mathematics

MDPI

Article

A Robust Hermitian and Skew-Hermitian Based Multiplicative Splitting Iterative Method for the Continuous Sylvester Equation

Mohammad Khorsand Zak [1,*] and Abbas Abbaszadeh Shahri [2]

[1] Department of Applied Mathematics, Aligudarz Branch, Islamic Azad University, Aligudarz P.O. Box 159, Iran

[2] Faculty of Engineering and Technology, Bircham International University, P.O. Box 2233 Madrid, Spain; shahri.abbas@bircham.edu

* Correspondence: khorsand@iau.ac.ir

Abstract: For solving the continuous Sylvester equation, a class of Hermitian and skew-Hermitian based multiplicative splitting iteration methods is presented. We consider two symmetric positive definite splittings for each coefficient matrix of the continuous Sylvester equations, and it can be equivalently written as two multiplicative splitting matrix equations. When both coefficient matrices in the continuous Sylvester equation are (non-symmetric) positive semi-definite, and at least one of them is positive definite, we can choose Hermitian and skew-Hermitian (HS) splittings of matrices A and B in the first equation, and the splitting of the Jacobi iterations for matrices A and B in the second equation in the multiplicative splitting iteration method. Convergence conditions of this method are studied in depth, and numerical experiments show the efficiency of this method. Moreover, by numerical computation, we show that multiplicative splitting can be used as a splitting preconditioner and induce accurate, robust and effective preconditioned Krylov subspace iteration methods for solving the continuous Sylvester equation.

Keywords: Sylvester equation; matrix equation; multiplicative splitting; Hermitian and skew-Hermitian splitting; iterative methods

MSC: 15A24; 15A30; 15A69; 65F10; 65F30

check for updates

Academic Editor: Patricia J. Y. Wong

Received: 9 December 2024
Revised: 31 December 2024
Accepted: 7 January 2025
Published: 20 January 2025

Citation: Khorsand Zak, M.; Abbaszadeh Shahri, A. A Robust Hermitian and Skew-Hermitian Based Multiplicative Splitting Iterative Method for the Continuous Sylvester Equation. *Mathematics* **2025**, *13*, 318. https://doi.org/10.3390/math13020318

1. Introduction

Matrix equations arise in a number of problems of scientific computations and engineering applications, such as in control theory [1,2], model reduction [3], signal processing [4] and image processing [5], and many researchers focus on the matrix equations [6–11]. Nowadays, the continuous Sylvester equation is possibly the most famous and the most broadly employed linear matrix equation [5,6,11–19]. It is given as

$$AX + XB = C, \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{m\times m}$ and $C \in \mathbb{R}^{n\times m}$ are defined matrices and $X \in \mathbb{R}^{n\times m}$ is an unknown matrix. A Lyapunov equation is a special case with $m = n, B = A^T$ and $C = C^T$. Here and in the subsequent sections, W^T is used to denote the transpose of the matrix $W \in \mathbb{R}^{n\times n}$. The continuous Sylvester Equation (1) has a unique solution if and only if A and $-B$ have no common eigenvalues, which will be assumed throughout this paper.

The continuous Sylvester equation has been studied extensively in recent decades as a key pipeline and has received wide applications in areas such as control theory, differential systems and signal processing. As a result, a number of algorithms focusing on the dimension reduction of the equation have been proposed [20]. The resulting algorithms can be extended to more general forms of the continuous algebraic Lyapunov equations and extended Sylvester equations, as well as to the Markov jump system. For the continuous Sylvester equation, various iterative methods are given in the literature. For instance, splitting the matrix equation to find an approximate solution of the continuous Sylvester equation has shown proper convergence, and is simpler than the corresponding multiplicative splitting and thusg has relatively lower computational complexity [20–24]. Another class of solutions focuses on overcoming the lack of stability via the combination of two split equations and corresponding new iterative methods to approximate the solution of the continuous Sylvester equation via the modified Richardson technique [23,25,26].

As a summary, if the equation can be expressed as a simple, symmetric, regular singular equation, it can exploit the mixed structure to solve the equation by finding a specific transformation to the standard form or degenerative decomposition for large-scale experiments. Toward a solution, different techniques like regular deflating linearization, the Hamiltonian boundary value method, dealing with fundamental subspaces which do not relate to eigenvalues of the M-matrix, utilizing standard polynomial eigenproblem algorithms without computation of any special splitting and implementing the Hessenberg matrix to convert the homogeneous equivalent matrix to a special block structure have been reported in the literature [25,27–29].

In general, the dimensions of A and B may be orders of magnitude different, and this fact is key in selecting the most appropriate numerical solution strategy [11]. For solving general Sylvester equations of small size, we use some methods classified such as direct methods. Among these direct methods are the Bartels–Stewart [30] and the Hessenberg–Schur [31] methods, which consist of transforming coefficient matrices A and B into triangular or Hessenberg form by an orthogonal similarity transformation and then solving the resulting system directly by a back-substitution process. When the coefficient matrices A and B are large and sparse, iterative methods are often the methods of choice for solving the Sylvester equation (1) efficiently and accurately. Many iterative methods have been developed for solving matrix equations, such as the alternating direction implicit (ADI) method [32], the Krylov subspace-based algorithms [13,33,34], the Hermitian and skew-Hermitian splitting (HSS) method, the inexact variant of HSS (IHSS) iteration method [6] and the nested splitting conjugate gradient (NSCG) method [17,35] and the nested splitting CGNR (NS-CGNR) method [18].

In order to study the numerical methods, we often rewrite the continuous Sylvester equation (1) as th following linear system of equations

$$\mathcal{A}x = c, \tag{2}$$

where the matrix $\mathcal{A}$ is of dimension $nm \times nm$ and is given by

$$\mathcal{A} = I_m \otimes A + B^T \otimes I_n, \tag{3}$$

where $\otimes$ denotes the Kronecker product $(A \otimes B = [a_{ij}B])$ and

$$\begin{aligned} c &= vec(C) = (c_{11}, c_{21}, \cdots, c_{n1}, c_{12}, c_{22}, \cdots, c_{n2}, \cdots, c_{nm})^T \\ x &= vec(X) = (x_{11}, x_{21}, \cdots, x_{n1}, x_{12}, x_{22}, \cdots, x_{n2}, \cdots, x_{nm})^T. \end{aligned}$$

Of course, this is quite expensive and a numerically poor way to determine the solution X of the continuous Sylvester equation (1), as the linear system of Equation (2) is costly to solve and can be ill-conditioned.

Now, we recall some necessary notations and useful results, which will be used in the following section. In this paper, we use $\lambda(M), ||M||_2$, $||M||_F$ and I_n to denote the eigenvalue, the spectral norm, the Frobenius norm of a matrix $M \in \mathbb{R}^{n\times n}$ and the identity matrix with dimension n, respectively. Note that $||.||_2$ is also used to represent the 2-norm of a vector. For nonsingular matrix $\mathcal{B}$, we denote by $\kappa(\mathcal{B}) = ||\mathcal{B}||_2||\mathcal{B}^{-1}||_2$ its spectral condition number, and for a symmetric a positive definite matrix $\mathcal{B}$, we define the $||\cdot||_{\mathcal{B}}$ norm of a vector $x \in \mathbb{R}^n$ as $||x||_{\mathcal{B}} = \sqrt{x^H\mathcal{B}x}$. Then, the induced $||\cdot||_{\mathcal{B}}$ norm of a matrix $M \in \mathbb{R}^{n\times n}$ is defined as $||M||_{\mathcal{B}} = ||\mathcal{B}^{\frac{1}{2}}M\mathcal{B}^{-\frac{1}{2}}||_2$. In addition, it holds that $||Mx||_{\mathcal{B}} \leq ||M||_{\mathcal{B}}||x||_{\mathcal{B}}$, $||M||_{\mathcal{B}} \leq \sqrt{\kappa(\mathcal{B})}||M||_2$ and $||I||_{\mathcal{B}} = 1$, where I is the identity matrix. For any matrices $A = [a_{ij}]$ and $B = [b_{ij}]$, $A \otimes B$ denotes the Kronecker product, defined as $A \otimes B = [a_{ij}B]$. For the matrix $X = (x_1, x_2, \cdots, x_m) \in \mathbb{R}^{n\times m}$, $vec(X)$ denotes the *vec* operator, defined as $vec(X) = (x_1^T, x_2^T, \cdots, x_m^T)^T$. Moreover, for a matrix $M \in \mathbb{R}^{n\times n}$ and the vector $vec(M) \in \mathbb{R}^{nm}$, we have $||M||_F = ||vec(M)||_2$.

For matrix $\mathcal{A} \in \mathbb{R}^{n\times n}$, $\mathcal{A} = \mathcal{B} - \mathcal{C}$ is called a splitting of the matrix $\mathcal{A}$ if $\mathcal{B}$ is nonsingular. This splitting is a convergent splitting if $\rho(\mathcal{B}^{-1}\mathcal{C}) < 1$ and a contractive splitting if $||\mathcal{B}^{-1}\mathcal{C}|| < 1$ for some matrix norm.

The reminder of this paper is organized as follows. Section 2 presents our main contribution. In other words, the multiplicative splitting iteration (MSI) method for the continuous Sylvester equation and its convergence properties are studied deeply. Section 3 is devoted to an extensive numerical experiments with full comparison with other state-of-the-art methods in the literature. In Section 4, we address some challenges and suggestions for future work. Finally, we present our conclusions in Section 5.

2. Multiplicative Splitting Iterations

2.1. Traditional MSI Method

Consider the linear system of Equation (2). Let $\mathcal{A} = \mathcal{M}_i - \mathcal{N}_i$ $(i = 1, 2)$ be two splittings of the coefficient matrix $\mathcal{A}$. The MSI method for solving the system of linear Equation (2) is defined as follows [36]:

MSI method for linear system of equations:

```
Given an initial guess $x^{(0)} \in \mathbb{R}^n$,
For $k = 1, 2, \cdots$ until convergence, do
    $u^{(k+1)} = \mathcal{M}_1^{-1}\mathcal{N}_1 x^{(k)} + \mathcal{M}_1^{-1}c$
    $x^{(k+1)} = \mathcal{M}_2^{-1}\mathcal{N}_2 u^{(k+1)} + \mathcal{M}_2^{-1}c$
end
```

The MSI method can be equivalently written in the form

$$x^{(k+1)} = \mathcal{T}_{msi}x^{(k)} + \mathcal{G}_{msi}c, \qquad k = 0, 1, 2, \cdots$$

where $\mathcal{T}_{msi} = \mathcal{M}_2^{-1}\mathcal{N}_2\mathcal{M}_1^{-1}\mathcal{N}_1$ and $\mathcal{G}_{msi} = \mathcal{M}_2^{-1}\mathcal{N}_2\mathcal{M}_1^{-1} + \mathcal{M}_2^{-1}$. See [36] for more details.

2.2. MSI Method for the Sylvester Equation

Based on the MSI method proposed in [36], we obtain the MSI method for the continuous Sylvester equation. Let $A = M_i - N_i$ and $B = P_i - Q_i$, $(i = 1, 2)$ be two splittings of the matrices A and B, such that M_i and P_i, $(i = 1, 2)$ are symmetric positive definite. The continuous Sylvester equation (1) can be equivalently written as the multiplicative splitting matrix equations

$$\begin{cases} M_1U + UP_1 = N_1X + XQ_1 + C \\ M_2X + XP_2 = N_2U + UQ_2 + C \end{cases}$$

Under the assumption that M_i and P_i, $(i = 1, 2)$ are symmetric positive definite, we easily know that there is no common eigenvalues between the matrices M_i and $-P_i$, $(i = 1, 2)$, so that this two multiplicative splitting matrix equations have unique solutions for all given right-hand-side matrices.

Now, based on the above observations, we can establish the following multiplicative splitting iterations for solving the continuous Sylvester Equation (1):

MSI method for Sylvester equation:

```
Given an initial guess $X^{(0)} \in \mathbb{R}^{m\times n}$,
For $k = 1, 2, \cdots$ until convergence, do
    Solve $M_1 U^{(k+1)} + U^{(k+1)} P_1 = N_1 X^{(k)} + X^{(k)} Q_1 + C$
    Solve $M_2 X^{(k+1)} + X^{(k+1)} P_2 = N_2 U^{(k+1)} + U^{(k+1)} Q_2 + C$
end
```

In special cases, when both coefficient matrices A and B in the Sylvester equation (1) are (non-symmetric) positive semi-definite, and at least one of them is positive definite, in the first equation in the MSI method we can choose Hermitian and skew-Hermitian (HS) splittings of matrices A and B, i.e., $A = H_A - S_A$ and $B = H_B - S_B$, where H_A, S_A, H_B, S_B are the Hermitian and skew-Hermitian parts of A and B, respectively. Also, in the second equation in the MSI method, we consider the splitting of the Jacobi iterations [37] for matrices A and B, i.e., $A = D_A - N_A$ and $B = D_B - N_B$, where D_A, N_A, D_B, N_B are the diagonal and non-diagonal parts of A and B, respectively. Therefore, we can rewrite this method as follows:

```
Given an initial guess $X^{(0)} \in \mathbb{R}^{m\times n}$,
For $k = 1, 2. \cdots$ until convergence, do
    Solve system $H_A U^{(k+1)} + U^{(k+1)} H_B = S_A X^{(k)} + X^{(k)} S_B + C$
    Solve system $D_A X^{(k+1)} + X^{(k+1)} D_B = N_A U^{(k+1)} + U^{(k+1)} N_B + C$
end
```

Achieving two Sylvester equations that we can easily solve is our motivation for choosing these splittings. This is because the first system can be solved by the Sylvester conjugate gradient method [38], and the following routine can be used for direct solution of the second system:

Directly solution of matrix equation $D_A X + X D_B = F$:

```
For $i = 1 : n$
    For $j = 1 : m$
        $x_{ij} = \frac{f_{ij}}{a_{ii} + b_{jj}}$
    end
end
```

where a_{ii} and b_{jj} are the diagonal elements of matrices A and B, respectively. Moreover, $F = N_A U + U N_B + C$, and U is the solution of the first equation in the MSI method for Sylvester equation.

2.3. Using Multiplicative Splitting as a Preconditioner

Given the fact that any matrix splitting can naturally induce a splitting preconditioner for the Krylov subspace methods (see [39]) in Section 3, by numerical computation, we show that multiplicative splitting can be used as a splitting preconditioner to induce accurate, robust and effective preconditioned Krylov subspace iteration methods for solving the continuous Sylvester equation.

2.4. Convergence Analysis

In the subsequent solution, we need the following lemmas.

Lemma 1 ([36])**.** *Let $B, C \in \mathbb{R}^{n\times n}$ be two Hermitian matrices. Then, $BC = CB$ if and only if B and C have a common set of orthonormal eigenvectors.*

Lemma 2 ([40])**.** *Let $\mathcal{A} \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix. Then, for all $x \in \mathbb{R}^n$, we have $||\mathcal{A}^{\frac{1}{2}}x||_2 = ||x||_{\mathcal{A}}$ and*

$$\sqrt{\lambda_{\min}(\mathcal{A})}||x||_{\mathcal{A}} \leq ||\mathcal{A}x||_2 \leq \sqrt{\lambda_{\max}(\mathcal{A})}||x||_{\mathcal{A}}.$$

Lemma 3 ([41])**.** *Suppose that $A, B \in \mathbb{R}^{n\times n}$ are two Hermitian matrices, and denote the minimum and the maximum eigenvalues of a matrix M with $\lambda_{\min}(M)$ and $\lambda_{\max}(M)$, respectively. Then,*

$$\begin{aligned}\lambda_{\max}(A+B) &\leq \lambda_{\max}(A) + \lambda_{\max}(B),\\ \lambda_{\min}(A+B) &\geq \lambda_{\min}(A) + \lambda_{\min}(B).\end{aligned}$$

Lemma 4 ([41])**.** *Let $A, B \in \mathbb{C}^{n\times n}$, and λ and μ be the eigenvalues of A and B, and x and y be the corresponding eigenvectors, respectively. Then $\lambda\mu$ is an eigenvalue of $A \otimes B$ corresponding to the eigenvector $x \otimes y$.*

Lemma 5. *Suppose that $\mathcal{A} = \mathcal{M} - \mathcal{N}$ is a splitting such that $\mathcal{M}$ is symmetric positive definite, with $\mathcal{M} = I_m \otimes M + P^T \otimes I_n$ and $\mathcal{N} = I_m \otimes N + Q^T \otimes I_n$. If*

$$\theta^3 \frac{\max|\lambda(N)| + \max|\lambda(Q)|}{\lambda_{\min}(M) + \lambda_{\min}(P)} < 1,$$

where $\theta = \sqrt{\frac{\lambda_{\max}(M)+\lambda_{\max}(P)}{\lambda_{\min}(M)+\lambda_{\min}(P)}}$, then $||\mathcal{M}^{-1}\mathcal{N}||_{\mathcal{M}} < 1$.

Proof. By Lemmas 3 and 4, we have

$$||\mathcal{M}||_2 = \lambda_{\max}(\mathcal{M}) \geq \lambda_{\min}(\mathcal{M}) \geq \lambda_{\min}(M) + \lambda_{\min}(P),$$

and

$$||\mathcal{N}||_2 = \max_{\lambda\in\Lambda(\mathcal{N})} |\lambda(\mathcal{N})| \leq \max|\lambda(N)| + \max|\lambda(Q)|.$$

Therefore, it follows that

$$\begin{aligned}||\mathcal{M}^{-1}\mathcal{N}||_{\mathcal{M}} &\leq \sqrt{\kappa(\mathcal{M})}||\mathcal{M}^{-1}\mathcal{N}||_2\\ &\leq \sqrt{\kappa(\mathcal{M})}||\mathcal{M}^{-1}||_2||\mathcal{N}||_2\\ &\leq (\kappa(\mathcal{M}))^{\frac{3}{2}}\frac{||\mathcal{N}||_2}{||\mathcal{M}||_2}\\ &\leq (\kappa(\mathcal{M}))^{\frac{3}{2}}\frac{\max|\lambda(N)|+\max|\lambda(Q)|}{\lambda_{\min}(M)+\lambda_{\min}(P)}.\end{aligned}$$

Again, the use of Lemmas 3 and 4 implies that

$$\sqrt{\kappa(\mathcal{M})} = \sqrt{\frac{\lambda_{\max}(\mathcal{M})}{\lambda_{\min}(\mathcal{M})}} \leq \sqrt{\frac{\lambda_{\max}(M)+\lambda_{\max}(P)}{\lambda_{\min}(M)+\lambda_{\min}(P)}} = \theta. \tag{4}$$

So, we can write

$$||\mathcal{M}^{-1}\mathcal{N}||_{\mathcal{M}} \leq \theta^3\frac{\max|\lambda(N)|+\max|\lambda(Q)|}{\lambda_{\min}(M)+\lambda_{\min}(P)}. \tag{5}$$

This clearly proves the lemma. □

Theorem 1. *Let $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{m\times m}$ and consider two splittings $A = M_i - N_i$ and $B = P_i - Q_i$ $(i = 1,2)$ such that M_i and P_i, $(i = 1,2)$ are symmetric positive definite. Denote by $\mathcal{A} = \mathcal{M}_i - \mathcal{N}_i$ $(i = 1,2)$ with $\mathcal{M}_i = I_m \otimes M_i + P_i^T \otimes I_n$ and $\mathcal{N}_i = I_m \otimes N_i + Q_i^T \otimes I_n$ $(i = 1,2)$, and assume that $\mathcal{M}_1\mathcal{A}^{-1}$ and $\mathcal{M}_2\mathcal{A}^{-1}$ are Hermitian matrices and $\mathcal{M}_1\mathcal{A}^{-1}\mathcal{M}_2 = \mathcal{M}_2\mathcal{A}^{-1}\mathcal{M}_1$. Then, the MSI method is convergent if $\varrho_1\varrho_2 < 1$, where*

$$\varrho_i = \theta_i^3 \frac{\max|\lambda(N_i)| + \max|\lambda(Q_i)|}{\lambda_{\min}(M_i) + \lambda_{\min}(P_i)}, \text{ and } \theta_i = \sqrt{\frac{\lambda_{\max}(M_i) + \lambda_{\max}(P_i)}{\lambda_{\min}(M_i) + \lambda_{\min}(P_i)}}, \ (i = 1,2).$$

Proof. By making use of the Kronecker product, we can rewrite the above-described MSI method in the following matrix–vector form:

$$\begin{cases} (I_m \otimes M_1 + P_1^T \otimes I_n)u^{(k+1)} = (I_m \otimes N_1 + Q_1^T \otimes I_n)x^{(k)} + c \\ (I_m \otimes M_2 + P_2^T \otimes I_n)x^{(k+1)} = (I_m \otimes N_2 + Q_2^T \otimes I_n)u^{(k+1)} + c \end{cases}$$

which can be arranged equivalently as

$$\begin{cases} \mathcal{M}_1 u^{(k+1)} = \mathcal{N}_1 x^{(k)} + c \\ \mathcal{M}_2 x^{(k+1)} = \mathcal{N}_2 u^{(k+1)} + c \end{cases}$$

which can be obtained by the following iteration method:

$$\begin{cases} u^{(k+1)} = \mathcal{M}_1^{-1}\mathcal{N}_1 x^{(k)} + \mathcal{M}_1^{-1}c \\ x^{(k+1)} = \mathcal{M}_2^{-1}\mathcal{N}_2 u^{(k+1)} + \mathcal{M}_2^{-1}c \end{cases} \tag{6}$$

Evidently, the above iteration scheme is the MSI method [36] for solving the system of linear Equation (2) with $\mathcal{A} = \mathcal{M}_i - \mathcal{N}_i$ $(i = 1,2)$. The MSI iteration (6) can be neatly expressed as a stationary fixed-point iteration as follows:

$$x^{(k+1)} = \mathcal{T}x^{(k)} + \mathcal{G}c$$

with $\mathcal{T} = \mathcal{M}_2^{-1}\mathcal{N}_2\mathcal{M}_1^{-1}\mathcal{N}_1$ and $\mathcal{G} = \mathcal{M}_2^{-1}\mathcal{N}_2\mathcal{M}_1^{-1} + \mathcal{M}_2^{-1}$.

Because $\mathcal{M}_1\mathcal{A}^{-1}\mathcal{M}_2 = \mathcal{M}_2\mathcal{A}^{-1}\mathcal{M}_1$ is equivalent to the two matrices $\mathcal{M}_1\mathcal{A}^{-1}$ and $\mathcal{M}_2\mathcal{A}^{-1}$ being commutative, according to Lemma 1, we know that $\mathcal{M}_1\mathcal{A}^{-1}$ and $\mathcal{M}_2\mathcal{A}^{-1}$ have a common set of orthonormal eigenvectors. That is say, there exists a unitary matrix $\mathcal{Q} \in \mathbb{R}^{nm\times nm}$ and two diagonal matrices $\Lambda_i = diag(\lambda_1^{(i)}, \lambda_2^{(i)}, \cdots, \lambda_{nm}^{(i)})$, $i = 1,2$, such that $\mathcal{Q}\mathcal{M}_i^{-1}\mathcal{A}\mathcal{Q}^* = \Lambda_i$, $i = 1,2$. Noticing that

$$\begin{aligned} \mathcal{T} &= \mathcal{M}_2^{-1}\mathcal{N}_2\mathcal{M}_1^{-1}\mathcal{N}_1 \\ &= \mathcal{M}_2^{-1}(\mathcal{M}_2 - \mathcal{A})\mathcal{M}_1^{-1}(\mathcal{M}_1 - \mathcal{A}) \\ &= (I - \mathcal{M}_2^{-1}\mathcal{A})(I - \mathcal{M}_1^{-1}\mathcal{A}) \\ &= (\mathcal{Q}^*\mathcal{Q} - \mathcal{Q}^*\mathcal{Q}\mathcal{M}_2^{-1}\mathcal{A}\mathcal{Q}^*\mathcal{Q})(\mathcal{Q}^*\mathcal{Q} - \mathcal{Q}^*\mathcal{Q}\mathcal{M}_1\mathcal{A}\mathcal{Q}^*\mathcal{Q}) \\ &= (\mathcal{Q}^*\mathcal{Q} - \mathcal{Q}^*\Lambda_2\mathcal{Q})(\mathcal{Q}^*\mathcal{Q} - \mathcal{Q}^*\Lambda_1\mathcal{Q}) \\ &= \mathcal{Q}^*(I - \Lambda_2)\mathcal{Q}^*\mathcal{Q}(I - \Lambda_1)\mathcal{Q} \\ &= \mathcal{Q}^*(I - \Lambda_2)(I - \Lambda_1)\mathcal{Q} \end{aligned}$$

by definition we have

$$\begin{aligned}\rho(\mathcal{T}) &\leq \max\nolimits_{1\leq i,j\leq nm} |(1-\lambda_i^{(2)})(1-\lambda_j^{(1)})| \\ &\leq \max\nolimits_{1\leq i\leq nm} |(1-\lambda_i^{(2)})| \max\nolimits_{1\leq j\leq nm} |(1-\lambda_j^{(1)})| \\ &= \rho(I-\mathcal{M}_2^{-1}\mathcal{A})\rho(I-\mathcal{M}_1^{-1}\mathcal{A}) \\ &= \rho(\mathcal{M}_2^{-1}\mathcal{N}_2)\rho(\mathcal{M}_1^{-1}\mathcal{N}_1) \\ &\leq ||\mathcal{M}_2^{-1}\mathcal{N}_2||_{\mathcal{M}_2}||\mathcal{M}_1^{-1}\mathcal{N}_1||_{\mathcal{M}_1}\end{aligned}$$

Therefore, by Lemma 5 we have

$$\rho(\mathcal{T}) \leq \theta_1^3 \frac{\max|\lambda(N_1)| + \max|\lambda(Q_1)|}{\lambda_{\min}(M_1)+\lambda_{\min}(P_1)} \theta_2^3 \frac{\max|\lambda(N_2)| + \max|\lambda(Q_2)|}{\lambda_{\min}(M_2)+\lambda_{\min}(P_2)} = \varrho_1\varrho_2$$

and this completes the proof. □

3. Numerical Results

All numerical experiments presented in this section were computed in double precision with a number of MATLAB R2018a codes. All iterations are started from the initial zero matrix $X^{(0)}$ and terminated when the current iteration satisfies

$$\frac{||R^{(k)}||_F}{||R^{(0)}||_F} \leq 10^{-8},$$

where $R^{(k)} = C - AX^{(k)} - X^{(k)}B$ is the residual of the kth iterate. Also, we use the tolerance $\varepsilon = 0.01$ for inner iterations in corresponding methods. For each experiment, we report the number of iterations or the number of total outer iteration steps (shown as out-itr) and CPU time. In the tables, the norm of the residual is shown as res-norm.

The MSI method was compared with two Hermitian- and skew-Hermitian-based splitting methods such the NSCG [17] and the HSS [6], and two familiar iterative methods such the GMRES [34] and the BiCGSTAB [14] methods. Note that although the NS-CGNR method is a Hermitian- and skew-Hermitian-based splitting method, it works well for problems with a dominant skew-Hermitian part and not efficiently for problems with a dominant Hermitian part, see [18]. Therefore, it is not fair to compare this version of the MSI method with it.

Example 1. *For this example, we use the matrices*

$$A = B = M + 2rN + \frac{100}{(n+1)^2}I,$$

where $M, N \in \mathbb{R}^{n\times n}$ are the tridiagonal matrices given by

$$M = \mathrm{tridiag}(-1,2,-1) \textit{ and } N = \mathrm{tridiag}(0.5,0,-0.5).$$

We consider $r = 0.01$ and $n = m = 256$ [6].

This class of problems may arise in the preconditioned Krylov subspace iteration methods used for solving the systems of linear equations resulting from the finite difference or Sinc–Galerkin discretization of various differential equations and boundary value problems [6].

We apply the iteration methods to this problem. The results are given in Table 1.

Table 1. Results of the Example 1.

Method	Out-Itr	CPU Time	Res-Norm
MSI	7	10.54	2.0887×10^{-6}
NSCG	7	8.19	2.1050×10^{-6}
HSS	298	75.32	3.2107×10^{-6}
GMRES(10)	151	40.56	3.0400×10^{-6}
BiCGSTAB	255	17.91	2.4616×10^{-6}

From the results presented in Table 1, it can be seen that for this problem, the MSI and the NSCG methods are more efficient than the other methods.

Example 2. *For the second experiment, consider* $A = \mathrm{tridiag}(-2, 4, -1)$ *and* $B = \mathrm{tridiag}(-1, 4, -2)$ *with dimensions* 2048×2048 *and* 128×128*, respectively.*

This is a problem with a strong Hermitian part [42,43]. The numerical results for this problem are listed in Table 2.

Table 2. Results of Example 2.

Method	Out-Itr	CPU Time	Res-Norm
MSI	7	5.66	4.3252×10^{-5}
NSCG	9	6.18	7.7746×10^{-5}
HSS	21	39.82	8.7083×10^{-5}
GMRES(10)	3	2.84	2.3509×10^{-6}
BiCGSTAB	14	2.67	9.7977×10^{-5}

Regarding Table 2, it is obvious that although the MSI method is more effective versus the NSCG and the HSS methods, the GMRES and the BiCGSTAB methods are more effective than it.

Example 3. *We consider the continuous Sylvester Equation* (1) *with* $n = m$ *and the coefficient matrices*

$$\begin{cases} A = diag(1, 2, \cdots, n) + rL^T, \\ B = 2^{-t} I_n + diag(1, 2, \cdots, n) + rL^T + 2^{-t} L, \end{cases}$$

where L is the strictly lower triangular matrix having ones in the lower triangle part [6]. Here, t is a problem parameter to be specified in actual computations.

The iteration methods were used for this problem, and the results are given in Table 3. Moreover, we compare the convergence history of the iterative methods by residual norm decreasing in Figure 1.

Table 3. Results of Example 3.

Method	Out-Itr	CPU Time	Res-Norm
MSI	5	16.43	0.0029
NSCG	8	21.65	0.0070
HSS	99	326.71	0.0288
GMRES(10)	20	49.87	0.0027
BiCGSTAB	75	16.22	0.0028

In Table 3, we report the number of outer iterations (out-itr), the CPU time and the residual norm (res-norm) after convergence. For this example, we observe that the MSI method is superior to the other iterative methods in terms of the number of iterations and it is similar to the BiCGSTAB method in terms of CPU time. Comparing the convergence history of the iterative methods by residual norm decreasing shows that the MSI method converges more rapidly and smoothly than the BiCGSTAB method (see Figure 1).

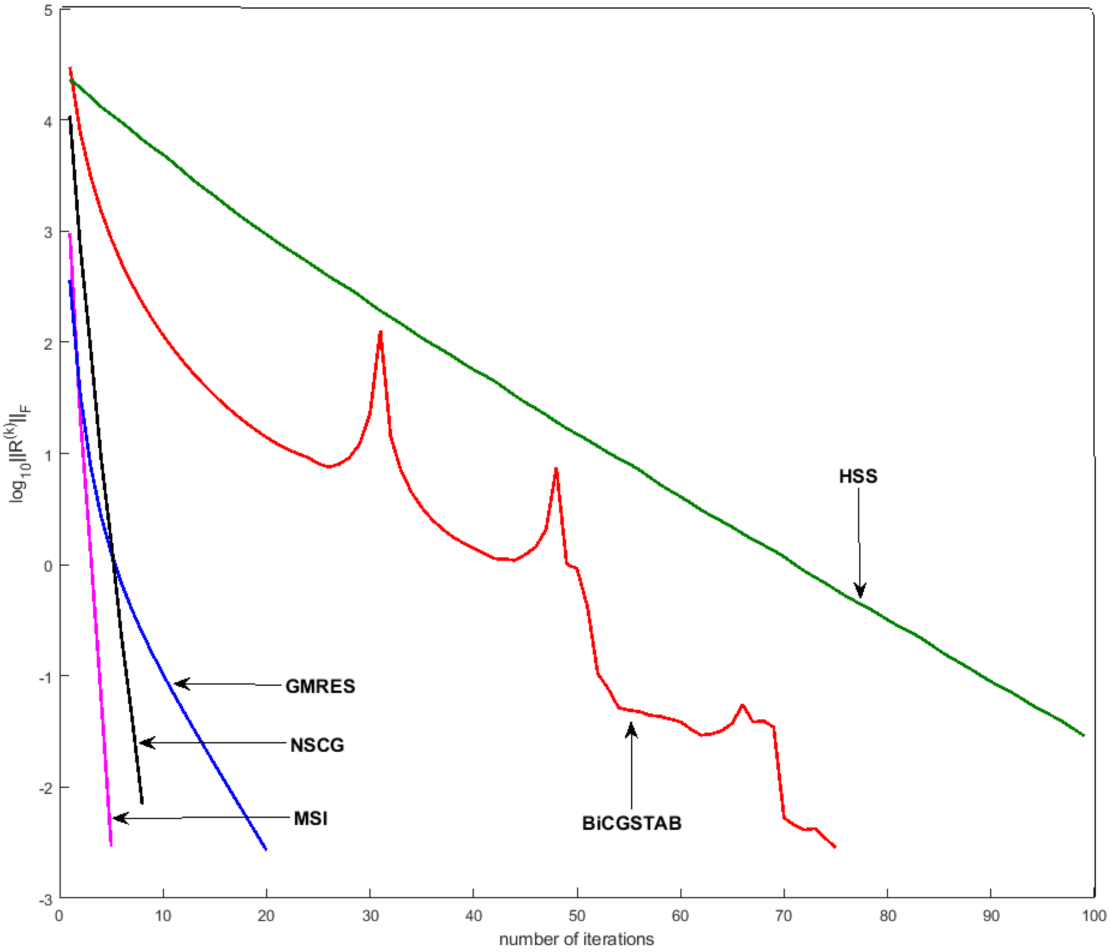

Figure 1. Convergence history of MSI versus the other iterative methods for Example 3.

Example 4. *For this example, we used the nonsymmetric sparse matrix SHERMAN3 of dimension* 5005×5005 *with* 20033 *nonzero entries from the Harwell–Boeing collection [44] instead of the coefficient matrix A. For the coefficient matrix B, we used* $B = \text{tridiag}(-1, 4, -2)$ *of dimension* 8×8 *[17].*

We apply the iteration methods to this problem, and the results are given in Table 4. Moreover, we compare the convergence history of the iterative methods by residual norm decreasing in Figure 2.

Table 4. Results of Example 4.

Method	Out-Itr	CPU Time	Res-Norm
MSI	34	78.437	1.57×10^{-4}
NSCG	64	121.265	2.61×10^{-4}
HSS	>5000	>1000	2.32
GMRES(10)	>5000	>1000	247.77
BiCGSTAB	†	†	NaN

In Table 4, we report the number of outer iterations (out-itr), the CPU time and the residual norm (res-norm) after convergence or in 5000 outer iterations. For this example, we observe that the MSI method is superior to the other iterative methods in terms of the number of iterations and CPU times, the NSCG method has an acceptable performance. Furthermore, the HSS and the GMRES methods have a very slow convergence rate (see Figure 2). From the Table 4 and Figure 2, we observe that the BiCGSTAB method was diverged for this problem. Therefore, we use splitting of the MSI, the NSCG and the HSS method as the splitting preconditioner denoted as MSI-BiCGSTAB, NSCG-BiCGSTAB and HSS-BiCGSTAB, respectively. The results of the preconditioned method for these preconditioners are given in Table 5. In Tables 4 and 5 dagger (†) and notation ">1000" show that no solution has been obtained after 5000 iteration or CPU time is more than 1000 s respectively.

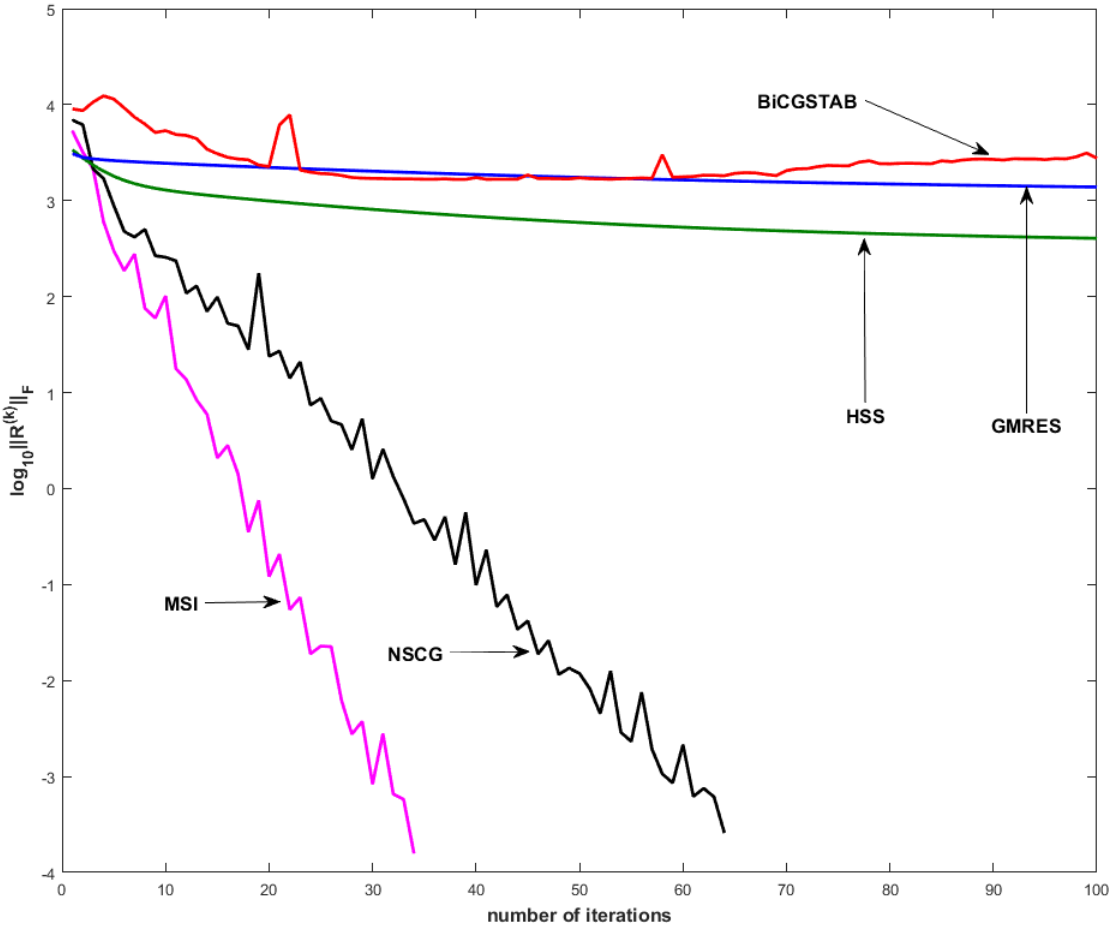

Figure 2. Convergence history of MSI versus the other iterative methods for Example 4.

Table 5. Results of the preconditioned BiCGSTAB for Example 4.

Method	Out-Itr	CPU Time	Res-Norm
BiCGSTAB	†	†	NaN
MSI-BiCGSTAB	2	113.51	9.0749×10^{-6}
NSCG-BiCGSTAB	4	263.92	5.6612×10^{-6}
HSS-BiCGSTAB	†	†	NaN

The results in Table 5 show that the use of the MSI method as a preconditioner improves the results obtained by the corresponding methods.

4. Future Work

As an important challenge in the fractional mathematical field, the extension of this work for solving the multi-term fractional Sylvester equation with frequency is of interest because most of the existing methods are only useful for untangling the multi-term fractional differential equation. Therefore, investigation of capturing appropriate methods that are applicable to solve a specific type of multi-term fractional equation is attractive and meaningful [45–48].

5. Conclusions

In this paper, we have proposed an efficient iterative method for solving the continuous Sylvester equation $AX + XB = C$. This method employs two symmetric positive definite splittings of the coefficient matrices A and B and present a multiplicative splitting iteration method. The convergence conditions have been derived based on the iteration matrix.

We have compared the MSI method with well-known iterative methods such as the NSCG method, the HSS method, the BiCGSTAB method and the GMRES method for some problems. We have observed that, for these problems, the MSI method is more efficient than the other methods.

In summary, by focusing on the results presented in Tables 1 and 3–5, one can observe that the MSI method is often superior to the other iterative methods. Moreover, the use of the multiplicative splitting as a preconditioner can induce an accurate and effective preconditioned BiCGSTAB method.

Author Contributions: Conceptualization, M.K.Z.; Methodology, M.K.Z. and A.A.S.; Software, M.K.Z. and A.A.S.; Validation, M.K.Z.; Investigation, M.K.Z.; Writing—original draft, M.K.Z.; Writing—review & editing, A.A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors are grateful to the anonymous referees for their comments which substantially improved the quality of this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Datta, B. *Numerical Methods for Linear Control Systems*; Elsevier: Amsterdam, The Netherlands; Academic Press: Cambridge, MA, USA, 2004.
2. Van der Schaft, A. *L_2-Gain and Passivity Techniques in Nonlinear Control*, 2nd ed.; Springer: London, UK, 2000.
3. Baur, U.; Benner, P. Cross-Gramian based model reduction for data-sparse systems. *Electron. Trans. Numer. Anal.* **2008**, *31*, 256–270.
4. Anderson, B.D.O.; Agathoklis, P.; Jury, E.I.; Mansour, M. Stability and the matrix Lyapunov equation for discrete 2-dimensional systems. *IEEE Trans. Circuits Syst.* **1986**, *33*, 261–267. [CrossRef]
5. Bouhamidi, A.; Jbilou, K. Sylvester Tikhonov-Regularization methods in image restoration. *J. Comput. Appl. Math.* **2007**, *206*, 86–98. [CrossRef]
6. Bai, Z.-Z. On Hermitian and skew-Hermitian splitting iteration methods for continuous Sylvester equations. *J. Comput. Math.* **2011**, *29*, 185–198.
7. Beik, F.P.A.; Salkuyeh, D.K. Weighted versions of Gl-FOM and Gl-GMRES for solving general coupled linear matrix equations. *Comput. Math. Math. Phys.* **2015**, *55*, 1606–1618. [CrossRef]
8. Dehghan, M.; Hajarian, M. Two algorithms for finding the Hermitian reflexive and skew-Hermitian solutions of Sylvester matrix equations. *Appl. Math. Lett.* **2011**, *24*, 444–449. [CrossRef]
9. Hajarian, M. Solving the general Sylvester discrete-time periodic matrix equations via the gradient based iterative method. *Appl. Math. Lett.* **2016**, *52*, 87–95. [CrossRef]
10. Jbilou, K. An Arnoldi based algorithm for large algebraic Riccati equations. *Appl. Math. Lett.* **2006**, *19*, 437–444. [CrossRef]
11. Simoncini, V. Computational methods for linear matrix equations. *SIAM Rev.* **2016**, *58*, 377–441. [CrossRef]

12. Dehghan, M.; Shirilord, A. The double-step scale splitting method for solving complex Sylvester matrix equation. *Comp. Appl. Math.* **2019**, *38*, 444–449. [CrossRef]
13. El Guennouni, A.; Jbilou, K.; Riquet, J. Block Krylov subspace methods for solving large Sylvester equation. *Numer. Algorithms* **2002**, *29*, 75–96. [CrossRef]
14. El Guennouni, A.; Jbilou, K.; Sadok, H. A block version of BiCGSTAB for linear systems with multiple right-hand sides. *Electron. Trans. Numer. Anal.* **2004**, *16*, 243–256.
15. Hajarian, M. Extending the CGLS algorithm for least squares solutions of the generalized Sylvester-transpose matrix equations. *J. Frankl. Inst.* **2016**, *353*, 1168–1185. [CrossRef]
16. Hajarian, M. Matrix iterative methods for solving the Sylvester-transpose and periodic Sylvester matrix equations. *J. Frankl. Inst.* **2013**, *350*, 3328–3341. [CrossRef]
17. Khorsand Zak, M.; Toutounian, F. Nested splitting CG-like iterative method for solving the continuous Sylvester equation and preconditioning. *Adv. Comput. Math.* **2014**, *40*, 865–880. [CrossRef]
18. Khorsand Zak, M.; Toutounian, F. An iterative method for solving the continuous Sylvester equation by emphasizing on the skew-Hermitian parts of the coefficient matrices. *Int. J. Comput. Math.* **2017**, *94*, 633–649. [CrossRef]
19. Tohidi, E.; Khorsand Zak, M. A new matrix approach for solving second-order linear matrix partial differential equations. *Mediterr. J. Math.* **2016**, *13*, 1353–1376. [CrossRef]
20. Bouhamidi, A.; Elbouyahyaoui, L.; Heyouni, M. The constant solution method for solving large-scale differential Sylvester matrix equations with time invariant coefficients. *Numer Algor* **2024**, *96*, 449–488. [CrossRef]
21. Chu, E.K.W.; Hou, L.; Szyld, D.B.; Zhou, J. Numerical solution of singular Sylvester equations. *J. Comput. Appl. Math.* **2024**, *436*, 115426. [CrossRef]
22. Sasaki, N.; Chansangiam, P. Modified Jacobi-Gradient Iterative Method for Generalized Sylvester Matrix Equation. *Symmetry* **2020**, *12*, 1831. [CrossRef]
23. Shafiei, S.G.; Hajarian, M. An iterative method based on ADMM for solving generalized Sylvester matrix equations. *J. Frankl. Inst.* **2022**, *15*, 8155–8170. [CrossRef]
24. Yu, S.; Peng, J.; Tang, Z.; Peng, Z. Iterative methods to solve the constrained Sylvester equation. *AIMS Math.* **2023**, *8*, 21531–21553. [CrossRef]
25. Chu, E.K.W.; Szyld, D.B.; Zhou, J. Numerical solution of singular Lyapunov equations. *Numer. Linear Algebra Appl.* **2021**, *28*, e2381. [CrossRef]
26. Du, X.; Iqbal, K.I.B.; Uddin, M.M.; Hossain, M.T.; Shuzan, M.N.I. A computationally effective time-restricted stability preserving H2-optimal model order reduction approach. *Results Control. Optim.* **2023**, *11*, 100217. [CrossRef]
27. Jiang, W.; Ge, S.S.; Hu, Q.; Li, D. Sliding-Mode Control for Perturbed MIMO Systems With Time-Synchronized Convergence. *IEEE Trans. Cybern.* **2024**, *54*, 4375–4388. [CrossRef]
28. Ma, L.; Chang, R.; Han, M.; Jiao, Y. Explicit solutions of conjugate, periodic, time-varying Sylvester equations. *J. Inequal. Appl.* **2023**, *2023*, 133. [CrossRef]
29. Sarumi, I.O.; Furati, K.M.; Khaliq, A.Q.M. Efficient second-order accurate exponential time differencing for time-fractional advection–diffusion–reaction equations with variable coefficients. *Math. Comput. Simul.* **2025**, *230*, 20–38. [CrossRef]
30. Bartels, R.H.; Stewart, G.W. Algorithm 432: Solution of the matrix equation $AX + XB = C$. *Circ. Syst. Signal Proc.* **1994**, *13*, 820–826. [CrossRef]
31. Golub, G.H.; Nash, S.; Van Loan, C. A Hessenberg-Schur method for the problem $AX + XB = C$. *IEEE Trans. Contr.* **1979**, *AC-24*, 909–913. [CrossRef]
32. Benner, P.; Li, R.C.;Truhar, N. On the ADI method for Sylvester equations. *J. Comput. Appl. Math.* **2009**, *233*, 1035–1045. [CrossRef]
33. Hu, D.Y.; Reichel, L. Krylov-subspace methods for the Sylvester equation. *Linear Algebra Appl.* **1992**, *172*, 283–313. [CrossRef]
34. Salkuyeh, D.K.; Toutounian, F. New approaches for solving large Sylvester equations. *Appl. Math. Comput.* **2006**, *173*, 9–18.
35. Khorsand Zak, M.; Toutounian, F. Nested splitting conjugate gradient method for matrix equation $AXB = C$ and preconditioning. *Comput. Math. Appl.* **2013**, *66*, 269–278. [CrossRef]
36. Bai, Z.-Z. On the convergence of additive and multiplicative splitting iterations for systems of linear equations. *J. Comput. Appl. Math.* **2003**, *154*, 195–214. [CrossRef]
37. Saad, Y. *Iterative Methods for Sparse Linear Systems*, 2nd ed.; SIAM: Philadelphia, PA, USA, 2003.
38. Evans, D.J.; Wan, C.R. A preconditioned conjugate gradient method for $AX + XB = C$. *Int. J. Comput. Math.* **1993**, *49*, 207–219. [CrossRef]
39. Bai, Z.-Z.; Yin, J.-F.; Su, Y.-F. A shift-splitting preconditioner for non-Hermitian positive definite matrices. *J. Comput. Math.* **2006**, *24*, 539–552.
40. Kelley, C.T. *Iterative Methods for Linear and Nonlinear Equations*; No. 16, Frontiers in Applied Mathematics; SIAM: Philadelphia, PA, USA, 1995.
41. Lütkepohl, H. *Handbook of Matrices*; John Wiley & Sons Press: England, UK, 1996.

42. Krukier, L.A.; Martynova, T.S.; Bai, Z.-Z. Product-type skew-Hermitian triangular splitting iteration methods for strongly non-Hermitian positive definite linear systems. *J. Comput. Appl. Math.* **2009**, *232*, 3–16 [CrossRef]
43. Wang, L.; Bai, Z.-Z. Skew-Hermitian triangular splitting iteration methods for non-Hermitian positive definite linear systems of strong skew-Hermitian parts. *BIT Numer. Math.* **2004**, *44*, 363–386. [CrossRef]
44. Duff, I.S.; Grimes, R.G.; Lewis, J.G. U*ser's Guide for the Harwell-Boeing Sparse Matrix Collection*; Technical Report RAL-92-086; Rutherford Applton Laboratory: Chilton, UK, 1992.
45. Dwivedi, H.K. A fast difference scheme for the multi-term time fractional advection diffusion equation with a non-linear source term. *Chin. J. Phys.* **2024**, *85*, 86–103. [CrossRef]
46. Lin, J.; Reutskiy, S.; Zhang, Y.; Sun, Y.; Lu, J. The novel analytical–numerical method for multi-dimensional multi-term time-fractional equations with general boundary conditions. *Mathematics* **2023**, *11*, 929. [CrossRef]
47. Rahman, G.; Wahid, F.; Gómez-Aguilar, J.F.; Ali, A. Analysis of multi-term arbitrary order implicit differential equations with variable type delay. *Phys. Scr.* **2024**, *99*, 115246. [CrossRef]
48. Santra, S. Analysis of a higher-order scheme for multi-term time-fractional integro-partial differential equations with multi-term weakly singular kernels. *Numer. Algorithms* **2024**. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.