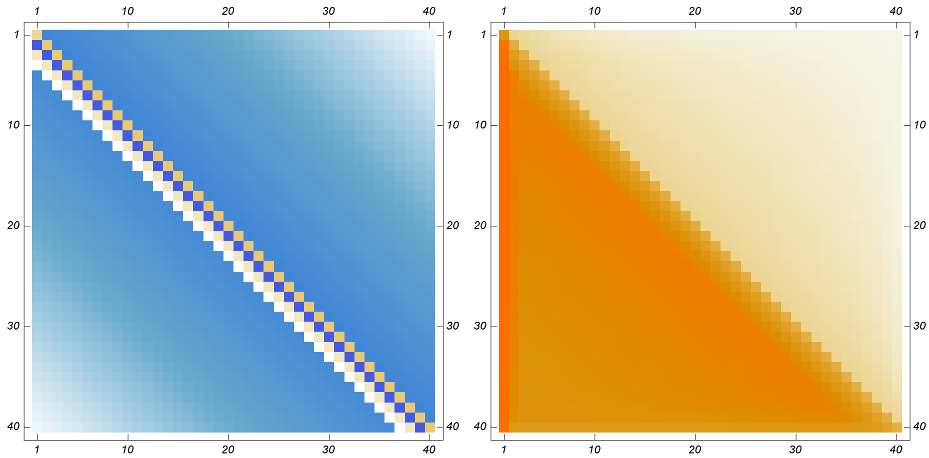
In [53], it was proven that T_x *is a symmetric positive definite Toeplitz matrix. The matrix-vector for solving the model problem (92) can be formulated as follows:*

$$\begin{aligned}&\frac{\xi^1 - \xi^0}{k} - \mathcal{A}\xi^1 = f^1, \\ &\frac{3\xi^j - 4\xi^{j-1} + \xi^{j-2}}{2k} - \mathcal{A}u^j = f^j, \quad 1 \leq j \leq N_t.\end{aligned} \tag{99}$$

The results of Example 8 are presented in Table 11 and Figure 11.

Table 11. Results of Example (8).

γ, p, q	Method	NM	CH	TS	$E1$	$E2$	$E3$
1.3, 10,10	MM	30	30	40	35	30	28
	CPU	0.0030	0.0031	0.0043	0.0037	0.0032	0.0027
1.5, 20, 20	MM	34	33	45	35	30	28
	CPU	0.0047	0.0045	0.0063	0.0049	0.0042	0.0036
1.8, 40, 40	MM	38	39	40	42	33	32
	CPU	0.0214	0.0221	0.0235	0.0251	0.0201	0.0185



(**a**) (**b**)

Figure 11. Representation of the (**a**) matrix and (**b**) inverse matrix for Example (8) when $p = q = 40$, respectively.

Finally, as seen from Tables 3–11 and Figures 4 and 5, we can find that the proposed method is faster than other numerical methods in terms of the number of matrix-matrix products and the elapsed CPU time.

8. Concluding Remarks

The inverse matrix calculation poses computational challenges in the solution of many problems due to its high computational cost. Therefore, avoiding a direct calculation and using efficient iterative methods are key aspects in mathematical modelling. In this paper, we presented a novel iterative method for solving nonlinear equations. The algorithm has good performance in terms of computational efficiency in the calculation of the Moore–Penrose and Drazin inverses. The key performance aspects of the method can be outlined as:

- Exhibits adequate results for specific real and complex matrices;
- Provides optimal results for real and complex square and rectangular random matrices of different dimensions;
- Shows a good feasibility for different dimensions when computing the Drazin inverse;
- The solution of the fractional elliptic Poisson equation shows superior results to other schemes;
- Yields good results for the solution of the fractional sub-diffusion equation for smooth and non-smooth solutions.

In synthesis, the results show that the theoretical findings are in accordance with numerical experiments, and we verified that the proposed algorithm is superior to others available in the literature. Finally, we point out that this new strategy has its own limitations and should be generalized and verified for more complicated linear and nonlinear problems.

In other words, the present paper is only an introduction to the topic, and there remains much work to do.

Author Contributions: Conceptualization, K.S.; Formal analysis, K.S., A.P. and R.E.; Investigation, A.P., J.A.T.M. and R.E.; Methodology, K.S., A.P. and R.E.; Validation, J.A.T.M. and R.E.; Visualization, K.S.; Writing—original draft, K.S., A.P. and R.E.; Writing–review—diting, J.A.T.M. All authors contributed equally to this paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Ph.D. thesis dissertations are public documents, but at the moment are not internet-accessible. If one is interested in revising the investigation, information can be requested from the authors.

Acknowledgments: The authors are thankful to the respected reviewers for their valuable comments and constructive suggestions towards the improvement of the original paper.

Conflicts of Interest: The authors declare that there is no conflict of interest regarding the publication of this article.

References

1. Penrose, R. A generalized inverse for matrices. *Pro. Camb. Philos. Soc.* **1955**, *51*, 406–413. [CrossRef]
2. Kelley, C.T. *Solving Nonlinear Equations with Newton's Method*; SIAM: Philadelphia, PA, USA, 2003.
3. Pan, V.Y. *Newton's Iteration for Matrix Inversion, Advances and Extensions, Matrix Methods: Theory Algorithms and Applications*; World Scientific: Singapore, 2010.
4. Li, H.B.; Huang, T.Z.; Zhang, Y.; Liu, X.P.; Gu, T.X. Chebyshev-type methods and preconditioning techniques. *Appl. Math. Comput.* **2011**, *218*, 260–270. [CrossRef]
5. Toutounian, F.; Soleymani, F. An iterative method for computing the approximate inverse of a square matrix and the Moore-Penrose inverse of a non-square matrix. *Appl. Math. Comput.* **2013**, *224*, 671–680. [CrossRef]
6. Pan, V.Y.; Soleymani, F.; Zhao, L. Highly efficient computation of generalized inverse of a matrix. *Appl. Math. Comput.* **2018**, *316*, 89–101.
7. Esmaeili, H.; Pirnia, A. An efficient quadratically convergent iterative method to find the Moore-Penrose inverse. *Int. J. Comput. Math.* **2017**, *94*, 1079–1088. [CrossRef]
8. Pan, V.Y.; Schreiber, R. An improved Newton iteration for the generalized inverse of a matrix with applications. *SIAM J. Sci. Stat. Comput.* **1991**, *12*, 1109–1131. [CrossRef]
9. Drazin, M.P. Pseudoinverses in associative rings and semigroups. *Am. Math. Mon.* **1958**, *65*, 506–514. [CrossRef]
10. Wilkinson, J.H. Note on the practical significance of the Drazin inverse. In *Recent Applications of Generalized Inverses*; Campbell, S.L., Ed.; Research Notes in Mathematics; Pitman Advanced Publishing Program: Boston, MA, USA, 1982; pp. 82–99.
11. Liu, X.; Cai, N. High-order iterative methods for the DMP inverse. *J. Math.* **2018**, *2018*, 8175935. [CrossRef]
12. Mosic, D.; Djordjevic, D.S. Block representations of the generalized Drazin inverse. *Appl. Math. Comput.* **2018**, *331*, 200–209.
13. Qiao, S.; Wie, Y. Acute perturbation of Drazin inverse and oblique projectors. *Math. China* **2018**, *13*, 1427–1445. [CrossRef]
14. Wang, X.Z.; Ma, H.; Stanimirovic, P.S. Recurrent neural network for computing the W-weighted Drazin inverse. *Appl. Math. Comput.* **2017**, *300*, 1–20. [CrossRef]
15. Duarte, F.B.M.; Machado, J.T. Chaotic phenomena and fractional-order dynamics in the trajectory control of redundent manipulators. *Nonlinear Dyn.* **2002**, *29*, 315–342. [CrossRef]
16. Ferreira, N.M.F.; Duarte, F.B.M.; Lima, M.F.M.; Marcos, M.G.; Machado, J.T. Application of fractional calculus in the dynamical analysis ans control of mechanical manipulators. *Fract. Calc. Appl. Anal.* **2008**, *11*, 91–113.
17. Caputo, M. Linear models of dissipation whose q is almost frequency independent-ii. *Geophys. J. R. Astron. Soc.* **1967**, *13*, 529–539. [CrossRef]
18. Kiryakova, V. *Generalized Fractional Calculus and Applications*; John Wiley and Sons, Inc.: New York, NY, USA, 1993.
19. Machado, J.T.; Kiryakova, V. The chronicles of fractional calculus. *Fract. Calc. Appl. Anal.* **2017**, *20*, 307–336. [CrossRef]
20. Machado, J.T.; Kiryakova, V.; Mainard, F. Recent history of fractional calculus. *Commun. Nonlinear Sci. Numer. Simul.* **2011**, *16*, 1140–1153. [CrossRef]
21. Machado, J.T.; Mainardi, F.; Kiryakova, V. Fractional calculus: Quo vadimus? (where are we going?). *Fract. Calc. Appl. Anal.* **2015**, *18*, 495–526. [CrossRef]

22. Machado, J.T.; Mainardi, F.; Kiryakova, V.; Atanacković, T. Fractional calculus: D'oú venons-nous? Que sommes-nous? Oú allons-nous? (Contributions to Round Table Discussion held at ICFDA 2016). *Fract. Calc. Appl. Anal.* **2016**, *19*, 1074–1104. [CrossRef]
23. Xu, B.; Chen, D.; Zhang, H.; Zhou, R. Dynamic analysis and modeling of a novel fractional-order hydro-turbine-generator unit. *Nonlinear Dyn.* **2015**, *81*, 1263–1274. [CrossRef]
24. Xu, B.; Chen, D.; Zhang, H.; Wang, F. The modeling of the fractional-order shafting system for a water jet mixed-flow pump during the startup process. *Commun. Nonlinear Sci. Numer. Simul.* **2015**, *29*, 12–24. [CrossRef]
25. Dehghan, M.; Hajarian, M. Some derivative free quadratic and cubic convergence iterative formulas for solving nonlinear equations. *Comput. Appl. Math.* **2010**, *29*, 19–30. [CrossRef]
26. Dehghan, M.; Hajarian, M. New iterative method for solving nonlinear equations with fourth-order convergence. *Int. J. Comput. Math.* **2010**, *87*, 834–839. [CrossRef]
27. Erfanifar, R.; Sayevand, K.; Esmaeili, H. On modified two-step iterative method in the fractional sense: Some applications in real world phenomena. *Int. J. Comput. Math.* **2020**, *97*, 2109–2141. [CrossRef]
28. Sayevand, K.; Erfanifar, R.; Esmaeilli, H. On computational efficiency and dynamical analysis for a class of novel multi-step iterative schemes. *Int. J. Appl. Comput. Math.* **2020**, *6*, 1–23. [CrossRef]
29. Li, W.; Li, Z. A family of iterative methods for computing the approximate inverse of a square matrix and inner inverse of a non-square matrix. *Appl. Math. Comput.* **2010**, *215*, 3433–3442. [CrossRef]
30. Wu, X. A note on computational algorithm for the inverse of a square matrix. *Appl. Math. Comput.* **2007**, *187*, 962–964. [CrossRef]
31. Chen, H.; Wang, Y. A Family of higher-order convergent iterative methods for computing the Moore Penrose inverse. *Appl. Math. Comput.* **2011**, *218*, 4012–4016. [CrossRef]
32. Soleymani, F.; Stanimirovic, P.S. A note on the stability of a p-th order iteration for finding generalized inverses. *Appl. Math. Lett.* **2014**, *28*, 77–81. [CrossRef]
33. Horn, R.A.; Johnson, C.R. *Matrix Analysis*; Cambridge University Press: Cambridge, UK; New York, NY, USA; New Rochelle, NY, USA; Melbourne, Australia; Sydney, Australia, 1986.
34. Stanimirovic, P.S.; Cvetkovic-Ilic, D.S. Successive matrix squaring algorithm for computing outer inverses. *Appl. Math. Comput.* **2008**, *203*, 19–29. [CrossRef]
35. King, C. A Note on Drazin Inverses. *Pac. J. Math.* **1977**, *70*, 383–390. [CrossRef]
36. Campbell, S.L. *Singular Systems of Differential Equations; Research Notes in Mathematics*; Pitman Advanced Publishing Program: Boston, MA, USA, 1980.
37. Ren, D.G. *Analysis and Design of Descriptor Linear Systems*; Springer: New York, NY, USA, 2010.
38. Kaczorek, T.; Borawski, K. *Descriptor Systems of Integer and Fractional Orders*; Studies in Systems, Decision and Control; Springer: Cham, Switzerland, 2021.
39. Li, X.; Wei, Y. Iterative methods for the Drazin inverse of a matrix with a complex spectrum. *Appl. Math. Comput.* **2004**, *147*, 855–862. [CrossRef]
40. Wang, G.; Wei, Y.; Qiao, S. *Generalized Inverses: Theory and Computations*; Science Press: New York, NY, USA, 2004.
41. Ben-Israel, A.; Greville, T.N.E. *Generalized Inverses: Theory and Applications*, 2nd ed.; Springer: New York, NY, USA, 2003.
42. Gill, P.R.; Murray, W.; Wright, M.H. *Numerical Linear Algebra and Optimization—Volume 1*; Addison-Wesley: Redwood City, CA, USA, 1991.
43. Toutounian, F.; Buzhabadi, R. New methods for computing the Drazin-inverse solution of singular linear systems. *Appl. Math. Comput.* **2017**, *294*, 343–352. [CrossRef]
44. Samko, S.G.; Kilbas, A.; Marichev, O. *Fractional Integrals and Derivatives: Theory and Applications*; Gordon and Breach: Switzerland, 1993.
45. Sayevand, K.; Machado, J.T.; Moradi, V. A new non-standard finite difference method for analyzing the fractional Navier–Stokes equations. *Comput. Math. Appl.* **2019**, *78*, 1681–1694. [CrossRef]
46. Youssef, I.K.; Dewaik, M.H.E. Solving Poisson's equations with fractional order using Haar wavelet. *Appl. Math. Nonlinear Sci.* **2017**, *2*, 271–284. [CrossRef]
47. Erfanifar, R.; Sayevand, K.; Ghanbari, N.; Esmaeili, H. A modified Chebyshev ϑ-weighted Crank-Nicolson method for analyzing fractional sub-diffusion equations. *Numer. Methods Partial Differ. Equ.* **2020**, *13*, 1–13. [CrossRef]
48. Mockary, S.; Babolian, E.; Vahidi, A.R. A fast numerical method for fractional partial differential equations. *Adv. Differ. Equ.* **2019**. [CrossRef]
49. Gu, X.M.; Huang, T.Z.; Ji, C.C.; Carpentieri, B.; Alikhanov, A.A. Fast iterative method with a second-order implicit difference scheme for time-space fractional convection-diffusion equation. *J. Sci. Comput.* **2017**, *72*, 957–985. [CrossRef]
50. Gu, X.M.; Zhao, Y.L.; Zhao, X.L.; Carpentieri, B.; Huang, Y.Y. A note on parallel preconditioning for the all-at-once solution of Riesz fractional diffusion equations. *Numer. Math. Theory Meth. Appl.* **2021**, *14*, 893–919.
51. Gu, X.M.; Wu, S.L. A parallel-in-time iterative algorithm for Volterra partial integro-differential problems with weakly singular kernel. *J. Comput. Phys.* **2020**, *417*, 109576. [CrossRef]

52. Yang, Q.; Liu, F.; Turner, I. Numerical methods for fractional partial differential equations with Riesz space fractional derivatives. *Appl. Math. Model.* **2010**, *34*, 200–218. [CrossRef]
53. Elik, C.C.; Duman, M. Crank-Nicolson method for the fractional diffusion equation with the Riesz fractional derivative. *J. Comput. Phys.* **2012**, *231*, 1743–1750.