East Asian Journal on Applied Mathematics
doi: 10.4208/eajam.190716.051116a

Vol. 7, No. 1, pp. 55-69
February 2017

Preconditioned Positive-Definite and Skew-Hermitian Splitting Iteration Methods for Continuous Sylvester Equations $AX + XB = C$

Rong Zhou[1], Xiang Wang[1,2,*] and Xiao-Bin Tang[3]

[1] *Department of Mathematics, School of Sciences, Nanchang University, Nanchang 330031, China.*
[2] *Numerical Simulation and High-Performance Computing Laboratory, School of Sciences, Nanchang University, Nanchang 330031, China.*
[3] *School of Statistics, University of International Business and Economics, Beijing 100029, China.*

Received 19 July 2016; Accepted (in revised version) 5 November 2016.

Abstract. In this paper, we present a preconditioned positive-definite and skew-Hermitian splitting (PPSS) iteration method for continuous Sylvester equations $AX + XB = C$ with positive definite/semi-definite matrices. The analysis shows that the PPSS iteration method will converge under certain assumptions. An inexact variant of the PPSS iteration method (IPPSS) has been presented and the analysis of its convergence property in detail has been discussed. Numerical results show that this new method is more efficient and robust than the existing ones.

AMS subject classifications: 65F10, 65F50

Key words: PPSS iteration method, IPPSS iteration method, Sylvester equations, convergence.

1. Introduction

In this paper, we consider the iteration solution of the following continuous Sylvester equation of the form

$$AX + XB = C, \tag{1.1}$$

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
(A_1) A, B, C are large and sparse matrices;
(A_2) at least one of A and B is non-Hermitian;
(A_3) both A and B are positive semi-definite, and at least one of them is positive definite.

It is well known that the continuous Sylvester equation (1.1) has a unique solution if and only if there is no common eigenvalue between A and $-B$. Note that a Lyapunov equation is a special case of the continuous Sylvester equation with $B = A^*$ and C being

*Corresponding author. *Email address:* wangxiang49@ncu.edu.cn (X. Wang)

 ©2017 Global-Science Press

Hermitian. Here and in the sequel, A^* represents the conjugate transpose of the matrix A. Both Lyapunov and Sylvester equations have numerous applications. We can obtain the history of this class of equations and many interesting and important theoretical results in [2]. The continuous Sylvester equation (1.1) has numerous applications in control and system theory [32], signal processing [1], model order reduction [31], image restoration [19], stability of linear systems [26], analysis of bilinear systems [29], power systems [27], linear algebra [23], numerical methods for differential equations [2, 9, 10, 12, 13, 16, 17], matrix nearness problem [30], finite element model updating [22], block-diagonalization of matrices [23] and so on. Many of these applications lead to stable Sylvester equations, i.e., Assumption (A_3) made in the above is satisfied. Therefore, how to effectively solve this kind of equations involving literally hundreds or thousands of variables is under research.

The continuous Sylvester equation (1.1) is mathematically equivalent to the system of linear equations

$$\mathscr{A}x = c, \tag{1.2}$$

where $\mathscr{A} = I \otimes A + B^T \otimes I$, and the vectors x and c contain the concatenated columns of the matrices X and C respectively, with $\otimes$ being the Kronecker product symbol. Of course, this is a numerically poor way to determine the solution X of the continuous Sylvester equation (1.1), as the system of linear equations (1.2) is costly to solve and could be ill-conditioned.

Standard direct methods for numerical solution of the continuous Sylvester equation (1.1) are the Bartels-Stewart and the Hessenberg-Schur methods [24], 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 not applicable and too expensive for large-scale problems. For large-scale continuous Sylvester equations, iterative methods have been developed by taking advantage of the sparsity and the low-rank structure of C. The Alternating Direction Implicit (ADI) method [3, 35, 36] and the Krylov subspace based algorithms [6, 21, 25, 28, 33] are the most common iterative methods. Advantages of Krylov subspace based algorithms over ADI iterations are that no knowledge about the spectra of A and B is needed and (except for [33]) no linear systems of equations with (shifted) A and B have to be solved, and but ADI iterations often enable faster convergence if optimal shifts to A and B can be effectively estimated [4].

The HSS iteration method for system of linear equations was firstly proposed by Bai, Golub and Ng in [7], and then it was extended to other equations and conditions in [7–18, 34, 37–40]. Recently, Bai in [5] proposed a Hermitian and skew-Hermitian splitting (HSS) iteration method for solving large sparse continuous Sylvester equations with non-Hermitian and positive definite/semidefinite matrices. In [37], Wang et al. applied the idea of PSS iteration method in [6] to solve the continuous Sylvester equations and in [41]. Zheng and Ma applied the NSS iteration method [4] to solve the continuous Sylvester equations. Recently, Dong and Gu presented a PMHSS iteration method [20] for Sylvester equations. Motivated by this, we further present and analyze a preconditioned positive definite and skew-Hermitian iteration method for solving the continuous Sylvester equations, which is called PPSS iteration method.

The rest of the paper is organized as follows. In Section 2, after a brief introduction of the PSS iteration method [37], we present a PPSS iteration method for the continuous Sylvester equation (1.1) and derive some convergence properties of the PPSS iteration method. In Section 3, we establish an inexact preconditioned positive definite and skew-Hermitian iteration method (IPPSS) for (1.1). In Section 4, some numerical examples are presented to show the efficiency of the proposed method.

In the remainder of this paper, a matrix sequence $\{Y^{(k)}\}_{k=0}^{\infty} \subseteq \mathbb{C}^{n\times n}$ is said to be convergent to a matrix $Y \in \mathbb{C}^{n\times n}$ if the corresponding vector sequence $\{y^{(k)}\}_{k=0}^{\infty} \subseteq \mathbb{C}^{n^2}$ is convergent to the corresponding vector $y \in \mathbb{C}^{n^2}$, where the vectors $y^{(k)}$ and y contain the concatenated columns of the matrices $Y^{(k)}$ and Y, respectively.

2. PPSS Iteration Method

In this section, we consider the scheme of PPSS 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 positive definite and skew-Hermitian parts

$$A = \mathscr{P}(A) + \mathscr{S}(A), \qquad B = \mathscr{P}(B) + \mathscr{S}(B).$$

Then A and B can be rewritten as

$$\begin{aligned} A &= (\alpha I + \mathscr{P}(A)) + (\mathscr{S}(A) - \alpha I) = (\alpha I + \mathscr{S}(A)) + (\mathscr{P}(A) - \alpha I), \\ B &= (\alpha I + \mathscr{P}(B)) + (\mathscr{S}(B) - \alpha I) = (\alpha I + \mathscr{S}(B)) + (\mathscr{P}(B) - \alpha I). \end{aligned}$$

It follows that the continuous Sylvester equation (1.1) can be equivalently written as follows:

$$\begin{cases} (\alpha I + \mathscr{P}(A))X + X(\alpha I + \mathscr{P}(B)) = (\alpha I - \mathscr{S}(A))X + X(\alpha I - \mathscr{S}(B)) + C, \\ (\alpha I + \mathscr{S}(A))X + X(\alpha I + \mathscr{S}(B)) = (\alpha I - \mathscr{P}(A))X + X(\alpha I - \mathscr{P}(B)) + C. \end{cases} \tag{2.1}$$

Then we can easily establish the following positive-definite and skew-Hermitian splitting iteration method:

PSS 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:

$$\begin{cases} (\alpha I + \mathscr{P}(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\alpha I + \mathscr{P}(B)) = (\alpha I - \mathscr{S}(A))X^{(k)} + X^{(k)}(\alpha I - \mathscr{S}(B)) + C, \\ (\alpha I + \mathscr{S}(A))X^{(k+1)} + X^{(k+1)}(\alpha I + \mathscr{S}(B)) = (\alpha I - \mathscr{P}(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\alpha I - \mathscr{P}(B)) + C. \end{cases} \tag{2.2}$$

Now, based on the above observations, we can establish the following preconditioned positive definite and skew-Hermitian splitting iteration for solving the continuous Sylvester equation (1.1).

PPSS 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:

$$\begin{cases} (\alpha V_1 + \mathscr{P}(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\alpha V_2 + \mathscr{P}(B)) \\ \qquad = (\alpha V_1 - \mathscr{S}(A))X^{(k)} + X^{(k)}(\alpha V_2 - \mathscr{S}(B)) + C, \\ (\alpha V_1 + \mathscr{S}(A))X^{(k+1)} + X^{(k+1)}(\alpha V_2 + \mathscr{S}(B)) \\ \qquad = (\alpha V_1 - \mathscr{P}(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\alpha V_2 - \mathscr{P}(B)) + C, \end{cases} \tag{2.3}$$

where α is positive real number and V_1, V_2 are 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 $\alpha V_1 + \mathscr{P}(A)$ and $-(\alpha V_2 + \mathscr{P}(B))$, as well as between the matrices $\alpha V_1 + \mathscr{S}(A)$ and $-(\alpha V_2 + \mathscr{S}(B))$, so that these two fixed-point matrix equations have unique solution for all given right-hand side matrices. Naturally, the two half-steps involved in each step of the PPSS iteration method can be solved effectively using mostly real arithmetic. It is clear that the PPSS iteration method reduces to PSS 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. By making use of the Kronecker product, we can rewrite the above described PPSS iteration method in the following matrix-vector form,

$$\begin{cases} \big(I \otimes (\alpha V_1 + \mathscr{P}(A)) + (\alpha V_2 + \mathscr{P}(B))^T \otimes I\big) vec(X^{(k+\frac{1}{2})}) \\ \qquad = \big(I \otimes (\alpha V_1 - \mathscr{S}(A)) + (\alpha V_2 - \mathscr{S}(B))^T \otimes I\big) vec(X^{(k)}) + vec(C), \\ \big(I \otimes (\alpha V_1 + \mathscr{S}(A)) + (\alpha V_2 + \mathscr{S}(B))^T \otimes I\big) vec(X^{(k+1)}) \\ \qquad = \big(I \otimes (\alpha V_1 - \mathscr{P}(A)) + (\alpha V_2 - \mathscr{P}(B))^T \otimes I\big) vec(X^{(k+\frac{1}{2})}) + vec(C). \end{cases} \tag{2.4}$$

Denote by $\mathscr{A} = \mathscr{P} + \mathscr{S}$, with

$$\mathscr{P} = I \otimes \mathscr{P}(A) + \mathscr{P}(B)^T \otimes I, \qquad \mathscr{S} = I \otimes \mathscr{S}(A) + \mathscr{S}(B)^T \otimes I$$

and

$$\mathscr{K} = I \otimes V_1 + V_2^T \otimes I,$$

so

$$\mathscr{K}(\alpha) = I \otimes (\alpha V_1) + (\alpha V_2)^T \otimes I = \alpha \mathscr{K}.$$

Then we obtain

$$\begin{cases} (\alpha\mathscr{K} + \mathscr{P})vec(X^{(k+\frac{1}{2})}) = (\alpha\mathscr{K} - \mathscr{S})vec(X^{(k)}) + vec(C), \\ (\alpha\mathscr{K} + \mathscr{S})vec(X^{(k+1)}) = (\alpha\mathscr{K} - \mathscr{P})vec(X^{(k+\frac{1}{2})}) + vec(C). \end{cases} \tag{2.5}$$

Evidently, the iteration scheme (2.5) is the PPSS iteration method for solving the system of linear equation (1.2). After concrete operations, the PPSS iteration (2.5) can be neatly expressed as a stationary fixed-point iteration as follows:

$$vec(X^{(k+1)}) = \mathscr{M}(\alpha)vec(X^{(k)}) + \mathscr{G}(\alpha)vec(C),$$

where

$$\mathscr{M}(\alpha)=(\alpha\mathscr{H}+\mathscr{S})^{-1}(\alpha\mathscr{H}-\mathscr{P})(\alpha\mathscr{H}+\mathscr{P})^{-1}(\alpha\mathscr{H}-\mathscr{S}), \tag{2.6}$$

and

$$\mathscr{G}(\alpha)=(\alpha\mathscr{H}+\mathscr{S})^{-1}\left(I+(\alpha\mathscr{H}-\mathscr{P})(\alpha\mathscr{H}+\mathscr{P})^{-1}\right). \tag{2.7}$$

In addition, if we introduce matrices

$$\mathscr{F}_1(\alpha)=\frac{1}{2\alpha}(\alpha\mathscr{H}+\mathscr{P})\mathscr{H}^{-1}(\alpha\mathscr{H}+\mathscr{S})$$

and

$$\mathscr{G}_1(\alpha)=\frac{1}{2\alpha}(\alpha\mathscr{H}-\mathscr{P})\mathscr{H}^{-1}(\alpha\mathscr{H}-\mathscr{S}),$$

then it holds that

$$\mathbf{A}=\mathscr{F}_1(\alpha)-\mathscr{G}_1(\alpha),\qquad \mathscr{M}(\alpha)=\mathscr{F}_1(\alpha)^{-1}\mathscr{G}_1(\alpha).$$

In the following, we study the convergence of the PPSS method. Since the iteration method can be regarded as a combination of two two-step splitting iterations, we firstly give the general convergence criterion for the two-step splitting iterative method (2.5).

Lemma 2.1 ([5]). *Let $A,B,C\in\mathbb{C}^{n\times n}$, $A=M_i-N_i$ $(i=1,2)$ be two splittings of matrix A, and let $X^{(0)}$ be a given initial matrix. If $\{X^{(k)}\}$ is a two-step iteration sequence defined by*

$$\begin{cases} M_1X^{(k+\frac{1}{2})}B=N_1X^{(k)}B+C,\\ M_2X^{(k+1)}B=N_2X^{(k+\frac{1}{2})}B+C,\end{cases}$$

$k=1,2,\cdots$, then

$$X^{(k+1)}=M_2^{-1}N_2M_1^{-1}N_1X^{(k)}+M_2^{-1}(I+N_2M_1^{-1})CB^{-1}.$$

This iterative process can be rewritten in vector form

$$x^{(k+1)}=I\otimes(M_2^{-1}N_2M_1^{-1}N_1)x^{(k)}+\left(B^{-T}\otimes M_2^{-1}(I+N_2M_1^{-1})\right)vec(C). \tag{2.8}$$

Moreover, if the spectral radius $\rho(I\otimes(M_2^{-1}N_2M_1^{-1}N_1))$ is less than 1, the $\{X^{(k)}\}$ converges to $X^*\in\mathbb{C}^{n\times n}$ for all $X^{(0)}\in\mathbb{C}^{n\times n}$.

Lemma 2.2 ([6]). *Let*

$$\sigma(\alpha)=\|\,(\alpha I-\hat{P})(\alpha I+\hat{P})^{-1}\,\|_2\,. \tag{2.9}$$

If $\hat{P}\in\mathbb{C}^{m\times m}$ is a positive semi-definite matrix, then it holds that

$$\sigma(\alpha)\le 1,\quad \forall\alpha>0.$$

If $\hat{P}\in\mathbb{C}^{m\times m}$ is a positive definite matrix, then it holds that

$$\sigma(\alpha)<1,\quad \forall\alpha>0.$$

Lemma 2.3 ([6]). *Let $\hat{S} \in \mathbb{C}^{m\times m}$ be a skew-Hermitian matrix. Then for any $\alpha > 0$,*
(a) $\alpha I + \hat{S}$ is a non-Hermitian positive definite matrix;
(b) the Cayley transform $\mathcal{Q}(\alpha) \equiv (\alpha I - \hat{S})(\alpha I + \hat{S})^{-1}$ of $\hat{S}$ is a unitary matrix.

We can know from Lemma 2.3 that $\| \mathcal{Q}(\alpha) \|_2 = 1$. Similar to Theorem 2.1 in [6], we have the following theorem.

Theorem 2.1. *Let $A = \mathcal{P}(A) + \mathcal{S}(A)$ and $B = \mathcal{P}(B) + \mathcal{S}(B)$, where $\mathcal{P}(A)$, $\mathcal{P}(B)$ are the positive definite parts and $\mathcal{S}(A)$, $\mathcal{S}(B)$ are the skew-Hermitian parts. Denote by $\mathbf{A} = \mathcal{P} + \mathcal{S}$ with*

$$\begin{aligned}
&\mathbf{A} = I \otimes A + B^T \otimes I,\\
&\mathcal{P} = I \otimes \mathcal{P}(A) + \mathcal{P}(B)^T \otimes I, \qquad \mathcal{S} = I \otimes \mathcal{S}(A) + \mathcal{S}(B)^T \otimes I,\\
&\mathcal{K}(\alpha) = I \otimes (\alpha V_1) + (\alpha V_2)^T \otimes I = \alpha \mathcal{K},
\end{aligned}$$

where $V_1 \in \mathbb{R}^{m\times m}$, $V_2 \in \mathbb{R}^{n\times n}$ are prescribed symmetric positive definite matrices and α is a positive constant, and $\mathcal{M}(\alpha)$ defined in (2.6) *is the iteration matrix in* (2.5)*, and $\sigma(\alpha)$ is defined in* (2.9)*. Then the spectral radius $\rho(\mathcal{M}(\alpha))$ is bounded by $\sigma(\alpha)$. Therefore, it holds that*

$$\rho(\mathcal{M}(\alpha)) \le \sigma(\alpha) < 1, \quad \forall \alpha > 0.$$

So the PPSS iteration method is convergent to the exact solution $X_* \in \mathbb{C}^{m\times n}$ of Sylvester equations (1.1).

Proof. We can easily verify that K is symmetric positive definite matrix. Noting that $\hat{\mathcal{P}} = \mathcal{K}^{-1/2}\mathcal{P}\mathcal{K}^{-1/2}$, $\hat{\mathcal{S}} = \mathcal{K}^{-1/2}\mathcal{S}\mathcal{K}^{-1/2}$, then we can know that $\tilde{\mathcal{P}}$ is positive definite and $\tilde{\mathcal{S}}$ is skew-Hermitian. So we have

$$\begin{aligned}
&\rho(\mathcal{M}(\alpha))\\
=&\rho\left((\alpha\mathcal{K} + \mathcal{S})^{-1}(\alpha\mathcal{K} - \mathcal{P})(\alpha\mathcal{K} + \mathcal{P})^{-1}(\alpha\mathcal{K} - \mathcal{S})\right)\\
=&\rho\left((\alpha\mathcal{K} - \mathcal{P})(\alpha\mathcal{K} + \mathcal{P})^{-1}(\alpha\mathcal{K} - \mathcal{S})(\alpha\mathcal{K} + \mathcal{S})^{-1}\right)\\
\le& \| (\alpha\mathcal{K} - \mathcal{P})(\alpha\mathcal{K} + \mathcal{P})^{-1}(\alpha\mathcal{K} - \mathcal{S})(\alpha\mathcal{K} + \mathcal{S})^{-1}) \|_2\\
=& \| \mathcal{K}^{-\frac{1}{2}}(\alpha\mathcal{K} - \mathcal{P})K^{-\frac{1}{2}}\mathcal{K}^{\frac{1}{2}}(\alpha\mathcal{K} + \mathcal{P})^{-1}K^{\frac{1}{2}}\mathcal{K}^{-\frac{1}{2}}(\alpha\mathcal{K} - \mathcal{S})\mathcal{K}^{-\frac{1}{2}}\mathcal{K}^{\frac{1}{2}}(\alpha\mathcal{K} + \mathcal{S})^{-1})\mathcal{K}^{\frac{1}{2}} \|_2\\
=& \| (\alpha I - \hat{\mathcal{P}})(\alpha I + \hat{\mathcal{P}})^{-1}(\alpha I - \hat{\mathcal{S}})(\alpha I + \hat{\mathcal{S}})^{-1} \|_2\\
\le& \| (\alpha I - \hat{\mathcal{P}})(\alpha I + \hat{\mathcal{P}})^{-1} \|_2 \| (\alpha I - \hat{\mathcal{S}})(\alpha I + \hat{\mathcal{S}})^{-1} \|_2\\
=& \| (\alpha I - \hat{\mathcal{P}})(\alpha I + \hat{\mathcal{P}})^{-1} \|_2\\
=&\sigma(\alpha) < 1, \quad \forall \alpha > 0.
\end{aligned} \tag{2.10}$$

Therefore the PPSS iteration method converges unconditionally to the exact solution $X_* \in \mathbb{C}^{m\times n}$. □

3. IPPSS Iteration Method

The two-half steps at each step of the PPSS iteration method for solving the continuous Sylvester equation (1.1) require finding solutions of two continuous Sylvester equations

$$(\alpha V_1 + \mathscr{P}(A))X + X(\alpha V_2 + \mathscr{P}(B)) = C_S \tag{3.1}$$

and

$$(\alpha V_1 + \mathscr{S}(A))X + X(\alpha V_2 + \mathscr{S}(B)) = C_H \tag{3.2}$$

where C_S and C_H are prescribed m by 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 PPSS iteration, we can solve the two subproblems inexactly by utilizing certain effective iteration methods such as Gauss-Seidel, SOR, ADI method or Krylov subspace based methods, which results in the following inexact preconditioned positive definite and skew-Hermitian splitting iteration for solving the continuous Sylvester equation (1.1).

IPPSS iteration method
Given an initial guess $X^{(0)} \in \mathbb{C}^{m\times n}$, for $k = 0, 1, 2, \cdots$ until $\{X^{(k+1)}\}_{k=0}^{\infty} \in \mathbb{C}^{m\times n}$ satisfies the stopping criterion, solve $X^{(k+\frac{1}{2})} \in \mathbb{C}^{m\times n}$ approximately from

$$(\alpha V_1 + \mathscr{P}(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\alpha V_2 + \mathscr{P}(B)) \approx (\alpha V_1 - \mathscr{S}(A))X^{(k)} + X^{(k)}(\alpha V_2 - \mathscr{S}(B)) + C$$

by employing an inner iteration with $X^{(k)}$ being the initial guess, and then solve $\{X^{(k+1)}\}_{k=0}^{\infty} \in \mathbb{C}^{m\times n}$ approximately from

$$(\alpha V_1 + \mathscr{S}(A))X^{(k+1)} + X^{(k+1)}(\alpha V_2 + \mathscr{S}(B)) \approx (\alpha V_1 - \mathscr{P}(A))X^{(k+\frac{1}{2})} + X^{(k+\frac{1}{2})}(\alpha V_2 - \mathscr{P}(B)) + C$$

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

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

IPPSS iteration method
Given an initial guess $X^{(0)} \in \mathbb{C}^{m\times n}$, for $k = 0, 1, 2, \cdots$ until $\{X^{(k+1)}\}_{k=0}^{\infty} \in \mathbb{C}^{m\times n}$ converges:

Step 1. Approximate the solution of

$$(\alpha V_1 + \mathscr{P}(A))Z^{(k)} + Z^{(k)}(\alpha V_2 + \mathscr{P}(B)) = R^{(k)},$$

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

$$P^k = R^{(k)} - \left((\alpha V_1 + \mathscr{P}(A))Z^{(k)} + Z^{(k)}(\alpha V_2 + \mathscr{P}(B))\right),$$

satisfies

$$\|P^{(k)}\|_F \le \varepsilon_k \|R^{(k)}\|_F.$$

Then compute

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

Step 2. Approximate the solution of

$$(\alpha V_1 + \mathscr{S}(A))Z^{(k+\frac{1}{2})} + Z^{(k+\frac{1}{2})}(\alpha V_2 + \mathscr{S}(B)) = R^{(k+\frac{1}{2})},$$

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

$$Q^{(k+\frac{1}{2})} = R^{(k+\frac{1}{2})} - \Big((\alpha V_1 + \mathscr{S}(A))Z^{(k+\frac{1}{2})} + Z^{(k+\frac{1}{2})}(\alpha V_2 + \mathscr{S}(B))\Big),$$

satisfies

$$\|Q^{(k+\frac{1}{2})}\|_F \le \eta_k \|R^{(k+\frac{1}{2})}\|_F.$$

Then compute

$$X^{(k+1)} = X^{(k+\frac{1}{2})} + Z^{(k+\frac{1}{2})}.$$

Here, ε_k and η_k are prescribed tolerances which are used to control the accuracies of the inner iterations.

Theorem 3.1. *Assume that the assumptions of Theorem* 2.1 *hold. If* $\{X^{(k+1)}\}_{k=0}^{\infty} \in \mathbb{C}^{m\times n}$ *is the iteration sequence generated by the IPPSS iteration method and if* $X_* \in \mathbb{C}^{m\times n}$ *is the exact solution of the continuous Sylvester equation* (1.1)*, then we have*

$$\| X^{(k+1)} - X_* \| \le (\sigma(\alpha) + \theta\varrho\eta_k)(1 + \theta\varepsilon_k) \| X^{(k)} - X_* \|_{\mathscr{S}}, \tag{3.3}$$

where $\|\cdot\|_{\mathscr{S}}$ *is defined as* $\| Y \|_{\mathscr{S}} = \| (\alpha V_1 + \mathscr{S}(A))Y + Y(\alpha V_1 + \mathscr{S}(B)) \|_F$*, for any matrix* $Y \in \mathbb{C}^{m\times n}$*, and the constants* ϱ *and* θ *are given by*

$$\begin{aligned}
\varrho &= \| (\alpha\mathscr{K} + \mathscr{S})(\alpha\mathscr{K} + \mathscr{P})^{-1} \|_2, \\
\theta &= \| \mathscr{A}(\alpha\mathscr{K} + \mathscr{S})^{-1} \|_2, \\
\sigma(\alpha) &= \| (\alpha I - \hat{P})(\alpha I + \hat{P})^{-1} \|_2 .
\end{aligned}$$

In particular, if

$$(\sigma(\alpha) + \theta\varrho\eta_{max})(1 + \theta\varepsilon_{max}) < 1,$$

then the iteration sequence $\{X^{(k+1)}\}_{k=0}^{\infty} \in \mathbb{C}^{m\times n}$ *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 [5]. □

Theorem 3.2. *Assume that the conditions of Theorem* 2.1 *hold. Suppose that both* $r_1(k)$ *and* $r_2(k)$ *are nondecreasing and positive sequence satisfying* $r_1(k) \ge 1$ *and* $r_2(k) \ge 1$*, and* $\lim_{k\to\infty} \sup r_1(k) = \lim_{k\to\infty} \sup r_2(k) = +\infty$*, and that both* δ_1 *and* δ_2 *are real constants in the interval* $(0, 1)$ *satisfying*

$$\varepsilon_k \le c_1\delta_1^{r_1(k)} \quad \textit{and} \quad \eta_k \le c_2\delta_2^{r_2(k)}, \quad k = 0, 1, 2, \cdots$$

where c_1 and c_2 are nonnegative constants. Then it holds that

$$\|X^{(k+1)}-X^*\|_{\mathscr{S}} \le \left(\sqrt{\sigma(\alpha)+\varphi\theta\delta^{r(k)}}\right)^2 \|X^{(k)}-X^*\|_{\mathscr{S}},$$

where ρ and θ are defined in the above theorem and $r(k)$ and δ are defined as

$$r(k)=\min\{r_1(k),r_2(k)\}, \qquad \delta=\max\{\delta_1,\delta_2\},$$

and

$$\varphi=\max\left\{\sqrt{c_1c_2\varrho}, \frac{1}{2\sqrt{\sigma(\alpha)}}(c_1\sigma(\alpha)+c_2\varrho)\right\}.$$

Proof. The conclusion is straightforward according to Theorem 3.2 in [5]. □

4. PTSS Iteration Method and It's Convergence Rate

In this part, we will give two typical practical choices of the PS-splitting, which is called TSS (triangular and skew-Hermitian splitting) iteration method. It has the form as follows:

$$A=\mathscr{T}(A)+\mathscr{S}(A), \qquad B=\mathscr{T}(B)+\mathscr{S}(B),$$

with

$$\begin{cases} \mathscr{T}(A)=\mathscr{D}(A)+\mathscr{L}(A)+\mathscr{U}^*(A) & and \quad \mathscr{S}(A)=\mathscr{U}(A)-\mathscr{U}^*(A) \\ \mathscr{T}(B)=\mathscr{D}(B)+\mathscr{L}(B)+\mathscr{U}^*(B) & and \quad \mathscr{S}(B)=\mathscr{U}(B)-\mathscr{U}^*(B) \end{cases} \tag{4.1}$$

or

$$\begin{cases} \mathscr{T}(A)=\mathscr{D}(A)+\mathscr{L}^*(A)+\mathscr{U}(A) & and \quad \mathscr{S}(A)=\mathscr{L}(A)-\mathscr{L}^*(A) \\ \mathscr{T}(B)=\mathscr{D}(B)+\mathscr{L}^*(B)+\mathscr{U}(B) & and \quad \mathscr{S}(B)=\mathscr{L}(B)-\mathscr{L}^*(B) \end{cases} \tag{4.2}$$

where $\mathscr{D}(A)$ and $\mathscr{D}(B)$ are the diagonal matrices, $\mathscr{L}(A)$ and $\mathscr{L}(B)$ are the strictly lower triangular matrices of the matrices A and B, and $\mathscr{U}(A)$ and $\mathscr{U}(B)$ are are the strictly upper triangular matrices of the matrices A and B, respectively.

PTSS iteration method
Give 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:

$$\begin{cases} (\alpha V_1+\mathscr{T}(A))X^{(k+\frac{1}{2})}+X^{(k+\frac{1}{2})}(\alpha V_2+\mathscr{T}(B)) \\ \qquad =(\alpha V_1-\mathscr{S}(A))X^{(k)}+X^{(k)}(\alpha V_2-\mathscr{S}(B))+C, \\ (\alpha V_1+\mathscr{S}(A))X^{(k+1)}+X^{(k+1)}(\alpha V_2+\mathscr{S}(B)) \\ \qquad =(\alpha V_1-\mathscr{T}(A))X^{(k+\frac{1}{2})}+X^{(k+\frac{1}{2})}(\alpha V_2-\mathscr{T}(B))+C, \end{cases} \tag{4.3}$$

where α is positive real number and V_1, V_2 are prescribed symmetric positive definite matrices.

Now we can give the following theorem.

Theorem 4.1. *Let $\hat{A} \in \mathbb{C}^{mn\times mn}$ be a positive definite matrix, and $\hat{A} = \mathscr{T} + \mathscr{S}$,with*

$$\mathscr{T} = I \otimes \mathscr{T}(A) + \mathscr{T}(B)^T \otimes I \qquad and \qquad \mathscr{S} = I \otimes \mathscr{S}(A) + \mathscr{S}(B)^T \otimes I$$

$\mathscr{T}(A)$, $\mathscr{T}(B)$, $\mathscr{S}(A)$ and $\mathscr{S}(B)$ are defined in above described PTSS iteration, and α is a positive constant. Then the spectral radius $\rho(\mathscr{M}(\alpha))$ of the iteration matrix $\mathscr{M}(\alpha)$ of the PPSS iteration is bounded by

$$\sigma(\alpha) = \| (\alpha I - \hat{\mathscr{T}})(\alpha I + \hat{\mathscr{T}})^{-1} \|_2 = \frac{\alpha - 1}{\alpha + 1}$$

where $\hat{\mathscr{T}} = \mathscr{K}^{-1/2}\mathscr{T}\mathscr{K}^{-1/2}$, and the eigenvalues of matrix $\hat{\mathscr{T}}$ are 1.

Therefore, it holds that

$$\rho(\mathscr{M}(\alpha)) \le \sigma(\alpha) < 1, \qquad \forall \alpha > 0.$$

Proof. According to Theorem 2.1, we know that

$$\rho(\mathscr{M}(\alpha)) \le \sigma(\alpha) < 1, \qquad \forall \alpha > 0,$$

where $\sigma(\alpha) = \| (\alpha I - \hat{\mathscr{T}})(\alpha I + \hat{\mathscr{T}})^{-1} \|_2$. We note that when $\mathscr{T}(A) = \mathscr{D}(A) + \mathscr{L}(A) + \mathscr{U}^*(A)$ and $\mathscr{T}(B) = \mathscr{D}(B) + \mathscr{L}^*(B) + \mathscr{U}(B)$ or $\mathscr{T}(A) = \mathscr{D}(A) + \mathscr{L}^*(A) + \mathscr{U}(A)$ and $\mathscr{T}(B) = \mathscr{D}(B) + \mathscr{L}(B) + \mathscr{U}^*(B)$, the positive definite parts of A and B, i.e., $\mathscr{T}(A)$ and $\mathscr{T}(B)$ are triangular matrices, so $\mathscr{T} = I \otimes \mathscr{T}(A) + \mathscr{T}(A)^T \otimes I$ is also a triangular matrix, and we can easily know the diagonal elements of $\mathscr{T}$ are

$$T_{(k-1)\times m+l,(k-1)\times m+l} = a_{l,l} + b_{k,k},$$

where $a_{l,l}$, $b_{k,k}$ are the diagonal elements of $\mathscr{T}(A)$ and $\mathscr{T}(B)$, with $1 \le k \le n$, $1 \le l \le n$.

We set the preconditioning matrices $V_1 = diag(\mathscr{T}(A)) = \mathscr{D}(A)$, $V_2 = diag(\mathscr{T}(B)) = \mathscr{D}(B)$, so the diagonal elements of $\mathscr{K}$ also are

$$T_{(k-1)\times m+l,(k-1)\times m+l} = a_{l,l} + b_{k,k}.$$

So the eigenvalues of matrix $\hat{\mathscr{T}}$ are 1. Then from [5] we can obtain :

$$\begin{aligned} V(\alpha) \equiv & (\alpha I - \hat{\mathscr{T}})(\alpha I + \hat{\mathscr{T}})^{-1} \\ \approx & (\alpha I - \mathscr{D}(\hat{T}))(\alpha I + \mathscr{D}(\hat{T}))^{-1} \end{aligned}$$

and

$$\begin{aligned} \sigma(\alpha) = & \|V(\alpha)\|_2 \approx \left\|(\alpha I - \mathscr{D}(\hat{\mathscr{T}}))(\alpha I + \mathscr{D}(\hat{\mathscr{T}}))^{-1}\right\|_2 \\ = & \left|\frac{\alpha - 1}{\alpha + 1}\right|. \end{aligned}$$

□

5. Numerical Examples

In this section, we present some examples to illustrate the performance of the PPSS iteration method and its invariant version for solving the continuous Sylvester equation (1.1). The numerical experiments are performed in Matlab on an Inter dual core processor (1.83GHz, 2GB RAM). All iterations of this section are started from zero matrix and stopped once the current residual norm satisfying

$$\|R^{(k)}\|_F/\|R^{(0)}\|_F \leq 10^{-6},$$

where $R^{(k)} = C - AX^{(k)} - X^{(k)}B$. The number of iteration steps (denoted as "IT") and the computing time in seconds (denoted as "CPU") are recorded. We set the preconditioning matrices $V_1 = diag(\mathscr{P}(A))$, $V_2 = diag(\mathscr{P}(B))$. For PPSS method, we use the Bartels-Stewart method to solve these two Sylvester equations in inner iterations.

Example 5.1. We consider the continuous Sylvester equation (1.1) 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

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

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

We solve this continuous Sylvester equation by the PPSS and the PSS iteration methods. The computing results of PPSS iteration method and the PSS 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 spectral radius of PPSS iteration matrix and the PSS iteration matrix in Table 2. From the results in Table 1, we observe that the PPSS is much better than the PSS both in terms of the number of iteration steps and computing time, and the spectral radius of PPSS iteration matrix (denoted as "ρ_{pp}") is much smaller than the spectral radius PSS iteration matrix (denoted as "ρ_p").

Table 1: IT and CPU for PPSS and PSS.

	PPSS						PSS					
	r=0.01		r=0.1		r=1		r=0.01		r=0.1		r=1	
n	IT	CPU	IT	CPU	IT	CPU	IT	CPU	IT	CPU	IT	CPU
8	14	0.022	14	0.024	13	0.024	87	0.127	87	0.121	95	0.125
16	25	0.123	25	0.124	22	0.094	76	0.386	77	0.371	76	0.363
32	35	0.703	35	0.698	33	0.601	77	1.577	77	1.559	69	1.359
64	41	3.790	41	3.925	40	3.141	74	8.942	74	6.985	64	5.829
128	44	18.527	44	18.213	43	15.261	74	30.913	73	30.501	61	25.141
256	46	93.821	46	94.260	45	68.992	74	148.790	73	156.470	57	109.471

Table 2: The spectral radius of the PPSS iteration matrix and the PSS iteration matrix.

	PPSS			PSS		
n	r=0.01	r=0.1	r=1	r=0.01	r=0.1	r=1
8	0.643	0.642	0.538	0.847	0.847	0.861
16	0.633	0.632	0.428	0.843	0.842	0.785
32	0.624	0.623	0.538	0.838	0.838	0.837
64	0.605	0.606	0.470	0.829	0.828	0.821

Example 5.2. To generate large sparse matrices A and B , we build them in the following structures:

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

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

The computing results of the PPSS and the PSS are listed in Table 3. The spectral radius of PPSS iteration matrix and the PSS iteration matrix are presented in Table 4.

Example 5.3. The continuous Sylvester (1.1) with $m = n$ and the matrices

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

Table 3: IT and CPU for PPSS and PSS.

	PPSS		PSS	
n	IT	CPU	IT	CPU
8	8	0.011	45	0.075
16	8	0.037	69	0.324
32	10	0.194	40	0.788
64	12	1.310	33	3.651
128	14	6.715	30	13.433
256	17	29.328	29	61.583

Table 4: The spectral radius of the PPSS iteration matrix and the PSS iteration matrix.

	PPSS	PSS
n	ρ_{pp}	ρ_p
8	0.393	0.737
16	0.212	0.818
32	0.434	0.726
64	0.485	0.706

Table 5: IT and CPU for IPPSS and PPSS.

	IPPSS		PPSS	
n	IT	CPU	IT	CPU
8	9	0.017	26	0.032
16	22	0.154	57	0.314
32	32	0.889	73	1.518
64	67	5.904	91	10.132
128	127	47.316	307	140.309

with L the strictly lower triangular matrix having ones in the lower triangle part and t being a problem parameter to be specified in actual computations.

This continuous Sylvester equation is solved by the PPSS and IPPSS iteration methods, and the results are listed in Table 5. Here, we set $\varepsilon_k = \epsilon_k = 0.01$, $k = 0, 1, 2, \cdots$ and use the ADI method as the inner iteration scheme. From the results in Table 5, we observe that the IPPSS is much better than the PPSS in both iteration step and CPU time.

From the results we can observe that the number of iteration steps (IT) of PPSS is smaller than that of PSS, and the PPSS has much less computational workload than PSS at each of the iteration steps, and the actual computing time (CPU) of PPSS may be less than that of PSS. So when the matrices A and B are large enough, the PPSS iteration methods considerably outperform the PSS iteration methods in both iteration step and CPU time.

Acknowledgments

This work is supported by NSFC with No. 11461046, NSF of Jiangxi Province with No. 20161ACB21005, No. 20151BAB201009 and No. 20151BAB211013, the Scientific Research Foundation of Educational Department of Jiangxi Province with No. GJJ14154, and the Scientific Research Foundation of Graduate School of Nanchang University with No. YC2015-S018. This work is also supported by Program for Innovative Research Team in UIBE (CXTD5-05) and by Collaborative Innovation Center for Chinese Multinational Enterprise (No. 201504YY006A).

References

[1] B.D.O. Anderson, P. Agathoklis, E.I. Jury, M. Mansour. *Stability and the matrix Lyapunov equation for discrete 2-dimensional systems*, IEEE Trans. Circuits Syst. 33 (1986) 261-267.

[2] R. Bhatia, P. Rosenthal. *How and why to solve the operator equation $AX - XB = Y$*, Bull. Lond. Math. Soc. 29 (1997) 1-21.

[3] P. Benner, R.-C. Li, N. Truhar. *On the ADI method for Sylvester equations*, J. Comput. Appl. Math. 233 (2009) 1035-1045.

[4] Z.-Z. Bai, G. H. Golub, M.K. Ng. *On successive-overrelaxation acceleration of the Hermitian and skew-Hermitian splitting iterations*, Numer. Linear Algebra Appl. 14 (2007) 319-335.

[5] Z.-Z. Bai, *On Hermitian and skew-Hermitian spliting iteration methods for continuous Sylvester equations*, J. Comput. Math. 29 (2011) 185-198.

[6] Z.-Z. Bai, G. H. Golub, L.-Z. Lu, J.-F. Yin. *Block triangular and skew-Hermitian splitting methods for positive-definite linear systems*, SIAM J. Sci. Comput. 26 (2005) 844-863.

[7] Z.-Z. Bai, G. H. Golub, M. K. Ng. *Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, SIAM J. Matrix Anal. Appl., 24 (2003) 603-626.

[8] Z.-Z. Bai, Y.-B. Deng, Y.-H. Gao. *Iterative orthogonal direction methods for Hermitian minimum norm solutions of two consistent matrix equations*, Numer. Linear Algebra Appl., 13 (2006) 801-823.

[9] Z.-Z. Bai, *A class of two-stage iterative methods for systmes of weakly nonlinear equations*, Numer. Algorithms, 14 (1997) 295-319.

[10] Z.-Z. Bai, M. Benzi, F. Chen. *Modified HSS iteration methods for a class of complex symmetric linear systems*, Computing, 87 (2010) 93-111.

[11] Z.-Z. Bai, M. Benzi, F. Chen. *On preconditioned MHSS iteration methods for complex symmetric linear systems*, Numer. Algorithms, 56 (2011) 297-317.

[12] Z.-Z. Bai, X.-P. Guo. *On Newton-HSS methods for systems of nonlinear equations with positive definite Jacobian matrices*, J. Comput. Math., 28 (2010) 235-260.

[13] Z.-Z. Bai, *Several splittings for non-Hermitian linear systems*, Sci. China (Ser. A: Math), 51(2008) 1339-1348.

[14] Z.-Z. Bai, G. H. Golub, C.-K. Li, *Convergence properties of preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive semidefinite matrices*, Math. Comput., 76 (2007) 287-298.

[15] Z.-Z. Bai, G. H. Golub, M. K. Ng, *On inexact Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, Linear Algebra Appl., 428 (2008) 413-440.

[16] Z.-Z. Bai, Y.-M. Huang, M. K. Ng. *On preconditioned iterative methods for Burgers equations*, SIAM J. Sci. Comput., 29 (2007) 415-439.

[17] Z.-Z. Bai, M. K. Ng. *Preconditioners for nonsymmetric block Toeplitz-like-plus-diagonal linear systems*, Numer. Math., 96 (2003) 197-220.

[18] Z.-Z. Bai, J.-F. Yin, Y.-F. Su. *A shift-splitting preconditioner for non-Hermitian positive definite matrices*, J. Comput. Math., 24 (2006) 539-552.
[19] D. Calvetti, L. Reichel. *Application of ADI iterative methods to the restoration of noisy images*, SIAM J. Matrix Anal. Appl. 17(1996) 165-186.
[20] Y.-X. Dong, C.-Q. Gu. *On PMHSS iteration methods for continuous Sylvester equations*, J. Comput. Math., 2016, to appear.
[21] V. Druskin, L. Knizhnerman, V. Simoncini. *Analysis of the rational Krylov subspace and ADI methods for solving the Lyapunov equation*, SIAM J. Numer. Anal. 49 (2011) 1875-1898.
[22] M. I. Friswell, J. E. Mottershead. *Finite Element Model Updating in Structural Dynamics, Kluwer Academic Publishers*, Dordrecht, Boston and London, 1995.
[23] G. H. Golub, C.F.V. Loan. *Matrix Computations, Third edition, The Johns Hopkins University Press*, Baltimore and London, MD, 1996.
[24] G. H. Golub, S. G. Nash, C. F. V. Loan. *A Hessenberg-Schur method for the problem* $AX+XB=C$, IEEE Trans. Automat. Control 24 (1979) 909-913.
[25] D.-Y. Hu, L. Reichel. *Krylov-subspace methods for the Sylvester equation*, Linear Algebra Appl. 172 (1992) 283-313.
[26] A. Halanay, V. R. ăsvan. *Applications of Lyapunov Methods in Stability*, Kluwer Academic Publishers, Dordrecht, Boston and London, 1993.
[27] M.D. Ilic. *New approaches to voltage monitoring and control*, IEEE Control Syst. Mag. 9 (1989) 5-11.
[28] K. Jbilou. *Low rank approximate solutions to large Sylvester matrix equations*, Appl. Math. Comput. 177 (2006) 365-376.
[29] F. L. Lewis, V. G. Mertzios, G. Vachtsevanos, M. A. Christodoulou. *Analysis of bilinear systems using walsh functions*, IEEE Trans. Automat. Control 35 (1990) 119-123.
[30] A.-P. Liao, Z.-Z. Bai, Y. Lei. *Best approximate solution of matrix equation* $AXB+CYD=E$, SIAM J. Matrix Anal. Appl. 27 (2005) 675-688.
[31] G. Obinata, B.D.O. Anderson. *Model Reduction For Control System Design*, Springer-Verlag, London, 2001.
[32] A. van der Schaft. *L2-Gain and Passivity Techniques in Nonlinear Control*, 2nd edition, Springer-Verlag, London, 2000.
[33] V. Simoncini. *A new iterative method for solving large-scale Lyapunov matrix equations*, SIAM J. Sci. Comput. 29 (2007) 1268-1288.
[34] R. Zhou, X. Wang, X.-B. Tang. *A generalization of the Hermitian and skew-Hermitian splitting iteration method for solving Sylvester equations*, Appl. Math. Comput. 271 (2015) 609-617.
[35] X. Wang, L. Dai, D. Liao. *A modified gradient based algorithm for solving Sylvester equations*, Appl. Math. Comput. 218 (2012) 5620-5628.
[36] Q. Niu, X. Wang, L.-Z. Lu. *A relaxed gradient based algorithm for solving Sylvester equations*, Asian J. Control 13 (2011) 461-464.
[37] X. Wang, W.-W. Li, L.-Z. Mao. *On positive-definite and skew-Hermitian splitting iteration methods for continuous Sylvester equation* $AX+XB=C$, Comput. Math. Appl. 66 (2013) 2352-2361.
[38] R. Zhou, X. Wang, P. Zhou. *A modified HSS iteration method for solving the complex linear matrix equation* $AXB=C$, J. Comput. Math. 34 (2016) 437-450.
[39] X. Wang, W.-W. Li. *A modified GPSS method for non-Hermitian positive definite linear systems*, Appl. Math. Comput. 234 (2014) 253-259.
[40] X. Wang, Y. Li, L. Dai. *On Hermitian and skew-Hermitian splitting iteration methods for the linear matrix equation* $AXB=C$, Comput. Math. Appl. 234 (2014) 253-259.
[41] Q.-Q. Zheng, C.-F. Ma. *On normal and skew-Hermitian splitting iteration methods for large sparse continuous Sylvester equations*, J. Comput. Appl. Math. 268 (2014) 145-154.