On Preconditioned Normal and Skew-Hermitian Splitting Iteration Method for Continuous Sylvester Equations $AX + XB = C^\ast$

Xue-Qing Liang\textsuperscript{a}, Xiang Wang\textsuperscript{a,b,\ast}, Xiao-Bin Tang\textsuperscript{c}, Xiao-Yong Xiao\textsuperscript{a,b}

\textsuperscript{a}Department of Mathematics, School of Sciences, Nanchang University, Nanchang 330031, P. R. China
\textsuperscript{b}Numerical Simulation and High-Performance Computing Laboratory, School of Sciences, Nanchang University, Nanchang 330031, P. R. China
\textsuperscript{c}School of Statistics, University of International Business and Economics, Beijing 100029, P. R. China

Abstract. In this paper, we present a preconditioned normal and skew-Hermitian splitting (PNSS) iteration method for continuous Sylvester equations $AX + XB = C$ with positive definite/semi-definite matrices. Theoretical analysis shows that the PNSS methods will converge unconditionally to the exact solution of the continuous Sylvester equations. An inexact variant of the PNSS iteration method (IPNSS) and the analysis of its convergence property in detail have been established. Numerical experiments further show that this new method is more efficient and robust than the existing ones.

1. Introduction

In this paper, we consider the iteration solution of the following continuous Sylvester equations:

$$AX + XB = C,$$

(1)

where $A \in \mathbb{C}^{m \times m}$, $B \in \mathbb{C}^{n \times n}$ and $C \in \mathbb{C}^{m \times n}$. Assume that

(A1) $A$, $B$, $C$ are large and sparse matrices;

(A2) at least one of $A$ and $B$ is non-Hermitian;

(A3) both $A$ and $B$ are positive semi-definite, and at least one of them is positive definite.

The continuous Sylvester equations (1) has a unique solution, under the assumption (A1–A3) that there is no common eigenvalue between $A$ and $-B$. Here and in the sequel, $A^\ast$ represents the conjugate transpose of the matrix $A$. In [1], many important theoretical results about this kind of equations can be found. The continuous Sylvester equations (1) plays an important role in many fields, such as system theory [30, 33, 34], model reduction [2, 35, 38], power systems [26], matrix nearness problem [31], numerical solution of differential equations [3–5, 28, 48–51], finite element model updating [21], noisy image restoration [17], stability of linear systems [24] and so on.

Keywords. Continuous Sylvester equations; PNSS iteration method; IPNSS iteration method; convergence.

2010 Mathematics Subject Classification. 65F10, 65F50

Received: 03 January 2017; Accepted: 09 October 2017

Communicated by Dragana Cvetković Ilić

This work is supported by NNSF of China No. 11461046, NSF of Jiangxi Province with Nos.20181ACB20001,20161ACB20005, and 20171BAB211006, and the Fundamental Research Funds for the Central Universities in UIBE (CXTD9-07).

Corresponding author: wangxiang49@ncu.edu.cn, TEL: +86 791 83969509

Email address: wangxiang49@ncu.edu.cn (Xiang Wang)
The continuous Sylvester equation (1) is mathematically equivalent to the system of linear equations

$$Ax = c,$$

where $A = I \otimes A + B^T \otimes I$, and the vectors $x$ and $c$ contain the concatenated columns of the matrices $X$ and $C$ respectively, with $\otimes$ being the Kronecker product symbol, and $B^T$ representing the transpose of the matrix $B$. However, it is expensive and ill-conditioned to use the iteration method for solving the system of linear equations (2).

There are a large number of numerical methods for solving the continuous Sylvester equations (1). The Bartels-Stewart and the Hessenberg-Schur methods [23] are direct methods, which consist in transforming $A$ and $B$ into triangular or Hessenberg-Schur form by an orthogonal similarity transformation and then solving the resulting system of linear equations directly by a back-substitution process. However, they are suitable for small-scale settings, and not applicable in large-scale settings. When the matrices $A$ and $B$ are large and sparse, iterative methods can solve the continuous Sylvester equation (1) efficiently and accurately with its sparsity and low rank structure. The two most common iterative methods are Alternating Direction Implicit (ADI) method [6, 7, 20], gradient based algorithms [32, 37] and the Krylov subspace based algorithms [8, 18, 25, 27, 36].

The HSS iteration method was firstly proposed by Bai, Golub and Ng in [9] for non-Hermitian positive definite linear systems. Then the method was extended to other equations and conditions in [10–15, 39, 40, 42–45]. In [16], Bai presented the Hermitian and skew-Hermitian splitting (HSS) iteration method for solving large sparse continuous Sylvester equations. Wang et al. applied the idea of PSS iteration method to solve the continuous Sylvester equations in [41], Zheng and Ma applied the NSS iteration method to solve the continuous Sylvester equations in [46]. Recently, Dong and Gu presented a PMHSS iteration method [19] for Sylvester equations. Zhou and Wang presented a PPSS iteration method [47] for Sylvester equations. Thus, motivated by this, we further present and analyze a preconditioned normal and skew-Hermitian iteration method (PNSS) for solving the continuous Sylvester equations.

The rest of the paper is organized as follows. In Section 2, after a brief introduction of the NSS iteration method, we present the PNSS iteration method for solving the continuous Sylvester equation (1) and analyze the convergence property of the PNSS iteration method. In Section 3, we establish an inexact preconditioned normal and skew-Hermitian iteration method (IPNSS) iteration for solving the continuous Sylvester equation (1). In Section 4, some numerical examples are presented to illustrate the efficiency of the PNSS method.

In the remainder of this paper, a matrix sequence $\{Y^{(k)}\}_{k=0}^{\infty} \subseteq \mathbb{C}^{m \times n}$ is said to be convergent to a matrix $Y \in \mathbb{C}^{m \times n}$ if the corresponding vector sequence $\{y^{(k)}\}_{k=0}^{\infty} \subseteq \mathbb{C}^{m \times n}$ is convergent to the corresponding vector $y \in \mathbb{C}^{m \times n}$, where the vectors $y^{(k)}$ and $y$ contain the concatenated columns of the matrices $Y^{(k)}$ and $Y$, respectively. If $\{Y^{(k)}\}_{k=0}^{\infty}$ is convergent, then its convergence factor and convergence rate are defined as those of $\{y^{(k)}\}_{k=0}^{\infty}$ correspondingly. In addition, we use $sp()$, $\|\cdot\|_2$, and $\|\cdot\|_F$ to denote the spectrum, the spectral norm, and the Frobenius norm, respectively. Note $\|\cdot\|_2$ is also used to represent the 2-norm of a vector.

2. The PNSS iteration method

In this section, we consider the scheme of PNSS iteration method and its convergence property. This iteration method is with inner and outer iterations while each step of the inner iteration is exactly computed by direct methods.

Firstly, we split $A$ and $B$ into normal and skew-Hermitian parts

$$A = N(A) + S(A), \quad B = N(B) + S(B).$$

Then $A$ and $B$ can be rewritten as

$$A = (aI + N(A)) + (S(A) - aI) = (aI + S(A)) + (N(A) - aI),$$
$$B = (\beta I + N(B)) + (S(B) - \beta I) = (\beta I + S(B)) + (N(B) - \beta I),$$
where \( \alpha, \beta \) be positive real numbers and \( I \) is the identity matrix of suitable dimension. It follows that the continuous Sylvester equations (1) can be equivalently written as follows:

\[
\begin{align*}
(aI + N(A))X + X(\beta I + N(B)) &= (aI - S(A))X + X(\beta I - S(B)) + C, \\
(aI + S(A))X + X(\beta I + S(B)) &= (aI - N(A))X + X(\beta I - N(B)) + C.
\end{align*}
\]

Then we can easily establish the following normal and skew-Hermitian splitting iteration method.

**The NSS iteration method**

Given an initial guess \( X^{(0)} \in \mathbb{C}^{m \times n} \), compute \( X^{(k+1)} \in \mathbb{C}^{m \times n} \) for \( k = 0, 1, 2, \cdots \), using the following iteration procedure until \( \{X^{(k)}\}_{k=0}^{\infty} \) satisfies the stopping criterion:

\[
\begin{align*}
(aI + N(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\beta I + N(B)) &= (aI - S(A))X^{(k)} + X^{(k)}(\beta I - S(B)) + C, \\
(aI + S(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\beta I + S(B)) &= (aI - N(A))X^{(k)} + X^{(k)}(\beta I - N(B)) + C,
\end{align*}
\]

where \( \alpha, \beta \) be positive real numbers and \( I \) is the identity matrix of suitable dimension.

**The PNSS iteration method**

Given an initial guess \( X^{(0)} \in \mathbb{C}^{m \times n} \), compute \( X^{(k+1)} \in \mathbb{C}^{m \times n} \) for \( k = 0, 1, 2, \cdots \), using the following iteration procedure until \( \{X^{(k)}\}_{k=0}^{\infty} \) satisfies the stopping criterion:

\[
\begin{align*}
(aV_1 + N(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\beta V_2 + N(B)) &= (aV_1 - S(A))X^{(k)} + X^{(k)}(\beta V_2 - S(B)) + C, \\
(aV_1 + S(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\beta V_2 + S(B)) &= (aV_1 - N(A))X^{(k)} + X^{(k)}(\beta V_2 - N(B)) + C,
\end{align*}
\]

where \( \alpha, \beta \) be two positive real numbers and \( V_1, V_2 \) be two prescribed symmetric positive definite matrices.

Under the assumptions \( (A_1, A_3) \), we can easily know that there is no common eigenvalue between the matrices \( aV_1 + N(A) \) and \( -\beta V_2 + N(B) \), as well as between the matrices \( aV_1 + S(A) \) and \( -\beta V_2 + S(B) \), so that the two fixed-point matrix equations have unique solutions for all given right-hand side matrices. Naturally, the two half-steps involved in each step of the PNSS iteration method can be solved effectively using mostly real arithmetic. It is clear that the PNSS iteration method reduces to NSS iteration method with \( V_1 = I_m, V_2 = I_n \), where \( I_m, I_n \) are the identity matrices of order \( m \) and \( n \) respectively. In particular, when \( \alpha = \beta \), we have

\[
\begin{align*}
(aV_1 + N(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(aV_2 + N(B)) &= (aV_1 - S(A))X^{(k)} + X^{(k)}(aV_2 - S(B)) + C, \\
(aV_1 + S(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(aV_2 + S(B)) &= (aV_1 - N(A))X^{(k)} + X^{(k)}(aV_2 - N(B)) + C.
\end{align*}
\]

By making use of the Kronecker product, it can be rewritten the above-described PNSS iteration method in the following matrix-vector form:

\[
\begin{align*}
(I \otimes (aV_1 + N(A)) + (aV_2 + N(B))^T \otimes I)vec(X^{(k+\frac{1}{2})}) &= (I \otimes (aV_1 - S(A)) + (aV_2 - S(B))^T \otimes I)vec(X^{(k)}) + vec(C), \\
(I \otimes (aV_1 + S(A)) + (aV_2 + S(B))^T \otimes I)vec(X^{(k+\frac{1}{2})}) &= (I \otimes (aV_1 - N(A)) + (aV_2 - N(B))^T \otimes I)vec(X^{(k+\frac{1}{2})}) + vec(C).
\end{align*}
\]

Denote by \( A = N + S \), with

\[ N = I \otimes N(A) + N(B)^T \otimes I, \quad S = I \otimes S(A) + S(B)^T \otimes I \]

and

\[ K(\alpha) = I \otimes (aV_1) + (aV_2)^T \otimes I = \alpha K. \]
Then we obtain

\[
\begin{cases}
(aK + N)\text{vec}(X^{(k+\frac{1}{2})}) = (aK - S)\text{vec}(X^{(k)}) + \text{vec}(C), \\
(aK + S)\text{vec}(X^{(k+1)}) = (aK - N)\text{vec}(X^{(k+\frac{1}{2})}) + \text{vec}(C).
\end{cases}
\]

Evidently, the iteration scheme is the PNSS iteration method for solving the system of linear equation (2). Then after concrete operations, we can obtain:

\[
\text{vec}(X^{(k+1)}) = M(a)\text{vec}(X^{(k)}) + G(a)\text{vec}(C),
\]

where

\[
M(a) = (aK + S)^{-1}(aK - N)(aK + N)^{-1}(aK - S),
\]

and

\[
G(a) = (aK + S)^{-1}(I + (aK - N)(aK + N)^{-1}).
\]

In addition, if we introduce matrices

\[
\mathcal{F}_1(a) = \frac{1}{2a}(aK + N)K^{-1}(aK + S)
\]

and

\[
\mathcal{G}_1(a) = \frac{1}{2a}(aK - N)K^{-1}(aK - S),
\]

then it holds that

\[A = \mathcal{F}_1(a) - \mathcal{G}_1(a), \quad M(a) = \mathcal{F}_1(a)^{-1}\mathcal{G}_1(a).\]

In the following, to study the convergence property of the PNSS method. The following lemmas are required.

**Lemma 1**[15] Let \(\tilde{S} \in C^{m \times mn}\) be a skew-Hermitian matrix. Then for any \(a > 0\), the Cayley transform \(Q(a) = (al - \tilde{S})(al + \tilde{S})^{-1}\) of \(\tilde{S}\) is an unitary matrix.

**Lemma 2**[46] If \(A \in C^{m \times m}, B \in C^{n \times n}, C \in C^{m \times n}\), let \(A = M_i(A) - N_i(A)(i = 1, 2), B = M_i(B) - N_i(B)(i = 1, 2)\) be two splittings of the matrices \(A, B\), respectively. Let \(X^{(0)} \in C^{m \times n}\) be a given initial matrix, if \(X^{(0)}\) is a two-step iteration sequence defined by

\[
\begin{cases}
M_1(A)X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}M_2(B) = N_1(A)X^{(k)} + X^{(k)}N_2(B) + C, \\
M_2(A)X^{(k+1)} + X^{(k+1)}M_2(B) = N_2(A)X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}N_2(B) + C,
\end{cases}
\]

\(k = 0, 1, 2, \ldots, x^{(k+1)} = \text{vec}(X^{(k+1)}), x^{(k+\frac{1}{2})} = \text{vec}(X^{(k+\frac{1}{2})}), x^{(k)} = \text{vec}(X^{(k)}),\)

then we can obtain

\[x^{(k+1)} = Mx^k + g,\]

where

\[M = [I \otimes M_2(A) + M_2(B)^T \otimes I]^{-1}[I \otimes N_2(A) + N_2(B) \otimes I] \times [I \otimes N_1(A) + M_1(B)^T \otimes I]^{-1}[I \otimes N_1(A) + N_1(B) \otimes I]\]

and

\[g = [I \otimes M_2(A) + M_2(B)^T \otimes I]^{-1}[I + (I \otimes N_2(A) + N_2(B) \otimes I)(I \otimes M_1(A) + M_1(B)^T \otimes I)]^{-1}\text{vec}(F).\]

Moreover, if the spectral radius \(\rho(M)\) of the iteration matrix \(M\) is less than 1, i.e., \(\rho(M) < 1\), then the iterative sequence \(X^{(0)}_{\infty}\) converges to the unique solution of the linear matrix equation (1) for arbitrary initial matrix \(X^{(0)} \in C^{m \times n}\).

Concerning the convergence property of the PNSS method, we apply the above results to obtain the following theorem.

**Theorem 1** Assume that \(A \in C^{m \times m}, B \in C^{n \times n}\) are positive semi-definite matrices, and at least one of them is positive definite. Let \(N(A) \in C^{m \times n}, N(B) \in C^{n \times n}\) be two normal matrices and \(S(A) \in C^{m \times m}\),
\[
S(B) \in \mathbb{C}^{m \times n} \text{ be two skew-Hermitian matrices, such that } A = N(A) + S(A), B = N(B) + S(B). \text{ Denote by } \\
N = I \otimes N(A) + N(B)^T \otimes I, S = I \otimes S(A) + S(B)^T \otimes I, K(a) = I \otimes (AV_1) + (AV_2)^T \otimes I = a^2 A, \text{ where } a \text{ is a positive constant, } V_1 \in \mathbb{C}^{m \times m}, V_2 \in \mathbb{C}^{n \times n} \text{ are prescribed symmetric positive definite matrices. Noting that } \\
N = \mathbb{K}^{-\frac{1}{2}}N\mathbb{K}^{-\frac{1}{2}}, S = \mathbb{K}^{-\frac{1}{2}}S\mathbb{K}^{-\frac{1}{2}}, \text{ and } \\
M(a) = (a^2 A + S)^{-1}(a^2 K - N)(a^2 K + N)^{-1}(a^2 K - S).
\]

Then the convergence factor of the PNSS iteration method is given by the spectral radius \(\rho(M(a))\) of the matrix \(M(a)\), which is bounded by

\[
\sigma(a) = \max_{\lambda_i \in \lambda(M)} \left| \frac{\alpha - \lambda_i}{\alpha + \lambda_i} \right|.
\]

Therefore, it holds that

\[
\rho(M(a)) \leq \sigma(a) < 1, \quad \forall \alpha > 0.
\]

i.e. the PNSS iteration is convergent to the exact solution \(X^* \in \mathbb{C}^{m \times n}\) of the continuous Sylvester equations (1). Moreover, we denote \(a_j = \text{Re}(\lambda_j), b_j = \text{Im}(\lambda_j)\).

**Proof:** By putting

\[
M_1(A) = aV_1 + N(A), M_1(B) = aV_2 + N(B), N_1(A) = aV_1 - S(A), N_1(B) = aV_2 - S(B)
\]

and

\[
M_2(A) = aV_1 + S(A), M_2(B) = aV_2 + S(B), N_2(A) = aV_1 - N(A), N_2(B) = aV_2 - N(B).
\]

in Lemma 2, we obtain

\[
M(a) = (a^2 A + S)^{-1}(a^2 K - N)(a^2 K + N)^{-1}(a^2 K - S).
\]

We can easily verify that \(K\) is symmetric positive definite matrix, \(\tilde{N}\) is normal and \(\tilde{S}\) is skew-Hermitian. By the similarity invariance of the matrix spectrum, we have

\[
\rho(M(a)) = \rho((a^2 K - N)(a^2 K + N)^{-1}(a^2 K - S)(a^2 K + S)^{-1})
\]

\[
\leq \| (a^2 K - N)(a^2 K + N)^{-1}(a^2 K - S)(a^2 K + S)^{-1} \|_2
\]

\[
= \| K^{\frac{1}{2}}(a^2 K - N)K^{-\frac{1}{2}}K^{\frac{1}{2}}(a^2 K + N)^{-1}K^{-\frac{1}{2}}K^{\frac{1}{2}}(a^2 K - S)K^{-\frac{1}{2}}K^{\frac{1}{2}}K^{\frac{1}{2}}(a^2 K + S)^{-1}K^{\frac{1}{2}}K^{\frac{1}{2}} \|_2
\]

\[
= \| (al - \tilde{N})(al + \tilde{N})^{-1}(al - \tilde{S})(al + \tilde{S})^{-1} \|_2
\]

\[
= \| (al - \tilde{N})(al + \tilde{N})^{-1} \|_2 \| (al - \tilde{S})(al + \tilde{S})^{-1} \|_2
\]

Here we use the special property of the matrix \(Q = (al - \tilde{S})(al + \tilde{S})^{-1}\) which can be obtained from Lemma 1, i.e., \(Q\) is an unitary matrix with its spectral norm equal to 1 (\(Q\) is also called the Cayley transform of \(S\)). Hence, we can further get

\[
\rho(M(a)) \leq \sigma(a) = \max_{\lambda_i \in \lambda(M)} \left| \frac{\alpha - \lambda_i}{\alpha + \lambda_i} \right| \leq \max_j \sqrt{\frac{(\alpha - a_j)^2 + b_j^2}{(\alpha + a_j)^2 + b_j^2}}
\]

since the \(a_j = \text{Re}(\lambda_j) > 0, j = 1, 2, ..., n, \text{ and } \alpha \text{ is a positive constant, it is easy to see that}

\[
\rho(M(a)) \leq \sigma(a) < 1, \quad \forall \alpha > 0.
\]

Therefore the PNSS iteration method converges unconditionally to the exact solution \(X^* \in \mathbb{C}^{m \times n}\) of (1), with the convergence factor being \(\rho(M(a))\).
3. The inexact PNSS iteration method

The two-half steps at each step of the PNSS iteration method for solving the continuous Sylvester equation (1) require finding solutions of two continuous Sylvester equations

\[(aV_1 + N(A))X + X(aV_2 + N(B)) = C_S\]

and

\[(aV_1 + S(A))X + X(aV_2 + S(B)) = C_H\]

where \(C_S\) and \(C_H\) are prescribed \(m - b y - n\) complex matrices. However, this may be very costly and impractical in actual implementations, particularly when the sizes of the matrices involved are very large.

To further improve the computational efficiency of the PNSS iteration, we can solve the two subproblems inexactly by utilizing certain effective iteration methods such as Gauss-Seidel, SOR, ADI or Krylov subspace based methods, which results in the following inexact preconditioned normal and skew-Hermitian splitting iteration for solving the continuous Sylvester equation (1).

**The IPNSS iteration method**

Give an initial guess \(X^{(0)} \in C^{m \times n}\), for \(k = 0, 1, 2, \ldots\) until \(\{X^{(k)}\}_{k=0}^{\infty} \subseteq C^{m \times n}\) satisfies the stopping criterion, solve \(X^{(k+\frac{1}{2})} \in C^{m \times n}\) approximately from

\[(aV_1 + N(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(aV_2 + N(B)) \approx (aV_1 - S(A))X^{(k)} + X^{(k)}(aV_2 - S(B)) + C\]

by employing an inner iteration with \(X^{(k)}\) as the initial guess, then solve \(\{X^{(k+\frac{1}{2})}\}_{k=0}^{\infty} \subseteq C^{m \times n}\) approximately from

\[(aV_1 + S(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(aV_2 + S(B)) \approx (aV_1 - N(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(aV_2 - N(B)) + C\]

by employing an inner iteration with \(X^{(k+\frac{1}{2})}\) as the initial guess, where \(a\) is given positive constant.

To simplify numerical implementation and convergence analysis, we may rewrite the above iteration method as the following scheme.

**The IPNSS iteration method**

Give an initial guess \(X^{(0)} \in C^{m \times n}\), for \(k = 0, 1, 2, \ldots\) until \(\{X^{(k)}\}_{k=0}^{\infty} \subseteq C^{m \times n}\) converges:

Step 1. Approximate the solution of

\[(aV_1 + N(A))Z^{(k)} + Z^{(k)}(aV_2 + N(B)) = R^{(k)},\]

with \(R^{(k)} = C - AX^{(k)} - X^{(k)}B\), by iterating until \(Z^{(k)}\) is such that the residual

\[P^k = R^{(k)} - ((aV_1 + N(A))Z^{(k)} + Z^{(k)}(aV_2 + N(B))),\]

satisfies

\[\|P^k\|_F \leq \varepsilon_k\|R^{(k)}\|_F.\]

Then compute

\[X^{(k+\frac{1}{2})} = X^{(k)} + Z^{(k)};\]

Step 2. Approximate the solution of

\[(aV_1 + S(A))Z^{(k+\frac{1}{2})} + Z^{(k+\frac{1}{2})}(aV_2 + S(B)) = R^{(k+\frac{1}{2})},\]

with \(R^{(k+\frac{1}{2})} = C - AX^{(k+\frac{1}{2})} - X^{(k+\frac{1}{2})}B\), by iterating until \(Z^{(k+\frac{1}{2})}\) is such that the residual

\[Q^{(k+\frac{1}{2})} = R^{(k+\frac{1}{2})} - ((aV_1 + S(A))Z^{(k+\frac{1}{2})} + Z^{(k+\frac{1}{2})}(aV_2 + S(B))),\]

satisfies

\[\|Q^{(k+\frac{1}{2})}\|_F \leq \eta_k\|R^{(k+\frac{1}{2})}\|_F.\]
Then compute
\[ X^{(k+1)} = X^{(k+\frac{1}{2})} + Z^{(k+\frac{1}{2})}. \]

Here, \( \varepsilon_k \) and \( \eta_k \) are prescribed tolerances which are used to control the accuracies of the inner iterations.

**Theorem 2** Assume that the assumptions of Theorem 1 hold. If \( \{X^{(k)}\}_{k=0}^{\infty} \subseteq \mathbb{C}^{m \times n} \) is the iteration sequence generated by the IPNSS iteration method and if \( X^* \in \mathbb{C}^{m \times n} \) is the exact solution of the continuous Sylvester equation (1), then we have
\[ \| X^{(k+1)} - X^* \|_S \leq (\sigma(\alpha) + \theta \eta_k) (1 + \theta \varepsilon_k) \| X^{(k)} - X^* \|_S, \]
where the norm \( \| \cdot \|_S \) is defined as \( \| Y \|_S = \| (\alpha V_1 + S(A))Y + Y(\alpha V_1 + S(B)) \|_F \), for any matrix \( Y \in \mathbb{C}^{m \times n} \), and the constants \( \varrho \) and \( \theta \) are given by
\[ \varrho = \|(\alpha K + S)(\alpha K + N)^{-1}\|_2, \]
\[ \theta = \|A(\alpha K + S)^{-1}\|_2, \]
\[ \sigma(\alpha) = \|(\alpha I - \tilde{N})(\alpha I + \tilde{N})^{-1}\|_2. \]

In particular, if \( (\sigma(\alpha) + \theta \eta_{\max})(1 + \theta \varepsilon) < 1 \), then the iteration sequence \( \{X^{(k)}\}_{k=0}^{\infty} \subseteq \mathbb{C}^{m \times n} \) converges to \( X^* \in \mathbb{C}^{m \times n} \), where \( \varepsilon_{\max} = \max_k \varepsilon_k \) and \( \eta_{\max} = \max_k \eta_k \).

**Proof:** The conclusion is straightforward according to Theorem 3.1 in [16].

**Theorem 3** Assume that the conditions of Theorem 1 hold. Suppose that both \( r_1(k) \) and \( r_2(k) \) are nonincreasing and positive sequence satisfying \( r_1(k) \geq 1 \) and \( r_2(k) \geq 1 \), and \( \limsup_{k \to \infty} r_1(k) = \limsup_{k \to \infty} r_2(k) = +\infty \), and that both \( \min(\delta_1, \delta_2) \) are real constants in the interval (0, 1) satisfying
\[ \varepsilon_k \leq c_1 \delta_1^{r_1(k)} \quad \text{and} \quad \eta_k \leq c_2 \delta_2^{r_2(k)}, \quad k = 0, 1, 2, \ldots \]
where \( c_1 \) and \( c_2 \) are nonnegative constants. Then it holds that
\[ \| X^{(k+1)} - X^* \|_S \leq (\sqrt{\sigma(\alpha) + \varphi \delta^{(k)}}) \| X^{(k)} - X^* \|_S, \]
where \( \varrho \) and \( \theta \) are defined in the above theorem and \( r(k) \) and \( \delta \) are defined as
\[ r(k) = \min\{r_1(k), r_2(k)\}, \quad \delta = \max\{\delta_1, \delta_2\}, \]
and
\[ \varphi = \max\{\sqrt{c_1 c_2 \theta}, \frac{1}{2 \sqrt{\sigma(\alpha)}} (c_1 \sigma(\alpha) + c_2 \theta)\}. \]

**Proof:** The conclusion is straightforward according to Theorem 3.2 in [16].

4. Numerical examples

In this section, we use several examples to illustrate the effectiveness of the PNSS iteration method for solving the continuous Sylvester equation \( AX + XB = C \). In addition, the numerical experiments are performed in Matlab (R2010b) on an Inter dual core processor (1.90GHz, 4GB RAM). All iterations of this section are started from zero matrix and terminated when the current residual norm satisfies
\[ \| R^{(k)} \|_F \leq 10^{-6}, \]
where \( R^{(k)} = C - AX^k - X^kB \). The number of iteration steps (denoted as “IT”) and the computing time in seconds (denoted as “CPU”) are listed in tables. In the PNSS iteration method for the continuous
Sylvester equation (1), \( N(A) \) and \( N(B) \) are two circulant matrices. We set the preconditioning matrices \( V_1 = \text{diag}(N(A)), V_2 = \text{diag}(N(B)) \).

**Example 1.** In this experiment, We consider the continuous Sylvester equation (1) with \( m = n \) and the matrices

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

where \( M, N \in \mathbb{C}^{n \times n} \) are the tridiagonal matrices given by

\[
M = \begin{pmatrix}
2.6 & -1 & & & \\
1 & 2.6 & -1 & & \\
& \ddots & \ddots & \ddots & \\
& & 1 & 2.6 & -1 \\
& & & 1 & 2.6
\end{pmatrix},
\]

\[
N = \begin{pmatrix}
0 & -0.5 & & & \\
0.5 & 0 & -0.5 & & \\
& \ddots & \ddots & \ddots & \\
& & 0.5 & 0 & -0.5 \\
& & & 0.5 & 0
\end{pmatrix}.
\]

This class of problems arise frequently in the preconditioned Krylov subspace iteration methods. We will solve this continuous Sylvester equation by the PNSS and the NSS iteration methods. The computing results of PNSS iteration method and the NSS iteration method are listed in Table 1, respectively. We compare the iteration steps and the computing time in seconds of both methods. We also present the optimal parameters \( \alpha_{\text{exp}} \) for the PNSS iteration method and the NSS iteration method in Table 2. From the results in Table 1, we observe that the PNSS is much better than the NSS both in terms of the number of iteration steps and computing time.
Table 2: The optimal values $\alpha_{\exp}$ for PNSS and NSS

<table>
<thead>
<tr>
<th>n</th>
<th>PNSS r=0.01</th>
<th>PNSS r=0.1</th>
<th>PNSS r=1</th>
<th>NSS r=0.01</th>
<th>NSS r=0.1</th>
<th>NSS r=1</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>0.68</td>
<td>0.68</td>
<td>0.82</td>
<td>1.99</td>
<td>1.97</td>
<td>1.93</td>
</tr>
<tr>
<td>16</td>
<td>0.65</td>
<td>0.62</td>
<td>0.66</td>
<td>3.00</td>
<td>2.91</td>
<td>2.82</td>
</tr>
<tr>
<td>32</td>
<td>0.79</td>
<td>0.69</td>
<td>0.56</td>
<td>3.73</td>
<td>3.20</td>
<td>3.02</td>
</tr>
<tr>
<td>64</td>
<td>0.92</td>
<td>0.87</td>
<td>0.42</td>
<td>4.18</td>
<td>3.37</td>
<td>3.23</td>
</tr>
<tr>
<td>128</td>
<td>1.26</td>
<td>1.20</td>
<td>0.30</td>
<td>5.52</td>
<td>5.10</td>
<td>4.91</td>
</tr>
<tr>
<td>256</td>
<td>1.52</td>
<td>1.41</td>
<td>0.23</td>
<td>6.85</td>
<td>6.39</td>
<td>6.10</td>
</tr>
</tbody>
</table>

Table 3: IT and CPU for PNSS and NSS

<table>
<thead>
<tr>
<th>n</th>
<th>PNSS IT</th>
<th>PNSS CPU</th>
<th>NSS IT</th>
<th>NSS CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>6</td>
<td>0.054</td>
<td>10</td>
<td>0.396</td>
</tr>
<tr>
<td>16</td>
<td>6</td>
<td>0.307</td>
<td>11</td>
<td>0.416</td>
</tr>
<tr>
<td>32</td>
<td>6</td>
<td>1.008</td>
<td>13</td>
<td>2.545</td>
</tr>
<tr>
<td>64</td>
<td>6</td>
<td>4.566</td>
<td>17</td>
<td>8.417</td>
</tr>
<tr>
<td>128</td>
<td>6</td>
<td>17.120</td>
<td>21</td>
<td>62.988</td>
</tr>
<tr>
<td>256</td>
<td>6</td>
<td>108.034</td>
<td>22</td>
<td>521.290</td>
</tr>
</tbody>
</table>

Example 2. To generate large sparse matrices $A$ and $B$, we build them in the following structures:

$$A = \begin{pmatrix} 10 & 1 & & 1 \\ 2 & 10 & 1 & \\ & \ddots & \ddots & \ddots \\ 2 & & 2 & 10 \end{pmatrix},$$

$$B = \begin{pmatrix} 8 & 1 & & 1 \\ 3 & 8 & 1 & \\ & \ddots & \ddots & \ddots \\ 3 & & 3 & 8 \end{pmatrix}.$$  

The computing results of the PNSS and the NSS are listed in the Table 3, the optimal parameters $\alpha_{\exp}$ for PNSS and NSS are presented in Table 4.

Table 4: The optimal parameters $\alpha_{\exp}$ for PNSS and NSS

<table>
<thead>
<tr>
<th>n</th>
<th>PNSS $\alpha_{\exp}$</th>
<th>NSS $\alpha_{\exp}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>1.14</td>
<td>7.8</td>
</tr>
<tr>
<td>16</td>
<td>1.14</td>
<td>7.3</td>
</tr>
<tr>
<td>32</td>
<td>1.14</td>
<td>6.2</td>
</tr>
<tr>
<td>64</td>
<td>1.14</td>
<td>5.0</td>
</tr>
<tr>
<td>128</td>
<td>1.14</td>
<td>4.1</td>
</tr>
<tr>
<td>256</td>
<td>1.14</td>
<td>3.8</td>
</tr>
</tbody>
</table>
From the results we can observe that the number of iteration steps (IT) of PNSS is smaller than that of NSS, and the PNSS has much less computational workload than NSS at each of the iteration steps, and the actual computing time (CPU) of PNSS may be less than that of NSS. So when the matrices A and B are large enough, the PNSS iteration methods considerably outperform the NSS iteration methods in both iteration step and computing time.

References