symmetry

MDPI

Article

Gradient Iterative Method with Optimal Convergent Factor for Solving a Generalized Sylvester Matrix Equation with Applications to Diffusion Equations

Nunthakarn Boonruangkan and Pattrawut Chansangiam *

Department of Mathematics, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand; 60605009@kmitl.ac.th

* Correspondence: pattrawut.ch@kmitl.ac.th; Tel.: +66-935-266600

Received: 6 October 2020; Accepted:16 October 2020; Published: 20 October 2020

Abstract: We introduce a gradient iterative scheme with an optimal convergent factor for solving a generalized Sylvester matrix equation $\sum_{i=1}^{p} A_i X B_i = F$, where A_i, B_i and F are conformable rectangular matrices. The iterative scheme is derived from the gradients of the squared norm-errors of the associated subsystems for the equation. The convergence analysis reveals that the sequence of approximated solutions converge to the exact solution for any initial value if and only if the convergent factor is chosen properly in terms of the spectral radius of the associated iteration matrix. We also discuss the convergent rate and error estimations. Moreover, we determine the fastest convergent factor so that the associated iteration matrix has the smallest spectral radius. Furthermore, we provide numerical examples to illustrate the capability and efficiency of this method. Finally, we apply the proposed scheme to discretized equations for boundary value problems involving convection and diffusion.

Keywords: gradient; linear iterative process; matrix norm; generalized Sylvester matrix equation; convection–diffusion equation

MSC: 15A12; 15A60; 15A69; 65F45; 65N22

1. Introduction

It is well known that several problems in control and system theory are closely related to a generalized Sylvester matrix equation of the form

$$\sum_{i=1}^{p} A_i X B_i = F, \tag{1}$$

where A_i, B_i and F are given matrices of conforming dimensions. Equation (1) includes the following special cases:

$$AX + XB = F, \tag{2}$$
$$AX + XA^T = F, \tag{3}$$
$$AXB + X = F, \tag{4}$$

known respectively as the Sylvester equation, the Lyapunov equation, and the Kalman–Yakubovich equation. Equations (1)–(4) have important applications in stability analysis, optimal control, observe design, output regulation problem, and so on; see e.g., [1–3]. Equation (1) can be solved directly using the vector operator and the Kronecker product. Here, recall that the vector operator

vec[·] turns each matrix into a column vector by stacking its columns consecutively. The Kronecker product of two matrices $A = [a_{ij}]$ and B is defined to be the block matrix $A \otimes B = [a_{ij}B]$. In fact, Equation (1) can be reduced to the linear system

$$Px = b \quad \text{where} \quad P = \sum_{i=1}^{p}(B_i^T \otimes A_i), \quad b = \text{vec}[F] \quad \text{and} \quad x = \text{vec}[X].$$

Thus, (1) has a unique solution if and only if P is non-singular. In particular for the Sylvester Equation (2), the uniqueness of the solution is equivalent to the condition that A and $-B$ have no common eigenvalues. For Equation (4), the uniqueness condition is that all possible products of the eigenvalues of A and B are not equal to -1. The exact solution $x = P^{-1}b$ is, in fact, computationally difficult due to the large size of the Kronecker multiplication. This inspires us to investigate certain iterative schemes to generate a sequence of approximate solutions, which are arbitrarily close to the exact solution. Efficient iterative methods produce a satisfactory approximated solution in a small iteration number.

Many researchers have developed such iterative methods for solving a class of matrix Equations (1)–(4); see e.g., [4–10]. One of an interesting iterative method, called the Hermitian and skew Hermitian splitting iterative method (HSS), was investigated by many authors, e.g., [11–14]. Gradient-based iterative methods were firstly introduced by Ding and Chen for solving (1), (2) and (4). After that, there are many iterative methods for solving (1)–(4) based on gradients and hierarchical identification principle, e.g., [15–17]. Convergence analyses of such methods are often relied on the Frobenius norm $\|\cdot\|_F$ and the spectral norm $\|\cdot\|_2$, defined for each matrix A by

$$\|A\|_F = (\text{tr}\, A^T A)^{\frac{1}{2}} \quad \text{and} \quad \|A\|_2 = (\lambda_{\max}(A^T A))^{\frac{1}{2}}.$$

Method 1 ([15]). *Assume that the matrix Equation (1) has a unique solution X. Construct*

$$X_i(k) = X(k-1) + \tau A_i^T[F - \sum_{j=1}^{p} A_j X(k-1) B_j]B_i^T, \quad i = 1, 2, \ldots, p,$$
$$X(k) = \frac{1}{p}\sum_{i=1}^{p} X_i(k).$$

If we choose $\tau = [\sum_{i=1}^{p} \|A_i\|_2^2 \|B_i\|_2^2]^{-1}$, then the sequence $\{X(k)\}_{k=0}^{\infty}$ converges to the exact solution X for any given initial matrices $X_1(0), X_2(0), ..., X_p(0)$.

A least-squares based iterative method for solving (1) was introduced as follows:

Method 2 ([15]). *Assume that the matrix Equation (1) has a unique solution X. For each $i = 1, 2, \ldots, p$, construct,*

$$X_i(k) = X(k-1) + \tau \sum_{i=1}^{p}(A_i^T A_i)^{-1} A_i^T[F - \sum_{j=1}^{p} A_j X(k-1) B_j]B_i^T(B_i B_i^T)^{-1}.$$

Compute

$$X(k) = \frac{1}{p}\sum_{i=1}^{p} X_i(k).$$

If we choose $0 < \tau < 2p$, then the sequence $\{X(k)\}_{k=0}^{\infty}$ converges to the exact solution X for any given initial matrices $X_1(0), X_2(0), \ldots, X_p(0)$.

In this paper, we propose a gradient-based iterative method with an optimal convergent factor (GIO) for solving the generalized Sylvester matrix Equation (1). This method is derived from least-squares optimization and hierarchical identification principle (see Section 2). Convergence analysis (see Section 3) reveals that the sequence of approximated solutions converges to the exact solution for any initial value if and only if the convergent factor is chosen properly. Then we discuss the convergent rate and error estimates for the method. Moreover, the convergent factor will be determined so that the convergent rate is fastest, or equivalently, the spectral radius of associated iteration matrix is minimized. In particular, the GIO method can solve the Sylvester Equation (2) (see Section 4). To illustrate the efficiency of the proposed method, we provide numerical experiments in Section 5. We compare the efficiency of our method for solving (2) with other iterative methods such as gradient based iterative method (GI) [15], least-squares iterative method (LS) [17], relaxed gradient based iterative method (RGI) [18], modified gradient based iterative method (MGI) [19], Jacobi-gradient based iterative method (JGI) [20,21] and accelerated Jacobi-gradient based iterative method AJGI [22]. In Section 6, we apply the GIO method to the convection–diffusion and the diffusion equation. Finally, we conclude the overall work in Section 7.

2. Introducing a Gradient Iterative Method

Let us denote by $\mathbb{R}^{r\times s}$ the set of $r \times s$ real matrices. Let $m, n, p, q \in \mathbb{N}$ be such that $mq = np$. Consider the matrix Equation (1) where $A_i \in \mathbb{R}^{m\times n}$, $B_i \in \mathbb{R}^{p\times q}$, $F \in \mathbb{R}^{m\times q}$ are given constant matrices and $X \in \mathbb{R}^{n\times p}$ is an unknown matrix to be found. Suppose that (1) has a unique solution, i.e., the matrix P is invertible. Now, we discuss how to solve (1) indirectly using an effective iterative method. According to the hierarchical identification principle, the system (1) is decomposed into p subsystems. For each $i \in \{1, 2, ..., p\}$, set

$$M_i := F - \sum_{\substack{j=1\\ j\neq i}}^{p} A_j X B_j. \tag{5}$$

Our aim is to approximate the solution of p subsystems:

$$M_i = A_i X B_i, \qquad i \in \{1, 2, ..., p\}, \tag{6}$$

so that the following least-squares error is minimized:

$$L_i(X) := \frac{1}{2}\|A_i X B_i - M_i\|_F^2. \tag{7}$$

The gradient of each L_i can be computed as follows:

$$\begin{aligned}
\frac{\partial}{\partial x} L_i(X) &= \frac{1}{2}\frac{\partial}{\partial x}\,\mathrm{tr}[(A_i X B_i - M_i)^T (A_i X B_i - M_i)] \\
&= \frac{1}{2}\Big(\frac{\partial}{\partial x}\,\mathrm{tr}[B_i^T X^T A_i^T A_i X B_i] - 2\frac{\partial}{\partial x}\,\mathrm{tr}[M_i^T A_i X B_i]\Big) \\
&= \frac{1}{2}(A_i^T A_i X B_i B_i^T + A_i^T A_i X B_i B_i^T) + \frac{1}{2}(2 B_i M_i^T A_i)^T \\
&= A_i^T (F - \sum_{j=1}^{p} A_j X B_j) B_i^T.
\end{aligned} \tag{8}$$

Let $X_i(k)$ be the estimate or iterative solution at iteration k, associated with the subsystem (6). From the gradient formula (8), the iterative scheme for $X_i(k)$ is given by the following equation:

$$X_i(k) \;=\; X(k-1) + \tau A_i^T [F - \sum_{j=1}^{p} A_j X B_j] B_i^T, \qquad i = 1, 2, ..., p, \tag{9}$$

where τ is a convergent factor. According to the hierarchical identification principle, the unknown parameter X in (9) is replaced by its estimate $X(k-1)$. After taking the arithmetic mean of $X_i(k)$, we obtain the following process:

Method 3. *Gradient-based iterative method with optimal convergent factor*

Initializing step: For $i = 1, 2, \ldots, p$, set $A_i' = A_i^T$ and $B_i' = B_i^T$. Choose $\tau \in \mathbb{R}$. Set $k := 0$. Choose initial matrix $X(0)$.

Updating step: For $k = 1$ to end, do:

$$\begin{aligned} E(k-1) \;&=\; F - \sum_{j=1}^{p} A_j X(k-1) B_j, \\ X(k) \;&=\; \frac{1}{p} \sum_{i=1}^{p} (X(k-1) + \tau A_i' E(k-1) B_i'). \end{aligned}$$

Note that the terms $E(k), A_i', B_i'$ were introduced in order to eliminate duplicated computations. To stop the process, one may impose a stopping rule such as the relative error $\|E(k)\|_F / \|F\|_F$ is less than a tolerance error ϵ. The convergence property of this method depends on the convergent factor τ. A discussion of possible/optimal values of τ will be presented in the next section.

3. Convergence Analysis

In this section, we show that the approximated solutions derived from Method 3 converge to the exact solution. First, we transform a recursive equation of the error of approximated solutions into a first-order linear iterative system $x(k) = Tx(k-1)$ where $x(k)$ is a vector and T is an iteration matrix. Then, we investigate the iteration matrix T to obtain the convergence rate and error estimations. Finally we discuss the fastest convergent factor and find the number of iterations corresponding to a given satisfactory error.

Theorem 1. *Assume that the matrix Equation (1) has a unique solution X. Let $\tau \in \mathbb{R}$. Then the approximate solutions derived from (9) converge to the exact solution for any initial value $X(0)$ if and only if*

$$0 < \tau \;<\; \frac{2}{\|P\|_2^2}. \tag{10}$$

In this case, the spectral radius of the associated iteration matrix $T = I_{np} - \tau P^T P$ is given by

$$\rho[T] = \max\{|1 - \tau\lambda_{\max}(P^T P)|, |1 - \tau\lambda_{\min}(P^T P)|\}. \tag{11}$$

Proof. At each k-th iteration, consider the error matrix $\tilde{X}(k) = X(k) - X$. We have

$$\begin{aligned} \tilde{X}(k) &= X(k-1) + \tau \sum_{i=1}^{p} A_i^T E_i(k-1) B_i^T - X \\ &= \tilde{X}(k-1) - \tau \sum_{i=1}^{p} A_i^T E_i(k-1) B_i^T. \end{aligned}$$

We shall show that $X(k) \to X$ by showing that $\tilde{X}(k) \to 0$ or $\mathrm{vec}[\tilde{X}(k)] \to 0$. By taking the vector operator to the above equation, we get

$$\begin{aligned}
\operatorname{vec}\tilde{X}(k) &= \operatorname{vec}\tilde{X}(k-1) - \sum_{i=1}^{p} \tau \operatorname{vec} A_i^T E_i(k-1) B_i^T \\
&= \operatorname{vec}\tilde{X}(k-1) - \sum_{i=1}^{p} \tau (B_i \otimes A_i^T) \operatorname{vec} \sum_{i=1}^{p} A_i \tilde{X}(k-1) B_i \\
&= \operatorname{vec}\tilde{X}(k-1) - \sum_{i=1}^{p} \tau (B_i^T \otimes A_i)^T (\sum_{j=1}^{p} B_j^T \otimes A_j) \operatorname{vec}\tilde{X}(k-1) \\
&= T \operatorname{vec}\tilde{X}(k-1). \qquad (12)
\end{aligned}$$

We see that (12) is a first-order linear iterative system in the form $x(k) = Tx(k-1)$. Thus, $\operatorname{vec}\tilde{X}(k) \to 0$ for any initial values $X_i(0)$ if and only if the iteration matrix T has spectral radius less than 1. Since T is symmetric, all its eigenvalues are real. Note that any eigenvalue of T is of the form $1 - \tau\lambda$ where λ is an eigenvalue of $P^T P$. Thus, its spectral radius is given by (11). It follows that $\rho[T] < 1$ if and only if

$$0 < \tau\lambda_{\max}(P^T P) < 2 \qquad \text{and} \qquad 0 < \tau\lambda_{\min}(P^T P) < 2. \qquad (13)$$

Since P is invertible, the matrix $P^T P$ is positive definite. Thus, $\lambda_{\max}(P^T P) > 0$. The condition (13) now becomes

$$0 < \tau < \frac{2}{\lambda_{\max}(P^T P)} = \frac{2}{\|P\|_2^2}.$$

Hence, we arrive at (10). □

Theorem 2. *Assume the hypothesis of Theorem 1, so that the sequence $\{X_k\}$ converges to the exact solution X for any initial value $X(0)$.*

(1). We have the following error estimates

$$\|X(k) - X\|_F \le \rho[T] \, \|X(k-1) - X\|_F, \qquad (14)$$

$$\|X(k) - X\|_F \le \rho^k[T] \, \|X(0) - X\|_F. \qquad (15)$$

Moreover, the asymptotic convergence rate of Method 3 is governed by $\rho[T]$ in (11).

(2). Let $\varepsilon > 0$ be a satisfactory error. We have $\|X(k) - X\|_F < \epsilon$ after the k-th iteration for any $k \in \mathbb{N}$ that satisfies

$$k > \frac{\log\epsilon - \log\|X(0) - X\|_F}{\log\rho(T)}. \qquad (16)$$

Proof. According to (12), we have

$$\begin{aligned}
\|X(k) - X\|_F &= \|\tilde{X}(k)\|_F = \|\operatorname{vec}\tilde{X}(k)\|_F \\
&= \|T \operatorname{vec}\tilde{X}(k-1)\|_F \le \|T\|_2 \|\operatorname{vec}\tilde{X}(k-1)\|_F.
\end{aligned}$$

Since T is symmetric, we have $\|T\|_2 = \rho[T]$. Thus for each $k \in \mathbb{N}$, the approximation (14) holds. By induction, we obtain the estimation (15). The estimate (15) implies that the asymptotic convergence rate of the method depends on $\rho[T]$. To prove the assertion, we have by taking logarithms that the condition (16) is equivalent to

$$\rho^k(T)\|X(0) - X\|_F < \epsilon.$$

Thus if (16) holds, then $\|X(k) - X\|_F < \epsilon$. □

The convergence rate exhibits how fast of the approximated solutions converge to the exact solution. Theorem 2 reveals that the smaller the spectral radius $\rho[T]$, the faster the approximated

solutions go to the exact solution. Moreover, by taking $\epsilon = 0.5 \times 10^{-n}$ in (16), we have that $X(k)$ has an accuracy of n decimal digits if k satisfies

$$k > \frac{\log 0.5 - \log \|X(0) - X\|_F - n}{\log \rho(T)}.$$

Recall that the condition number of a matrix A (relative to the spectral norm) is defined by

$$\kappa(A) = \left(\frac{\lambda_{\max}(A^T A)}{\lambda_{\min}(A^T A)}\right)^{\frac{1}{2}}.$$

Theorem 3. *Assume the hypothesis of Theorem 1. Then the optimal value of $\tau > 0$ for which Method 3 has the fastest asymptotic convergence rate is determined by*

$$\tau_{opt} = \frac{2}{\lambda_{\max}(P^T P) + \lambda_{\min}(P^T P)}. \tag{17}$$

In this case, the spectral radius of the iteration matrix is given by

$$\rho[T] = \frac{\lambda_{\max}(P^T P) - \lambda_{\min}(P^T P)}{\lambda_{\max}(P^T P) + \lambda_{\min}(P^T P)} = \frac{\kappa^2(P) - 1}{\kappa^2(P) + 1}. \tag{18}$$

Proof. The convergence of Method 3 implies that (10) holds. Then, Method 3 has the convergence rate as the same to the linear iteration (12), and thus, it is governed by the spectral radius $\rho[T]$ in (11). The fastest convergence rate is equivalent to the smallest of $\rho[T]$. Thus, we make the following minimization:

$$\begin{aligned} &\text{Minimize} && \rho[T] = \max\{|1 - \tau\lambda_{\min}(P^T P)|, |1 - \tau\lambda_{\max}(P^T P)|\} \\ &\text{subject to} && 0 < \tau < \frac{2}{\lambda_{\max}(P^T P)}. \end{aligned}$$

Thus, the optimal value is reached at (17) so that the minimum is given by (18). □

We see that if the condition number of P is closer to 1 then the approximate solutions converge faster to the exact solution. Note that the condition number of P is close to 1 if and only if the maximum eigenvalue of $P^T P$ is close to the minimum eigenvalue of $P^T P$.

4. The GIO Method for the Sylvester Equation

In this section, we discuss the gradient-based iterative method with optimal convergent factor for solving Sylvester matrix equation. Moreover we discover convergence criteria, convergence rate, error estimate and optimal factor.

Let $m, n, p, q \in \mathbb{N}$ be such that $m = n$ and $p = q$. Consider the Sylvester matrix Equation (2) where $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{p\times q}$, $F \in \mathbb{R}^{m\times q}$ are given constant matrices and $X \in \mathbb{R}^{n\times p}$ is an unknown matrix to be found. Suppose that (2) has a unique solution, i.e., $Q := I_p \otimes A + B^T \otimes I_n$ is invertible, or equivalently, A and $-B$ have no common eigenvalues.

Method 4. *Initializing step: Set $A' = A^T, B' = B^T$. Choose $\tau \in \mathbb{R}$. Set $k := 0$. Choose initial matrix $X(0)$.*

Updating step: For $k = 1$ to end, do:

$$\begin{aligned} E(k-1) &= F - AX(k-1) - X(k-1)B, \\ X(k) &= X(k-1) + \tau[A'E(k-1)B']. \end{aligned}$$

Corollary 1. *Assume that the Sylvester matrix Equation (2) has a unique solution X. Let $\tau \in \mathbb{R}$. Then the following hold:*

(i) *The approximate solutions generated by Method 4 converge to the exact solution for any initial value $X(0)$ if and only if*

$$0 < \tau < \frac{2}{\|Q\|_2^2}. \tag{19}$$

In this case, the spectral radius of the associated iteration matrix $S = I_{np} - \tau Q^T Q$ is given by

$$\rho[S] = \max\{|1 - \tau\lambda_{\max}(Q^T Q)|, |1 - \tau\lambda_{\min}(Q^T Q)|\}. \tag{20}$$

(ii) *The asymptotic convergence rate of Method 4 is governed by $\rho[S]$ in (20).*

(iii) *The optimal value of $\tau > 0$ for which Method 4 has the fastest asymptotic convergence rate is determined by*

$$\tau_{opt} = \frac{2}{\lambda_{\max}(Q^T Q) + \lambda_{\min}(Q^T Q)}. \tag{21}$$

Remark 1. *Note that Q is the Kronecker sum of A and B^T. Thus, if A and B are positive semidefinite, then*

$$\begin{aligned} \|Q\|_2^2 = \lambda_{\max}(Q^T Q) = \lambda_{\max}^2(Q) &= (\lambda_{\max}(A) + \lambda_{\max}(B))^2, \\ \lambda_{\min}(Q^T Q) = \lambda_{\min}^2(Q) &= (\lambda_{\min}(A) + \lambda_{\min}(B))^2. \end{aligned}$$

5. Numerical Examples for Generalized Sylvester Matrix Equation

In this section, we show the capability and efficiency of the proposed method by illustrating some numerical examples. To compare the performance of any algorithms, we must use the same PC environment, and consider informed errors together with iteration numbers (IT) and computational times (CT: in seconds). Our iterations have been carried out by MATLAB R2013a, Intel(R) Core(TM) i5-760 CPU @ 2.80 GHz, RAM 8.00 GB PC environment. We measure the computational time taken for an iterative process by the MATLAB functions tic and toc. In Example 1, we show that our method is also efficient although matrices are non-square and we discuss the effect of changing the convergent factor τ. In Example 2, we consider a larger square matrix system and show that our method is still efficient. In Example 3, we compare the efficiency of our method to another recent iterative methods. The matrix equation considered in this example is the Sylvester equation with square coefficient matrices since it fits with all of the recent methods. In all illustrated examples, we compare the efficiency of iterative methods to the direct method $x = P^{-1}b$ mentioned in Introduction. Let us denote by $\text{tridiag}(u, v, w)$ the tridiagonal matrix with main diagonal u, v and w.

Example 1. *Consider the matrix equation $A_1XB_1 + A_2XB_2 + A_3XB_3 = F$ when $A_1, A_2, A_3 \in \mathbb{R}^{40\times 60}$, $B_1, B_2, B_3 \in \mathbb{R}^{20\times 30}$ and $F \in \mathbb{R}^{40\times 30}$ are tridiagonal matrices given by*

$$\begin{aligned} A_1 &= \text{tridiag}(-2, 2, -2), \quad A_2 = \text{tridiag}(2, -2, 5), \quad A_3 = \text{tridiag}(2, -1, 2), \\ B_1 &= \text{tridiag}(4, 3, -1), \quad B_2 = \text{tridiag}(1, -2, -1), \quad B_3 = \text{tridiag}(3, 1, 3). \end{aligned}$$

Here, the exact solution is given by $X = \text{tridiag}(1, -1, 1)$. We apply Method 3 to compute the sequence $X(k)$ of approximated solutions. Take initial point

$$X(0) = 10^{-6} \times \text{tridiag}(1, 1, 1).$$

The optimal convergent factor can be computed as follows:

$$\tau_{opt} = \frac{2}{\lambda_{\min}(P^T P) + \lambda_{\max}(P^T P)} \approx \frac{2}{4.15 \times 10^{-13} + 1009.74} \approx 0.0019806.$$

The effect of changing convergent factors τ is illustrated in Figure 1. We see that as k large enough, the relative error $\|E(k)\|_F/\|F\|_F$ for τ_{opt} goes faster to 0 than for other convergent factors. If τ does not satisfy the condition (10), then the approximated solutions diverge for the given initial matrices. Moreover, Table 1 shows that the computational time of our algorithm (GIO) is significantly less than the time of the direct method. Table 1 also demonstrates that, when we fix the error $\|E(k)\|_F$ to be less than 5×10^{-3}, the GIO algorithm outperforms another GI algorithms with different convergent factors in both iteration numbers and computational times.

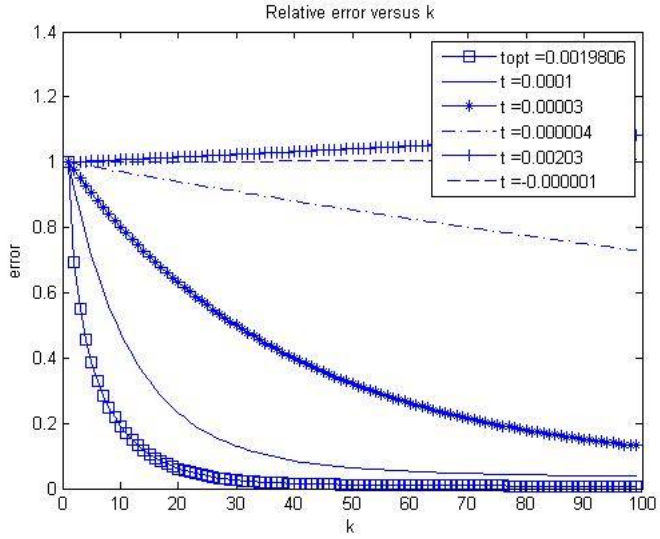

Figure 1. Relative error for Example 1.

Table 1. Iteration numbers and computational times for Example 1.

Method	IT	CT
Direct	-	3.1380
GIO	161	0.0413
GI ($\tau = 0.0001$)	3061	0.2508
GI ($\tau = 0.00003$)	10,204	0.8994

Example 2. *Consider the matrix equation $A_1XB_1 + A_2XB_2 + A_3XB_3 = F$ where all matrices are 100×100 tridiagonal matrices given by*

$$A_1 = \text{tridiag}(1,2,1), \quad A_2 = \text{tridiag}(-1,-2,-1), \quad A_3 = \text{tridiag}(-1,3,-1),$$
$$B_1 = \text{tridiag}(2,2,3), \quad B_2 = \text{tridiag}(1,2,-2), \quad B_3 = \text{tridiag}(3,2,-1).$$

Here, the exact solution is $X = \text{tridiag}(1,1,1)$. To apply Method 3, we take initial matrix

$$X(0) \; = \; 10^{-6} \times \text{tridiag}(0,2,0).$$

We can compute $\tau_{opt} \approx 0.002553$. Figure 2 shows that the relative error $\|E(k)\|_F \; / \; \|F\|_F$ for τ_{opt} goes faster to 0 than for other convergent factors. If τ does not satisfy (10), then the approximate solutions diverge for the given initial matrices. From Table 2, we see that the computational time of our algorithm is significantly less than the time of the direct method. Furthermore, when the satisfactory error $\|E(k)\|_F$ is less than $\epsilon = 0.5$, the GIO algorithm has more efficiency than another GI algorithms in both iteration numbers and computational times.

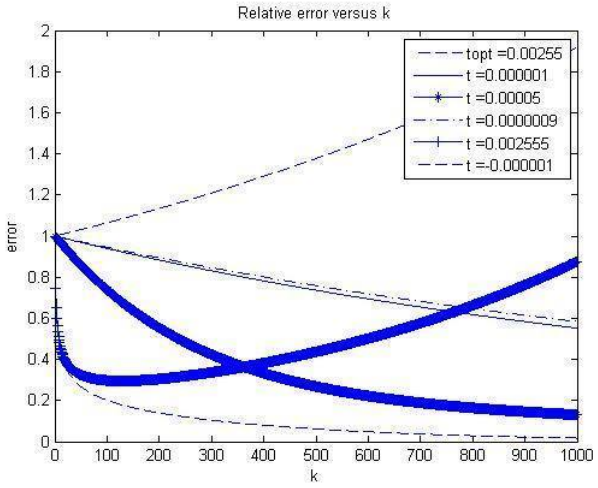

Figure 2. Relative errors for Example 2.

Table 2. Iteration numbers and computational times for Example 2.

Method	IT	CT
Direct	-	53.4063
GIO	389	0.5439
GI ($\tau = 0.00005$)	19,314	28.0245
GI ($\tau = 0.000001$)	96,557	148.4039

Example 3. *Consider the Sylvester equation* $AX + XB = F$, *where* $A, X, B, F \in \mathbb{R}^{10\times 10}$ *are given by* $A = \text{tridiag}(-1, 3, 1)$, $B = \text{tridiag}(-3, 2, 3)$, $X = \text{tridiag}(-3, 1, 4)$. *We compare the efficiency of our method (GIO) with another iterative methods such as GI, LS, RGI, MGI, JGI and AJGI. We choose the same convergent factor* $\tau = 0.01836$ *and the same initial matrix* $X(0) = \text{tridiag}(0, 10^{-6}, 0)$. *To compare the efficiency of these methods, we fix the iteration number to be* 50 *and consider the relative errors* $\|E(k)\|_F / \|F\|_F$. *The results are displayed in Figure 3. The iteration numbers and the computational times when we fix the error* $\|E(k)\|_F$ *to be less than* 5×10^{-3} *are illustrated in Table 3. We see that our method is outperform to the direct method and another iterative methods with less iteration number and lower computational time. In particular, the approximated solutions generated from JGI method diverge.*

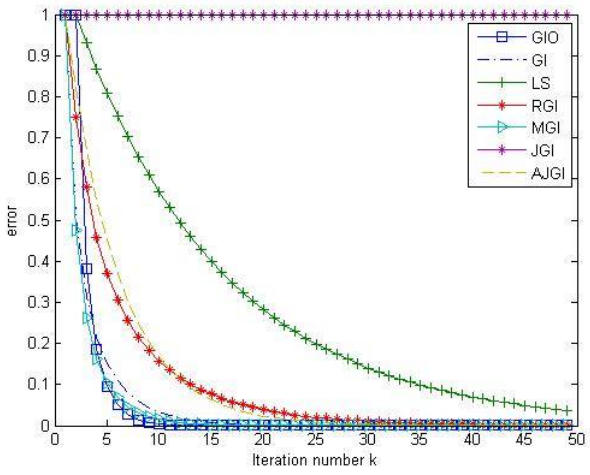

Figure 3. Relative errors for Example 3.

Table 3. Iteration numbers and computational times for Example 3.

Method	GIO	GI	LS	RGI	MGI	JGI	AJGI	Direct
IT	18	33	167	70	25	-	51	-
CT	0.000273	0.000589	0.0114	0.0012	0.000789	-	0.0014	0.1704

6. An Application to Discretization of the Convection-Diffusion Equation

In this section, we apply the GIO method to a discretization of convection–diffusion equation in the form

$$\frac{\partial u}{\partial t} + \mu \frac{\partial u}{\partial x} = \alpha \frac{\partial^2 u}{\partial x^2} \quad \text{for} \quad c \leq x \leq d \quad \text{and} \quad 0 \leq t \leq L \tag{22}$$

where μ and α are the convection and diffusion coefficients, respectively. Equation (22) is accompanied by the initial condition $u(x,0) = f(x)$ and boundary conditions $u(c,t) = g(t), u(d,t) = h(t)$ where f, g, h are given functions. To make a discretization of Equation (22), we divide $[c,d]$ into M subintervals, each of equal length $h = (d-c)/M$. In the same manner, we define a grid for the N subintervals $l = L/N$. Then we make discretization at the grid point $u_m^n = u(x_m, t_n)$ where

$$x_m = c + mh \quad \text{and} \quad t_n = nl \tag{23}$$

for $1 \leq m \leq M$ and $1 \leq n \leq N$. By applying the forward time central space method, we have

$$(\frac{u_m^{n+1} - u_m^n}{l}) + \mu(\frac{u_{m+1}^n - u_{m-1}^n}{2h}) = \alpha(\frac{u_{m-1}^n - 2u_m^n + u_{m+1}^n}{h^2}).$$

Rearranging the above equation leads to

$$u_m^{n+1} = (p + \frac{1}{2}r)u_{m-1}^n + (1-2p)u_m^n + (p - \frac{1}{2}r)u_{m+1}^n$$

where $r = \mu l/n$ and $p = \alpha l/h^2$ are the convection and diffusion numbers, respectively. We can transform (22) into a linear system of MN unknowns $u_{11}, ..., u_{MN}$ in the form

$$P_{CD}\,\mathrm{vec}(U) = b, \tag{24}$$

where $U = [u_m^n]$, $P_{CD} \in \mathbb{R}^{M\times N}$ has $N \times N$ blocks of the form I_M on its diagonal and tridiag$(-p - \frac{1}{2}r, -1+2p, -p+\frac{1}{2}r)$ under its diagonal. The vector b is partitioned in M blocks as $[b_1^T \;\; b_2^T \;\; \dots \;\; b_N^T]^T$ where $b_1 = \begin{bmatrix}\phi(1) & \phi(2) & \cdots & \phi(m-1)\end{bmatrix}^T$ and

$$b_j = \begin{bmatrix} (p+\frac{1}{2}r)g(t+(i-1)l) \\ 0 \\ \vdots \\ 0 \\ (p-\frac{1}{2}r)h(t+(i-1)l) \end{bmatrix}, \quad j = 2, \dots, N \tag{25}$$

here $\phi(i) = (p+\frac{1}{2}r)f(c+(i-1)h) + (1-2p)f(c+ih) + (p-\frac{1}{2}r)f(c+(i+1)h)$

We can see that Equation (24) is the generalized Sylvester equation where $p = 1$, $A = P_{CD}$, $X = \mathrm{vec}(U)$, $B = I$ and $F = b$. From Method 3, we obtain the following:

Method 5. *Input $M, N \in \mathbb{N}$ as number of partition. Set $P'_{CD} = P_{CD}^T$.*

Initializing step: Choose $u(0) \in \mathbb{R}^{MN}$. *For each* $m = 1, 2, \ldots, M$ *and* $n = 1, 2, \ldots, N$, *compute* x_m, t_n *as in Equation* (23) *and*

$$\tau_{opt} = \frac{2}{\lambda_{\max}(P_{CD}^T P_{CD}) + \lambda_{\min}(P_{CD}^T P_{CD})}.$$

Updating step: For $k = 1$ *to end, do:*

$$E(k-1) = b - P_{CD}u(k-1),$$
$$u(k) = \frac{1}{p}\sum_{i=1}^{p}(u(k-1) + \tau_{opt}P'_{CD}E(k-1)).$$

To stop the method, one may impose a stopping rule such as $\|E(k)\|_F / \|b\|_F < \epsilon$ *where* ϵ *is a tolerance error.*

Now, we provide a numerical experiment for a convection-diffusion equation.

Example 4. *Consider the convection–diffusion equation*

$$\frac{\partial u}{\partial t} + 0.1\frac{\partial u}{\partial x} = 0.01\frac{\partial^2 u}{\partial x^2} \quad for \quad 0 \le x \le 1 \quad and \quad 0 \le t \le 10 \tag{26}$$

with the initial and boundary conditions given as:

$$u(x, 0) = 100x \quad and \quad u(0, t) = u(1, t) = 0.$$

Let $M = 5, N = 10$, *so that* P_{CD} *is of dimension* 50×50. *In this case, we have* $h = 0.2, l = 1, r = 0.5$ *and* $p = 0.25$. *We choose* $u(0) = 10^{-6}\left[1 \cdots 1\right] \in \mathbb{R}^{50}$.

After compiling Method 5 for 100 iterations, we see from Figure 4 that the relative error $\|E(k)\|_F$ *goes faster to* 0 *than for other methods such as GI, LS, RGI, MGI, JGI and AJGI. Moreover, Table 4 displays comparison of numerical and direct solutions for the convection–diffusion equation.*

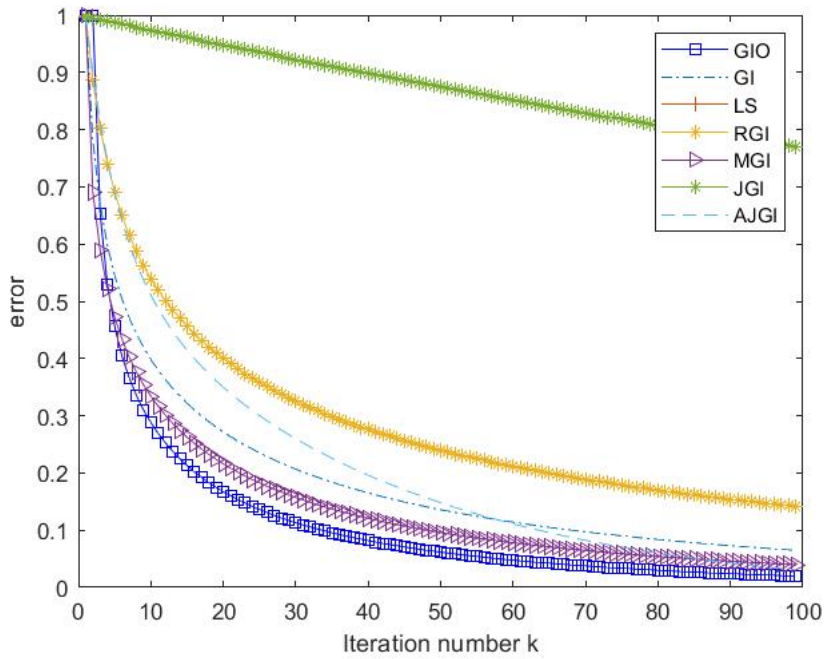

Figure 4. Relative errors for Example 4.

Table 4. Iteration numbers, computational times and errors for Example 4.

Method	IT	CT	Error
Direct	-	2.085	0
GIO	100	0.0113	0.0199
GI	100	0.0281	0.0648
LS	100	0.0469	1.6574
RGI	100	0.0324	0.1417
MGI	100	0.0313	0.0397
JGI	100	0.2813	0.7698
AJGI	100	0.0938	0.0307

A particular case $\mu = 0$ of Equation (22) is called the diffusion equation. In this case, the formulas of P_{CD} and $b_1, \ldots, b_N$ are reduced as $r = 0$.

Example 5. *Consider the diffusion equation*

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \quad for \quad 0 \leq x \leq 1 \quad and \quad 0 \leq t \leq 10$$

with the initial and boundary conditions given as:

$$u(x,0) = 6\sin(\pi x) \quad and \quad u(0,t) = u(1,t) = 0. \tag{27}$$

The exact solution is

$$u^*(x,t) = 6e^{-\pi^2 t}\sin(\pi x).$$

Let $M = 10, l = 0.01$ *In this case, we have* $h = 0.1$, *and* $p = 1$. *We choose initial matrix* $u(0) = 10^{-6}\left[1 \cdots 1\right] \in \mathbb{R}^{100}$.

After compiling Method 5 for 200 iterations (Figure 5), we see that our method is outperform to another iterative methods with less iteration number and lower computational time. The 3D-plot in Figure 6 shows that the iterative solution is well approximated to the exact solution.

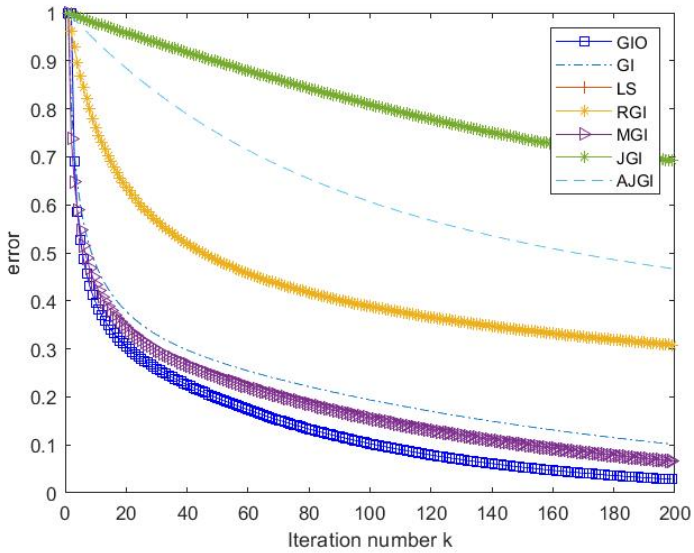

Figure 5. Relative errors for Example 5.

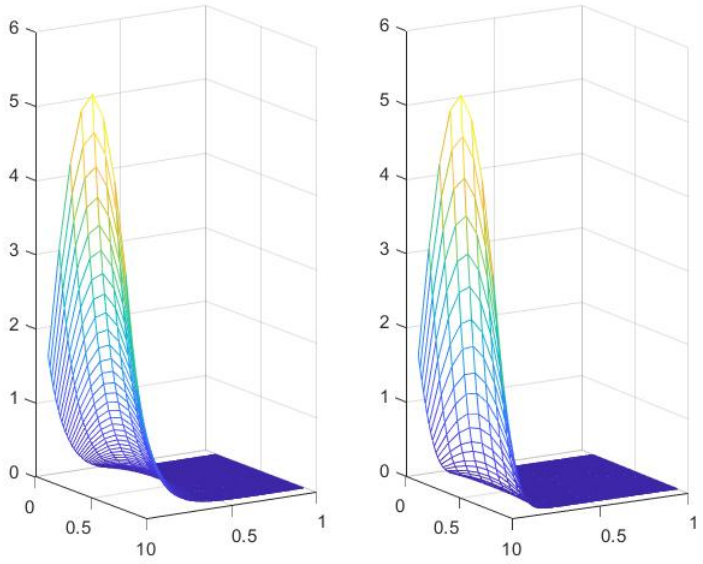

Figure 6. The exact (**left**) and the iterative (**right**) solutions for Example 5.

7. Conclusions

We propose a gradient-based iterative method with an optimal convergent factor for solving a generalized Sylvester matrix equation. The convergence analysis reveals that the sequence of approximated solutions converge to the exact solution for any initial value if and only if the convergent factor is chosen properly. The convergent rate and error estimations depend on the spectral radius of the associated iteration matrix. Moreover, we obtain the fastest convergent factor so that the associated iteration matrix has the smallest spectral radius. Furthermore, the proposed algorithm is applicable for the discretization of the diffusion equations. The numerical experiments illustrate that our method is applicable for any conformable square/rectangular matrices of small/large sizes. Moreover, they reveal that our method performs well comparing to recent iterative methods.

Author Contributions: N.B. and P.C. contributed equally and significantly in writing this article. All authors read and approved the final manuscript.

Funding: The first author would like to thank Science Achievement Scholarship of Thailand (SAST), Grant No. 01/2560, from Ministry of Education for financial support during the Ph.D. study.

Acknowledgments: This work was supported by Ministry of Education, Thailand.

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

References

1. Benner, P. Factorized solution of Sylvester equations with application in control. In *Theory Networks and System*; International Symposium of Mathematics: Berlin, Germany, 2014.
2. Tsui, C.C. On robust observer compensator design. *Automatica* **1988**, *24*, 687–692. [CrossRef]
3. Van Dooren, P. Reduce order observer: A new algorithm and proof. *Syst. Control Lett.* **1984**, *4*, 243–251. [CrossRef]
4. Bartels, R.; Stewart, G. Solution of the matrix equation $AX + XB = C$. *Circuits Syst. Signal Process.* **1994**, *13*, 820–826. [CrossRef]
5. Sadeghi, A. A new approach for computing the solution of Sylvester matrix equations. *J. Interpolat. Approx. Sci. Comput.* **2016**, *2*, 66–76. [CrossRef]
6. Li, S.-Y.; Shen, H.-L.; Shao, X.-H. PHSS iterative method for solving generalized Lyapunov equation. *Mathematics* **2019**, *7*, 38. [CrossRef]

7. Shen, H.-L.; Li, Y.-R.; Shao, X.-H. The four-parameter PSS method for solving the Sylvester equation. *Mathematics* **2019**, *7*, 105. [CrossRef]
8. Ding, F.; Chen, T. Hierarchical gradient-based identification methods for multivariable discrete time systems. *Automatica* **2005**, *41*, 397–402. [CrossRef]
9. Jonsson, I.; Kagstrom, B. Recursive blocked algorithms for solving triangular system Part I: One-side and coupled Sylvester-type matrix equation. *ACM Trans. Math. Softw.* **2002**, *28*, 392–415. [CrossRef]
10. Zhang, H.M.; Ding, F. A property of the eigenvalues of the symmetric positive definite matrix and the iterative algorithm for coupled Sylvester matrix equations. *J. Frankl. Inst.* **2014**, *351*, 340–357. [CrossRef]
11. Wang, X.; Li, Y.; Dai, L. On the Hermitian and skew-Hermitian splitting iteration methods for the linear matrix equation $AXB = C$. *Comput. Math. Appl.* **2013**, *65*, 657–664. [CrossRef]
12. Zhu, M.Z.; Zhang, G.F. A class of iteration methods based on the HSS for Toeplitz system of weakly nonlinear equation. *Comput. Appl. Math.* **2015**, *290*, 433–444. [CrossRef]
13. Bai, Z.Z. On Hermitian and skew-Hermitian splitting iteration method for continuous Sylvester equation. *J. Comput. Math.* **2011**, *29*, 185–198. [CrossRef]
14. Zheng, Q.Q.; Ma, C.F. On normal and skew-Hermitian splitting iteration methods for large sparse continuous Sylvester equation. *J. Comp. Appl. Math.* **2014**, *268*, 145–154. [CrossRef]
15. Ding, F.; Chen, T. Gradient based iterative algorithms for solving a class of matrix equation. *IEEE Trans. Autom. Control* **2005**, *50*, 1216–1221. [CrossRef]
16. Ding, F.; Chen, T. Iterative least square solutions of coupled Sylvester matrix equation. *Syst. Control Lett.* **2005**, *54*, 95–107. [CrossRef]
17. Ding, F.; Liu, X.P.; Ding, J. Iterative solution of the generalized Sylvester matrix equations by using the hierarchical identification principle. *Appl. Math. Comput.* **2008**, *197*, 41–50. [CrossRef]
18. Nui, Q.; Wang, X.; Lu, L.-Z. A relaxed gradient based iterative algorithms for solving Sylvester equation. *Asian J. Cont.* **2011**, *13*, 461–464.
19. Xie, Y.; Ma, C.F. The accelerated gradient based iterative algorithm for solving a class of generalized Sylvester-transpose matrix equation. *Appl. Math. Comput.* **2016**, *273*, 1257–1269. [CrossRef]
20. Fan, W.; Gu, C.; Tian, Z. Jacobi-gradient iterative algorithms for Sylvester matrix equations. In *Linear Algebra Society Topics*; Shanghai University: Shanghai, China, 2007; pp. 16–20.
21. Li, S.K.; Huang, T.Z. A shift-splitting Jacobi-gradient algorithm for Lyapunov matrix equation arising form control theory. *J. Comput. Anal. Appl.* **2011**, *13*, 1246–1257.
22. Tian, Z.; Tian, M.; Gu, C.; Hao, X. An accelerated Jacobi-gradient based iterative algorithm for solving Sylvester matrix equation. *Filomat* **2017**, *31*, 2381–2390. [CrossRef]