Iranian Journal of Numerical Analysis and Optimization
Vol. 12, No. 1, (2022), pp 111-129
DOI:10.22067/ijnao.2021.71036.1044
https://ijnao.um.ac.ir/

How to cite this article
Research Article

Preconditioned global GPBiCG method for solving saddle point problems with multiple right-hand sides and its convergence analysis

A. Taherian and F. Toutounian*

Abstract

We propose the preconditioned global generalized product-type method based on the preconditioned global BiCG method to solve nonsymmetric saddle point problems with multiple right-hand sides. We apply an indefinite preconditioner to enhance the convergence rate of the method. We also present some theoretical analysis and discuss the convergence of the PGl-GPBiCG method. Some useful properties of the preconditioned matrix are established. Moreover, we present the bounds for the residual norm of the PGl-GPBiCG method according to the residual norm of the global GMRES method that guarantees convergence. Finally, some numerical examples are presented to show the efficiency of the new method in comparison with the preconditioned global BiCGSTAB method, and a comparison with another preconditioner is also provided.

AMS subject classifications (2020): 65F10.

Keywords: Saddle point problems; Preconditioned global GPBiCG; Indefinite preconditioning; Multiple right-hand sides; Convergence analysis.

*Corresponding author

Received 18 June 2021; revised 21 August 2021; accepted 25 August 2021

Atefeh Taherian
Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: a_taherian@mail.um.ac.ir

Faezeh Toutounian
Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran
and
The Center of Excellence on Modeling and Control Systems, Ferdowsi University of Mashhad, Iran. e-mail: toutouni@math.um.ac.ir

1 Introduction

Many problems in science and engineering applications such as mixed or mixed-hybrid finite element discretization of partial differential equations in computational fluid dynamics [13, 18, 31, 32, 41] and constrained optimization [29, 42, 43] eventuate in systems with multiple right-hand sides as

$$\begin{bmatrix} A & B \\ \epsilon B^T & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}, \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$ is a symmetric positive definite matrix, $B \in \mathbb{R}^{n\times m}$ has a full column rank, $X \in \mathbb{R}^{n\times s}$, $Y \in \mathbb{R}^{m\times s}$, $F_1 \in \mathbb{R}^{n\times s}$, and $F_2 \in \mathbb{R}^{m\times s}$. To convenience, we denote system (1) as follows:

$$\mathcal{A}\mathcal{X} = \mathcal{B}, \tag{2}$$

where $\mathcal{X} = [\mathcal{X}^{(1)}, \ldots, \mathcal{X}^{(s)}]$ and $\mathcal{B} = [b^{(1)}, \ldots, b^{(s)}]$. Under the assumptions mentioned above, the coefficient matrix $\mathcal{A}$ is nonsingular.

Several efficient iterative methods have been proposed during the recent decades to solve the saddle point problems, such as the Uzawa method [11, 12, 45], the Hermitian and skew-Hermitian splitting (HSS) iteration methods [3, 5, 9, 35, 37], SOR-type schemes [7, 19, 20, 26, 40, 47] and Krylov subspace methods [1, 2, 10, 21, 33, 34], and so on. In order to improve the efficiency of standard iterative solvers, many preconditioners have been presented in the literature, for example, block diagonal preconditioners [28, 38], constraint preconditioners [6, 25], block triangular preconditioners [4, 14, 36, 44], parametrized block triangular preconditioners [24], and HSS preconditioners [5, 9, 22, 30].

In this paper, we study the preconditioned global BiCG (PGl-BiCG) method [23] and preconditioned global GPBiCG (PGl-GPBiCG) method [46] for solving the linear systems with multiple right-hand sides (1). We concentrate on the use of the indefinite preconditioner

$$P = \begin{bmatrix} I & B \\ \epsilon B^T & 0 \end{bmatrix}. \tag{3}$$

This choice has been shown to be particularly effective on problems associated with constrained nonlinear programming [25, 27, 31]. In addition, in [33], the authors have shown that there is a tight connection between short term recurrence methods such as BiCG and the indefinite CG method used in [27]. More precisely, they are equivalent for a special choice of auxiliary vector, with which BiCG simplifies. In this paper, we discuss the convergence of the PGl-GPBiCG method and present two bounds for the residual norm of the method, which guarantee convergence.

The outline of the paper is as follows. In Section 2, we mention some properties of indefinite preconditioner. In Section 3, we review the PGl-

BiCG method. In Section 4, we describe the PGl-GPBiCG method to solve the system (1). Section 5 is devoted to the convergence analysis of the global GMRES method. In Section 6, we study some theoretical properties of the convergence of the PGl-GPBiCG method. In Section 7, numerical experiments confirm the described theoretical results. In Section 8, we present our conclusions.

Throughout this paper, we use the following notations. Inner product for two $n \times s$ matrices X and Y is defined as $\langle X, Y \rangle_F = \mathrm{tr}(X^T Y)$, where $\mathrm{tr}(Z)$ denotes the trace of the square matrix Z. The associated norm is the Frobenius norm denoted by $\| \cdot \|_F$. We will use the notation $\langle \cdot, \cdot \rangle_2$ for the usual inner product in $\mathbb{R}^n$, and the related norm will be denoted by $\| \cdot \|_2$. For a matrix $V \in \mathbb{R}^{n \times s}$, the block Krylov subspace $\mathcal{K}_k(A, V)$ is defined by $\mathcal{K}_k(A, V) = \mathrm{span}\{V, AV, A^2V, \ldots, A^{k-1}V\}$. Moreover, $Z \in \mathbb{K}_j(A, V)$ means that $Z = \sum_{i=0}^{j-1} \xi_i A^i V$, where $\xi_i \in \mathbb{R}$, for $i = 0, \ldots, j-1$. Finally, 0_s, I_s, and $0_{l \times s}$ will denote the zero, the identity, and zero matrices in $\mathbb{R}^{s \times s}$, $\mathbb{R}^{s \times s}$, and $\mathbb{R}^{l \times s}$, respectively. For brevity, we use the MATLAB-like notation $[v; w]$ to represent the vector $[v^T w^T]^T$.

2 Properties of the indefinite preconditioner

In this work, we use the global version of GPBiCG (Gl-GPBiCG)[46] for the solution of nonsymmetric saddle point problems with multiple right-hand sides (2). This method without a good preconditioner converges very slowly when applied to saddle point problems with multiple right-hand sides. In order to accelerate the convergence, we use the indefinite matrix P (defined in (3)) as a right preconditioner for the Gl-GPBiCG algorithm applied to the problem (1):

$$\mathcal{A}P^{-1} \begin{bmatrix} \tilde{X} \\ \tilde{Y} \end{bmatrix} = \mathcal{B}, \qquad \begin{bmatrix} X \\ Y \end{bmatrix} = P^{-1} \begin{bmatrix} \tilde{X} \\ \tilde{Y} \end{bmatrix}, \tag{4}$$

where

$$P^{-1} = \begin{bmatrix} I - \Pi & \frac{1}{\epsilon} B(B^T B)^{-1} \\ (B^T B)^{-1} B^T & -\frac{1}{\epsilon}(B^T B)^{-1} \end{bmatrix}, \qquad \mathcal{A}P^{-1} = \begin{bmatrix} G & S \\ 0 & I \end{bmatrix},$$

with $\Pi = B(B^T B)^{-1} B^T$, $G = A(I - \Pi) + \Pi$, and $S = \frac{1}{\epsilon}(A - I)B(B^T B)^{-1}$. Once an approximate solution $[\tilde{X}_k; \tilde{Y}_k]$ is determined, an approximate solution to the unpreconditioned problem is recovered as $[X_k; Y_k] = P^{-1}[\tilde{X}_k; \tilde{Y}_k]$. Choosing the vector $[\tilde{X}_0; \tilde{Y}_0] = [0; F_2]$ as the starting approximate solution, the initial residual is given by

$$R_0 = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} - \mathcal{A}P^{-1} \begin{bmatrix} \tilde{X}_0 \\ \tilde{Y}_0 \end{bmatrix} = \begin{bmatrix} F_1 - \frac{1}{\epsilon}(A - I)B(B^T B)^{-1} F_2 \\ 0 \end{bmatrix} = \begin{bmatrix} R_0^{(1)} \\ 0 \end{bmatrix},$$

so that the second block component of R_0 is identically zero. Problem (4) can thus be reformulated as determining an approximation $[\bar{X}_k; \bar{Y}_k]$ to the solution $[\bar{X}; \bar{Y}]$ of the system

$$\mathcal{A}P^{-1}\begin{bmatrix}\bar{X}\\ \bar{Y}\end{bmatrix} = R_0, \tag{5}$$

so that $[\tilde{X}_k; \tilde{Y}_k] = [\tilde{X}_0; \tilde{Y}_0] + [\bar{X}_k; \bar{Y}_k]$. In addition, for every $[U; 0] \in \mathbb{R}^{(m+n)\times s}$, we have

$$\mathcal{A}P^{-1}\begin{bmatrix}U\\ 0\end{bmatrix} = \begin{bmatrix}GU\\ 0\end{bmatrix} \qquad \text{and} \qquad P^{-1}\begin{bmatrix}U\\ 0\end{bmatrix} = \begin{bmatrix}(I-\Pi)U\\ (B^TB)^{-1}B^TU\end{bmatrix}. \tag{6}$$

In [33], the authors proved the following proposition for showing that the matrix G does have a full set of eigenvectors Z and for giving a bound for condition number of Z.

Proposition 1. Let us assume that the matrix $G = A(I-\Pi)+\Pi$ has $n-m$ nonunit eigenvalues $\lambda_i, i = 1, \ldots, n-m$. Let Z_2 be an orthonormal basis of span$\{\Pi\}$ and let the columns of $Y_1 \in \mathbb{R}^{n\times(n-m)}$ be eigenvectors of $(I-\Pi)A(I-\Pi)$ corresponding to all its nonzero eigenvalues. Then there exists an eigenvector matrix in the form $Z = [Z_1, Z_2]$ of $A(I-\Pi)+\Pi$ such that

$$\kappa(Z) \le (1+\|\gamma\|)^2 \quad \textit{with} \quad \|\gamma\| \le \frac{\|A\|_2}{\min_i |\lambda_i - 1|}, \tag{7}$$

where $\Lambda = diag(\lambda_i)$ and $\gamma = Z_2^TAY_1(\Lambda - I)^{-1}$.

The following proposition shows that the eigenvalues of $G = A(I-\Pi)+\Pi$ belong to a real positive interval when A is a symmetric positive definite matrix.

Proposition 2. Let τ be an eigenvalue of $G = A(I-\Pi)+\Pi$. Then either $\tau = 1$ or τ is a nonzero eigenvalue of $(I-\Pi)A(I-\Pi)$.

Proof. The proof is similar to that of Proposition 5 in [31]. □

3 The preconditioned global BiCG method

In this section, we employ the PGl-BiCG method [23] to solve system (4). Let $X_0 \in \mathbb{R}^{n\times s}$ be an initial guess with the residual $R_0 = \mathcal{B} - \mathcal{A}^{-1}X_0$ and let $\tilde{R}_0$ be an arbitrary $n \times s$ matrix. At step k, the residual R_k generated by this algorithm is such that, $R_k - R_0$ lies in the right matrix Krylov subspace $\mathcal{K}_k(\mathcal{A}, \mathcal{A}R_0)$ and R_k is F-orthogonal to the left matrix Krylov subspace $\mathcal{K}_k(\mathcal{A}^T, \tilde{R}_0) = \text{span}\{\tilde{R}_0, \mathcal{A}^T\tilde{R}_0, \ldots, \mathcal{A}^{T^{k-1}}\tilde{R}_0\}$.

For solving (1) by preconditioned Global Biconjugate Gradient (PGl-BiCG) algorithm by using the preconditioner P (defined in (3)) and equations (4) and (5), we choose $[\tilde{X}_0; \tilde{Y_0}] = [0; F_2]$ as an initial guess. So, $R_0 = \binom{R_0^{(1)}}{0}$. As in [33], we set $\tilde{R}_0 = P^{-1}R_0 = \binom{(I-\Pi)R_0^{(1)}}{(B^TB)^{-1}B^TR_0^{(1)}}$, $P_0 = R_0$, $\tilde{P}_0 = \tilde{R}_0$, and we obtain $\tilde{R}_k = P^{-1}R_k = \binom{(I-\Pi)R_k^{(1)}}{(B^TB)^{-1}B^TR_k^{(1)}}$ and $\tilde{P}_k = P^{-1}P_k = \binom{(I-\Pi)P_k^{(1)}}{(B^TB)^{-1}B^TP_k^{(1)}}$. Therefore, the iterates $\tilde{R}_k$ and $\tilde{P}_k$ can be computed explicitly from R_k and P_k, and the auxiliary "tilde" recurrence can be omitted. Now, by using the relations (6) and ignoring from the last m rows of the matrices that are zero matrices, we can summarize the PGl-BiCG algorithm for solving (2) as follows:

Algorithm 1: The right PGl-BiCG algorithm for solving (1)

```
Set $\begin{bmatrix}\tilde{X}_0\\ \tilde{Y}_0\end{bmatrix} = \begin{bmatrix}0\\ F_2\end{bmatrix}$ and $\begin{bmatrix}\bar{X}_0\\ \bar{Y}_0\end{bmatrix} = \begin{bmatrix}0\\ 0\end{bmatrix}$.
Compute $R_0^{(1)} = F_1 - \frac{1}{\epsilon}(A-I)B(B^TB)^{-1}F_2$ and set $R_0 = \begin{bmatrix}R_0^{(1)}\\ 0\end{bmatrix}$ and $P_0 = \begin{bmatrix}P_0^{(1)}\\ P_0^{(2)}\end{bmatrix} = \begin{bmatrix}R_0^{(1)}\\ 0\end{bmatrix}$
for $k = 0, 1, 2, \dots$ until convergence
    $\alpha_k = \frac{<R_k^{(1)},(I-\Pi)R_k^{(1)}>_F}{<GP_k^{(1)},(I-\Pi)P_k^{(1)}>_F}$
    $\bar{X}_{k+1} = \bar{X}_k + \alpha_k P_k^{(1)}$,                $\bar{Y}_{k+1} = O_{m\times s}$
    $R_{k+1}^{(1)} = R_k^{(1)} - \alpha_k G P_k^{(1)}$,              $R_{k+1}^{(2)} = 0$
    $\beta_k = \frac{<R_{k+1}^{(1)},(I-\Pi)R_{k+1}^{(1)}>_F}{<R_k^{(1)},(I-\Pi)R_k^{(1)}>_F}$
    $P_{k+1}^{(1)} = R_{k+1}^{(1)} + \beta_k P_k^{(1)}$,             $P_{k+1}^{(2)} = 0$
end
$\tilde{X}_{k+1} = \tilde{X}_0 + \bar{X}_{k+1}$,                     $\tilde{Y}_{k+1} = \tilde{Y}_0 + \bar{Y}_{k+1}$,
$X_{k+1} = (I - \Pi)\tilde{X}_{k+1} + \frac{1}{\epsilon}B(B^TB)^{-1}\tilde{Y}_{k+1}$,    $Y_{k+1} = (B^TB)^{-1}B^T\tilde{X}_{k+1} - \frac{1}{\epsilon}(B^TB)^{-1}\tilde{Y}_{k+1}$
```

In practical implementation of Algorithm 1, we can factorize B as $B = QR$ and use the relation $(B^TB)^{-1} = R^{-1}R^{-T}$. We also note that the cost of solving with R is very low due to the particular structure and sparsity of the matrix.

From Algorithm 1, the first block of residuals and first block of matrix directions can be expressed as follows:

$$R_k^{(1)PGl-BiCG} = \mathcal{R}_k(G)R_0^{(1)}, \quad P_k^{(1)PGl-BiCG} = \mathcal{P}_k(G)R_0^{(1)}, \tag{8}$$

where $\mathcal{R}_k(t)$, $\mathcal{P}_k(t) \in \mathbb{P}_k, \mathbb{P}_k$ is the set of polynomials $p_k(t)$ of degree k with scalar coefficients satisfying $p_k(0) = 1$. The polynomials $\mathcal{R}_k(t)$ and $\mathcal{P}_k(t)$ are related together with the recurrence formulas as follows:

$$\begin{aligned}\mathcal{R}_{k+1}(t) &= \mathcal{R}_k(t) - \alpha_k t\mathcal{P}_k(t), && (9)\\ \mathcal{P}_{k+1}(t) &= \mathcal{R}_{k+1}(t) + \beta_k \mathcal{P}_k(t). && (10)\end{aligned}$$

4 The preconditioned global GPBiCG method

In the PGl-GPBiCG method [46], the matrix residual satisfies

$$R_k = \mathcal{H}_k(\mathcal{A}^{-1}P)\mathcal{R}_k(\mathcal{A}^{-1}P)R_0, \tag{11}$$

where $\mathcal{R}_k$ is defined as above and $\mathcal{H}_k$ is an accelerating scalar polynomial, which is computed as the following recurrence:

$$\begin{aligned}&\mathcal{H}_0(t) = 1, \quad \mathcal{G}_0(t) = \zeta_0,\\ &\mathcal{H}_k(t) = \mathcal{H}_{k-1}(t) - t\mathcal{G}_{k-1}(t),\\ &\mathcal{G}_k(t) = \zeta_k \mathcal{H}_k(t) + \eta_k \mathcal{G}_{k-1}(t), \quad k = 1, 2, \ldots.\end{aligned} \tag{12}$$

If the PGl-GPBiCG algorithm is written for system (5) with $R_0 = \begin{bmatrix} R_0^{(1)} \\ 0 \end{bmatrix}$, then the relation (11) can be written as

$$R_k = \begin{bmatrix} \mathcal{H}_k(G)\mathcal{R}_k(G)R_0^{(1)} \\ 0 \end{bmatrix}, \tag{13}$$

and for solving (1.2), the PGl-GPBiCG algorithm can be summarized as Algorithm 2.

5 Convergence analysis of the global GMRES method

In this section, we consider the block linear system $GY = C$, where $G \in \mathbb{R}^{n\times n}$ and $Y, C \in \mathbb{R}^{n\times s}$, and we recall some convergence properties of the global GMRES method, which is needed in what follows. We consider the case where the matrix G is diagonalizable. Let $G = ZDZ^{-1}$, where D is a diagonal matrix whose elements are the eigenvalues $\lambda_1, \ldots, \lambda_n$ and Z is the eigenvector matrix. In [8], the following upper bounds for the Frobenius norm of the kth residual $R_k = C - GY_k$ of the global GMRES method was given. Here $R_0 = C - GY_0$ denotes the residual corresponding to the initial solution Y_0.

Theorem 1. [8] Let $\mathbb{P}_k$ be the set of polynomials of degree less or equal than k, and let $\kappa_2(Z) = \|Z\|_2\|Z^{-1}\|_2$. Then we have the following results:

$$\|R_k\|_F \le \kappa_2(Z)\|R_0\|_F \min_{p\in\mathbb{P}_k;p(0)=1} \big(\max_{\lambda\in Sp(G)} |p(\lambda)|\big), \tag{14}$$

Algorithm 2: The right PGl-GPBiCG algorithm

```
Set $\begin{bmatrix}\tilde{X}_0\\ \tilde{Y}_0\end{bmatrix} = \begin{bmatrix}0\\ F_2\end{bmatrix}$ and $\begin{bmatrix}\bar{X}_0\\ \bar{Y}_0\end{bmatrix} = \begin{bmatrix}0\\ 0\end{bmatrix}$.
Compute $R_0^{(1)} = F_1 - \frac{1}{\epsilon}(A-I)B(B^TB)^{-1}F_2$ and set $R_0 = \begin{bmatrix}R_0^{(1)}\\ 0\end{bmatrix}$,
Set $P_0 = R_0^{(1)}$, $T_{-1} = W_{-1} = P_{-1} = U_{-1} = 0_{n\times s},\ \beta_{-1} = 0$
for $k = 0, 1, 2, \dots$ until convergence
    $P_k = R_k^{(1)} + \beta_{k-1}(P_{k-1} - U_{k-1})$
    $\alpha_k = \frac{<R_k^{(1)},(I-\Pi)R_0^{(1)}>_F}{<GP_k,(I-\Pi)R_0^{(1)}>_F}$
    $Y_k = T_{k-1} - R_k^{(1)} - \alpha_k W_{k-1} + \alpha_k GP_k$
    $T_k = R_k^{(1)} - \alpha_k GP_k$
    $\zeta_k = \frac{<Y_k,Y_k>_F<GT_k,T_k>_F - <Y_k,T_k>_F<GT_k,Y_k>_F}{<GT_k,GT_k>_F<Y_k,Y_k>_F - <Y_k,GT_k>_F<GT_k,Y_k>_F}$
    $\eta_k = \frac{<GT_k,GT_k>_F<Y_k,T_k>_F - <Y_k,GT_k>_F<GT_k,T_k>_F}{<GT_k,GT_k>_F<Y_k,Y_k>_F - <Y_k,GT_k>_F<GT_k,Y_k>_F}$
        (if $k = 0$, then $\eta_k = 0, \zeta_k = \frac{<GT_k,T_k>_F}{<GT_k,GT_k>_F}$
    $U_k = \zeta_k GP_k + \eta_k(T_{k-1} - R_k^{(1)} + \beta_{k-1}U_{k-1})$
    $Z_k = \zeta_k R_k^{(1)} + \eta_k Z_{k-1} - \alpha_k U_k$
    $\bar{X}_{k+1} = \bar{X}_k + \alpha_k P_k + Z_k,$                $\bar{Y}_{k+1} = O_{m\times s}$
    $R_{k+1}^{(1)} = T_k - \eta_k Y_k - \zeta_k GT_k$                $R_{k+1}^{(2)} = 0$
    $\beta_k = \frac{\alpha_k}{\zeta_k}\frac{<R_{k+1}^{(1)},(I-\Pi)R_0^{(1)}>}{<R_k^{(1)},(I-\Pi)R_0^{(1)}>}$
    $W_k = GT_k + \beta_k GP_k$
end
$\tilde{X}_{k+1} = \tilde{X}_0 + \bar{X}_{k+1},$                $\tilde{Y}_{k+1} = \tilde{Y}_0 + \bar{Y}_{k+1},$
$X_{k+1} = (I-\Pi)\tilde{X}_{k+1} + \frac{1}{\epsilon}B(B^TB)^{-1}\tilde{Y}_{k+1},$        $Y_{k+1} = (B^TB)^{-1}B^T\tilde{X}_{k+1} - \frac{1}{\epsilon}(B^TB)^{-1}\tilde{Y}_{k+1}$
```

where $Sp(G)$ is the set of eigenvalues of matrix G.

Theorem 2. [8] Let the initial residual R_0 be decomposed as $R_0 = Z\beta$, where β is an $n \times s$ matrix whose columns are denoted by $\beta^{(1)}, \dots, \beta^{(s)}$. Then

$$\|R_k\|_F^2 \leq \frac{\|Z\|_2^2}{e_1^T(V_{k+1}^T\tilde{\mathcal{D}}V_{k+1})^{-1}e_1},$$

where

$$\tilde{\mathcal{D}} = diag\{\sum_{i=1}^{s}|\beta_1^{(i)}|^2, \dots, \sum_{i=1}^{s}|\beta_n^{(i)}|^2\} \qquad \text{and} \qquad V_{k+1} = \begin{pmatrix} 1 & \lambda_1 & \cdots & \lambda_1^k \\ \vdots & \vdots & & \vdots \\ 1 & \lambda_n & \cdots & \lambda_n^k \end{pmatrix}.$$

The coefficients $\beta_1^{(i)}, \dots, \beta_n^{(i)}$ are the components of the vector $\beta^{(i)}$, and e_1 is the first unit vector of $\mathbb{R}^{k+1}$.

From the fact that the eigenvalues of the matrix $G = A(I - \Pi) + \Pi$ are positive, the min-max problem in the bound of Theorem 1 can be explicitly solved. Denoting by $\lambda_{\min}$ and $\lambda_{\max}$ the (positive) extreme eigenvalues of the matrix G, the following asymptotic bound holds:

$$\|R_k\|_F \leq \kappa_2(Z)\|R_0\|_F(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1})^k, \tag{15}$$

where $\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}$ stands for the ratio of the extremal (real) eigenvalues of the matrix $G = A(I - \Pi) + \Pi$ (and so it is not its condition number!)[34].

6 Convergence behavior of PGl-GPBiCG

In this section, we propose an upper bound for the residual norm of the PGl-GPBiCG method. First, we derive an upper bound for the Frobenius norm of the kth residual $R_k^{PGl-BiCG}$ of the preconditioned global BiCG method. Using the steps 6 and 8 of Algorithm 1 and defining $\bar{R}_{k+1}^{(1)} = [R_0^{(1)} \quad R_1^{(1)} \quad \ldots \quad R_k^{(1)}]$, where $R_i^{(1)}$, $i = 0, \ldots, k$, are the first block of the PGl-BiCG residual matrices, we can easily show that

$$G\bar{R}_k^{(1)} = \bar{R}_k^{(1)}\hat{T}_k - \frac{1}{\alpha_{k-1}}R_k^{(1)}\bar{E}_k^T,$$

with $\bar{E}_k^T = [0_s, \ldots, 0_s, I_s] \in \mathbb{R}^{s\times ks}$ and $\hat{T}_k = \hat{L}_k\hat{\Lambda}_k^{-1}\hat{U}_k$, where

$$\hat{L}_k = \begin{pmatrix} I_s & & & & \\ -I_s & I_s & & & \\ & -I_s & \ddots & & \\ & & \ddots & I_s & \\ & & & -I_s & I_s \end{pmatrix}, \qquad \hat{U}_k = \begin{pmatrix} I_s & -\beta_0 I_s & & & \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ & & & I_s & -\beta_{k-2}I_s \\ & & & & I_s \end{pmatrix},$$

$\hat{\Lambda}_k = \operatorname{diag}[\alpha_0 I_s, \ldots, \alpha_{k-1}I_s]$, and α_i and β_i, $i = 0, 1, \ldots, k$, are scalars obtained from the PGl-BiCG algorithm. In addition, $\hat{T}_k$ is an invertible tridiagonal block matrix such that $\bar{E}_k^T\hat{T}_k^{-1}\bar{E}_1 = \alpha_{k-1}I_s$, where $\bar{E}_1^T = [I_s, \ldots, 0_s, 0_s] \in \mathbb{R}^{s\times ks}$.

By assuming that the matrix $\bar{R}_{k+1}^{(1)}$ is of full rank and $(\bar{R}_{k+1}^{(1)})^+ = ((\bar{R}_{k+1}^{(1)})^T\bar{R}_{k+1}^{(1)})^{-1}(\bar{R}_{k+1}^{(1)})^T$, as in [39], an upper bound for the norm of the first block of the residual of PGl-GPBiCG can be stated in the following theorem.

Theorem 3. Suppose that $G\bar{R}_k^{(1)} = \bar{R}_k^{(1)}\hat{T}_k - \frac{1}{\alpha_{k-1}}R_k^{(1)}\bar{E}_k^T$ with $\bar{E}_k^T\hat{T}_k^{-1}\bar{E}_1 = \alpha_{k-1}I_s$, and $\hat{V} = [I_{ks} \quad 0_{ks\times s}](\bar{R}_{k+1}^{(1)})^+$. If the matrix $\bar{R}_{k+1}^{(1)}$ is of full rank, then for any polynomial $p(t)$ of degree not exceeding k with $p(0) = 1$, we have

$$\|R_k^{(1)^{PGl-BiCG}}\|_F \leq \|\bar{N}_k\|_F \parallel p(G)R_0^{(1)} \parallel_F, \tag{16}$$

where $\bar{N}_k = I_n - (\bar{R}_k^{(1)} - R_k^{(1)}\bar{F}_k^T)\hat{V}$ with $\bar{F}_k^T = [I_s, I_s, \ldots, I_s] \in \mathbb{R}^{s\times ks}$.

Proof. The proof is similar to that of Theorem 1 in [39]. □

Now, we attempt to obtain an upper bound for the residual norm of the PGl-GPBiCG method. As in [39], we define the parameters $\bar{\zeta}_{j+1} = \zeta_j$, $\bar{\eta}_{j+1} = \eta_j$, and matrices

$$H_{j+1} = \mathcal{H}_j(G)R_k^{(1)^{PGl-BiCG}}, \qquad G_{j+1} = \frac{1}{\zeta_j}\mathcal{G}_j(G)R_k^{(1)^{PGl-BiCG}}, \quad j = 0, \ldots, k.$$

Using the relations (12), the iterates H_j and G_j can be computed by the recurrence formulas:

$$\begin{aligned} H_1 &= R_k^{(1)^{PGl-BiCG}}, \quad G_1 = R_k^{(1)^{PGl-BiCG}}, \\ H_{j+1} &= H_j - \bar{\zeta}_j G G_j, \\ G_{j+1} &= H_{j+1} + \frac{\bar{\eta}_{j+1}}{\bar{\zeta}_{j+1}}\bar{\zeta}_j G_j, \quad j = 1, 2, \ldots, k. \end{aligned} \tag{17}$$

By assuming that $\bar{\zeta}_j \neq 0, j = 1, \ldots, k+1$, we have
$H_{j+1}, G_{j+1} \in \mathbb{K}_{j+1}(G, R_k^{(1)^{PGl-BiCG}})$. Moreover, we obtain

$$H_{k+1} = R_k^{(1)^{PGl-GPBiCG}}. \tag{18}$$

Under the assumption that all the generated coefficients ζ_j are not zero and the grade μ of $R_k^{(1)^{PGl-BiCG}}$ with respect to G is not less than k, the recurrence formulas (17) determine the matrices

$$\bar{H}_k = [H_1, H_2, \ldots, H_k] \qquad \text{and} \qquad \bar{G}_k = [G_1, G_2, \ldots, G_k],$$

whose matrix columns H_j and G_j, $j = 1, \ldots, k$, are linear independent.

As in [39], by defining $\gamma_{j+1} = \bar{\zeta}_j\bar{\eta}_{j+1}\bar{\zeta}_{j+1}^{-1}$, for $j = 1, 2, \ldots, k$, the relations (17) can be written as follows:

$$G\bar{G}_k = \bar{H}_k\bar{L}_k\bar{\Lambda}_k^{-1} - \bar{\zeta}_k^{-1}H_{k+1}\bar{E}_k^T, \qquad \bar{H}_k = \bar{G}_k\bar{U}_k, \tag{19}$$

where $\bar{\Lambda}_k = \text{diag}[\bar{\zeta}_1 I_s, \ldots, \bar{\zeta}_k I_s]$, $\bar{E}_k^T = [0_s, \ldots, 0_s, I_s] \in \mathbb{R}^{s\times ks}$,

$$\bar{L}_k = \hat{L}_k, \quad \text{and} \quad \bar{U}_k = \begin{pmatrix} I_s & -\gamma_2 I_s & & & \\ & I_s & & & \\ & & \ddots & & \\ & & & \ddots & -\gamma_k I_s \\ & & & & I_s \end{pmatrix}. \tag{20}$$

Combining two equations in (19), we get

$$G\bar{H}_k = \bar{H}_k \bar{S}_k - \bar{\zeta}_k^{-1} H_{k+1} \bar{E}_k^T, \tag{21}$$

with $\bar{S}_k = \bar{L}_k \bar{\Lambda}_k^{-1} \bar{U}_k$ and $\bar{E}_k^T \bar{S}_k^{-1} \bar{E}_1 = \bar{\zeta}_k I_s$, where $\bar{E}_1^T = [I_s, 0_s, \ldots, 0_s] \in \mathbb{R}^{s \times ks}$.

By assuming that the matrix $\bar{H}_{k+1}$ is of full rank and that $\bar{H}_{k+1}^+ = (\bar{H}_{k+1}^T \bar{H}_{k+1})^{-1} \ \bar{H}_{k+1}^T$, similar to Theorem 3, we can present the following theorem for the PGl-GPBiCG method.

Theorem 4. Let $G\bar{H}_k = \bar{H}_k \bar{S}_k - \bar{\zeta}_k^{-1} H_{k+1} \bar{E}_k^T$ with $\bar{E}_k^T \bar{S}_k^{-1} \bar{E}_1 = \bar{\zeta}_k I_s$, and let $\bar{V} = [I_{ks} \quad 0_{ks \times s}] \bar{H}_{k+1}^+$. If the matrix $\bar{H}_{k+1}$ is of full rank, then for any polynomial $p(t)$ of degree not exceeding k with $p(0) = 1$, we have

$$\|R_k^{(1)_{PGl-GPBiCG}}\|_F \leq \|\bar{M}_k\|_F \ \| \ p(G) R_k^{(1)_{PGl-BiCG}} \ \|_F, \tag{22}$$

where $\bar{M}_k = I_n - (\bar{H}_k - H_{k+1} \bar{F}_k^T) \bar{V}$ with $\bar{F}_k^T = [I_s, I_s, \ldots, I_s] \in \mathbb{R}^{s \times ks}$.

Proof. The proof is similar to that of Theorem 1 in [39]. □

Now, based on the above observations, we state the following theorem for bounding the residual norm of the PGl-GPBiCG method.

Theorem 5. With the notation of Theorems 3 and 4 for $R_0 = \begin{bmatrix} R_0^{(1)} \\ 0 \end{bmatrix}$ and assuming that $G = A(I - \Pi) + \Pi$ is diagonalizable, the right preconditioned global GPBiCG residual satisfies

$$\|R_k^{PGl-GPBiCG}\|_F \leq (\kappa_2(Z))^2 \|\bar{N}_k\|_F \|\bar{M}_k\|_F \|R_0\|_F (\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1})^{2k}, \tag{23}$$

where $\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}$ stands for the ratio of the extremal (real) eigenvalues of matrix $G = A(I - \Pi) + \Pi$.

In addition, with the notation of Theorem 2 and $R_0^{(1)} = \beta Z$, we have

$$\|R_k^{PGl-GPBiCG}\|_F \leq \kappa_2(Z) \|\bar{N}_k\|_F \|\bar{M}_k\|_F \frac{\|Z\|_2}{\sqrt{(e_1^T (V_{k+1}^T \tilde{D} V_{k+1})^{-1} e_1)}} (\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1})^k, \tag{24}$$

where $\tilde{D}$ and V_{k+1} are defined in Theorem 2.

So, the convergence is guaranteed.

Proof. Since the inequality (22) holds for any polynomial $p(t)$ of degree not exceeding k with $p(0) = 1$, the use of the residual polynomial of the global GMRES ($p_k^{Gl-GMRES}(G)$) method in (22) implies that

$$\begin{aligned}\|R_k^{PGl-GPBiCG}\|_F &\leq \|\bar{M}_k\|_F \parallel p_k^{Gl-GMRES}(G) R_k^{(1)_{PGl-BiCG}} \parallel_F \\ &\leq \kappa_2(Z)\|\bar{M}_k\|_F \parallel R_k^{(1)_{PGl-BiCG}} \parallel_F \left(\tfrac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k,\end{aligned}$$

The second inequality follows from (15). Now, from the fact that (16) also holds for any polynomial $p(t)$ of degree not exceeding k with $p(0) = 1$, we get

$$\|R_k^{(1)_{PGl-GPBiCG}}\|_F \leq \kappa_2(Z)\|\bar{N}_k\|_F\|\bar{M}_k\|_F \parallel p_k^{Gl-GMRES}(G) R_0^{(1)} \parallel_F \left(\tfrac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k. \tag{25}$$

This together with the inequality (15) implies that

$$\|R_k^{(1)_{PGl-GPBiCG}}\|_F \leq (\kappa_2(Z))^2\|\bar{N}_k\|_F\|\bar{M}_k\|_F\|R_0^{(1)}\|_F\left(\tfrac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^{2k}. \tag{26}$$

Since $R_k^{(2)} = 0$, so we have the inequality (23).

Finally, due to Theorem 2, the inequality (25) yields the inequality given in (24). □

7 Numerical results

In this section, we illustrate the efficiency of the PGl-GPBiCG method for solving system (1). All the numerical experiments were performed in MATLAB R2017b with PC-Intel(R) Core(TM) i7, CPU 3.60 GHz, 16.00 GB of RAM. In all the examples, the starting guess was taken to be zero. The stopping criterion

$$\frac{\|R_k\|_F}{\|R_0\|_F} \leq 10^{-9}$$

was used.

Example 1. In this example, a set of 6 problems was taken from the University of Florida Sparse Matrix Collection[17] as the matrix A. These matrices with their generic properties are given in Table 1. Also, we consider $\epsilon = 1$, $B = \text{rand(n, m)}$, $F_1 = \text{rand}(n, s)$, and $F_2 = \text{rand}(m, s)$, where the function $\text{rand}(l, k)$ creates an $l \times k$ random matrix with coefficients uniformly distributed in [0, 1].

We compare the Frobenius norm of the residuals (RES), the number of iterations (Iter), and the CPU time in seconds (CPU) required for convergence of the PGl-GPBiCG and PGl-BiCGSTAB methods for $s = 5, 10, 20$.

	Matrix	Property		
		order	nnz	spd
1	gr_30_30	900	7744	yes
2	nos3	960	15844	yes
3	1138_bus	1138	4054	yes
4	bcsstm11	1473	1473	yes
5	Dubcova1	16129	253009	yes
6	bodyy4	17546	121550	yes

Table 1: Test problems information

The results obtained by these algorithms are presented in Table 2. As can be seen, the methods are similar in terms of the Frobenius norm of residuals ($\| R_k \|_F$), the number of iterations of the PGl-GPBiCG method is less than that of the PGl-BiCGSTAB method and the CPU time required for convergence in the PGl-GPBiCG method is similar or lower than that in the PGl-BiCGSTAB method except, for examples, nos3 with $s = 10, 20$.

		PGl-BiCGSTAB			PGl-GPBiCG		
Matrix \ s		5	10	20	5	10	20
gr_30_30	Iter	47	49	49	38	38	38
	CPU	0.02	0.04	0.07	0.02	0.03	0.06
	RES	1.1082e-07	3.4408e-07	6.3886e-07	2.6191e-07	4.2295e-07	6.1987e-07
nos3	Iter	192	183	193	167	180	172
	CPU	0.16	0.17	0.23	0.16	0.24	0.30
	RES	2.7939e-07	7.3530e-07	8.0708e-07	3.9410e-07	7.6350e-07	1.0669e-06
1138_bus	Iter	8364	4011	5393	2793	2677	2543
	CPU	9.61	6.02	10.72	4.23	4.84	6.33
	RES	4.8730e-06	6.4503e-06	9.1015e-06	3.2011e-06	5.1722e-06	1.0123e-05
bcsstm11	Iter	335	381	441	216	335	250
	CPU	0.98	1.20	1.72	0.59	1.05	1.11
	RES	4.0741e-07	2.0770e-07	4.6741e-07	4.0252e-07	5.4683e-07	2.0180e-07
Dubcova1	Iter	105	99	99	85	85	86
	CPU	30.48	35.21	40.87	24.62	31.08	37.33
	RES	1.2814e-06	2.1262e-06	3.1051e-06	1.5025e-06	2.1308e-06	3.2268e-06
bodyy4	Iter	182	194	170	165	166	165
	CPU	68.18	80.66	80.75	62.54	69.50	79.63
	RES	1.5969e-06	2.1782e-06	3.2651e-06	1.6013e-06	2.3639e-06	3.1067e-06

Table 2: Numerical results for Example 1 with $s = 5, 10, 20$

Example 2. In this example, we consider the Stokes equation [16, 15] as

$$\begin{aligned} -\nu\Delta\mathbf{u} + \bigtriangledown p &= \tilde{f}, \quad \text{in } \Omega \\ \bigtriangledown \cdot \mathbf{u} &= \tilde{g}, \quad \text{in } \Omega \\ \mathbf{u} &= 0, \quad \text{on } \partial\Omega \\ \int_\Omega p(x)dx &= 0, \end{aligned} \tag{27}$$

where $\Omega = (0,1) \times (0,1) \subset \mathbb{R}^2$, $\partial\Omega$ is the boundary of Ω, ν is the viscosity scalar, and u and p denote the velocity and the pressure, respectively. By discretizing (27), we obtain the system of linear equations as

$$\begin{bmatrix} A & B \\ -B^T & 0 \end{bmatrix} \begin{bmatrix} U \\ P \end{bmatrix} = \begin{bmatrix} F_1 \\ -F_2 \end{bmatrix},$$

in which

$$A = \begin{bmatrix} I \otimes T + T \otimes I & 0 \\ 0 & I \otimes T + T \otimes I \end{bmatrix} \in \mathbb{R}^{2q^2 \times 2q^2}, \quad B = \begin{bmatrix} I \otimes F \\ F \otimes I \end{bmatrix} \in \mathbb{R}^{2q^2 \times q^2},$$

where T and F are tridiagonal matrices given by

$$T = \frac{\nu}{h^2} \cdot \operatorname{tridiag}(-1, 2, -1) \in \mathbb{R}^{q \times q}, \quad F = \frac{1}{h} \cdot \operatorname{tridiag}(-1, 1, 0) \in \mathbb{R}^{q \times q},$$

and $\otimes$ denotes the Kronecker product. Also, $h = \frac{1}{q+1}$ is the discretization mesh size. We set $n = 2q^2$ and $m = q^2$. Hence, the total number of variables is $n + m = 3q^2$. We choose the right-hand side such that the exact solution of saddle point problem is a matrix of ones. For this example, we test three different ν's, that is, $\nu = 0.01, 0.1, 1$ and $q = 16, 32$.

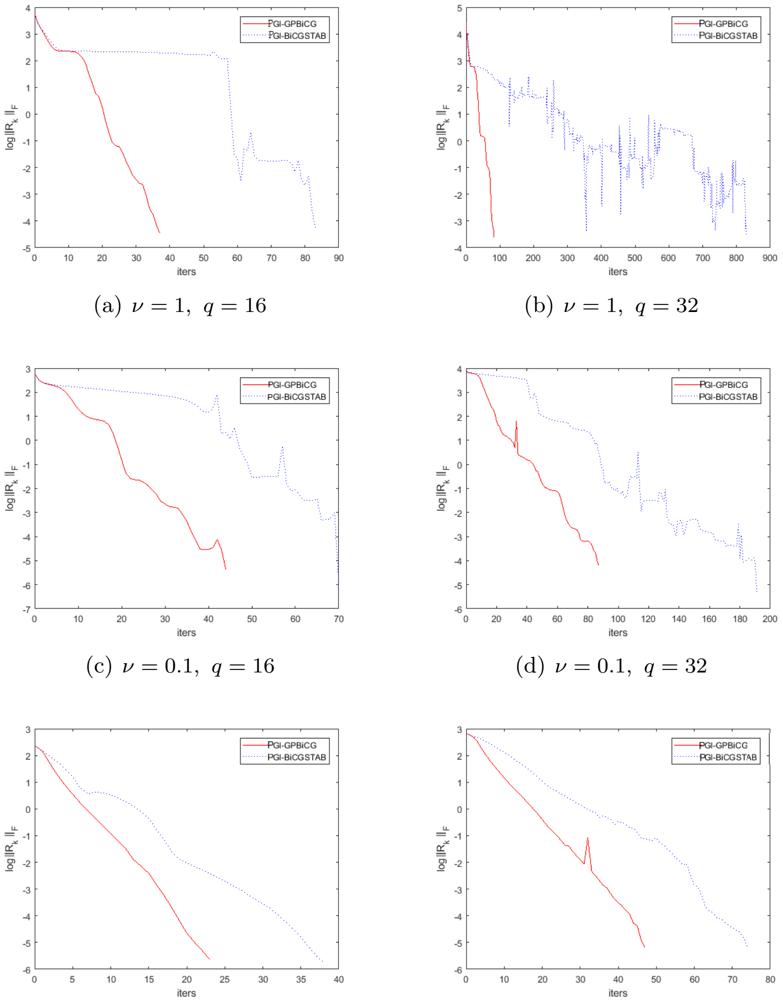
In Table 3, we list the Frobenius norm of the residuals (RES), the number of iterations (Iter), and the CPU time in seconds (CPU) required for the convergence of the PGl-GPBiCG and PGl-BiCGSTAB methods with $\nu = 1, 0.1, 0.01$, $q = 16, 32$, and $s = 5$. As expected, the number of iterations of PGl-GPBiCG is less than that of PGl-BiCGSTAB. The CPU time obtained for PGl-GPBiCG is smaller than the one for PGl-BiCGSTAB except for $\nu = 0.01$ and $q = 16$.

		PGl-BiCGSTAB		PGl-GPBiCG	
$\nu \setminus q$		16	32	16	32
0.01	Iter	38	74	23	47
	CPU	0.03	2.16	0.03	1.97
	RES	1.7694e-06	6.4937e-06	2.3360e-06	6.6436e-06
0.1	Iter	70	222	44	80
	CPU	0.04	3.10	0.03	2.20
	RES	9.3235e-07	1.8710e-05	4.2220e-06	2.3113e-05
1	Iter	83	828	37	82
	CPU	0.05	6.70	0.04	2.21
	RES	5.5359e-05	2.7881e-04	3.5071e-05	2.3870e-04

Table 3: Numerical results for Example 2 with $\nu = 0.01, 0.1, 1$

In Figure 1, we display the convergence history of the PGl-GPBiCG and PGl-BiCGSTAB algorithms for Stokes problem with $s = 5$, $q = 16, 32$, and

$\nu = 0.01, 0.1, 1$, respectively. In these figures, the horizontal axis is the number of iterations (iters) and the vertical axis is the logarithm of the Frobenius norm of residuals ($\log_{10} \parallel R_k \parallel_F$). The results show that whatever v becomes lower, the methods are smoother. Moreover, the PGl-GPBiCG method is more effective and smoother than the PGl-BiCGSTAB method.



(a) $\nu = 1,\ q = 16$

(b) $\nu = 1,\ q = 32$

(c) $\nu = 0.1,\ q = 16$

(d) $\nu = 0.1,\ q = 32$

(e) $\nu = 0.01\ q = 16$

(f) $\nu = 0.01,\ q = 32$

Figure 1: Convergence history of the PGl-GPBiCG algorithm for Stokes problem with $s = 5$ and different values of ν and q.

Finally, we compare the numerical results obtained by the Gl-GPBiCG method with the preconditioner (3) and the preconditioner given in [1],

$$P_{\epsilon,\alpha,Q} = \begin{bmatrix} A & B \\ \epsilon B^T & \alpha Q \end{bmatrix}$$

with $Q = I$. For $q = 16, 32, 64$, $\nu = 1$, $\alpha = 0.05, 0.1, 0.5, 1, 10$, and $s = 5$ the numerical results are given in Table 4. As expected, the preconditioner discussed in this paper needs less CPU time than the preconditioner given in [1] except for $q = 64$ and $\alpha = 0.5, 1, 10$.

		PGl-GPBiCG (pre. in [1])					PGl-GPBiCG (pre. (3))
$q \setminus \alpha$		0.05	0.1	0.5	1	10	
	Iter	196	88	25	22	35	37
16	CPU	0.66	0.35	0.11	0.10	0.14	0.04
	RES	2.7200e-04	6.6880e-05	4.2144e-05	7.7757e-05	6.2674e-05	3.5071e-05
	Iter	276	109	38	35	53	82
32	CPU	17.10	6.87	2.50	2.35	3.53	2.21
	RES	0.0013	6.1728e-04	1.5556e-4	2.5773e-04	1.3334e-04	2.3870e-04
	Iter	318	162	56	56	91	201
64	CPU	315.67	162.91	60.20	60.15	94.10	128.57
	RES	0.0060	0.0017	0.0021	0.0014	2.5978e-04	0.0015

Table 4: Numerical results for Example 2 with $\alpha = 0.05, 0.1, 0.5, 1, 10$

8 Conclusion

In this paper, we applied the Gl-GPBiCG method to solve the nonsymmetric saddle point problems with multiple right-hand sides. By using the indefinite preconditioner (3), we accelerated the convergence rate of the method. Also, we studied some theoretical properties of the PGl-GPBiCG method and presented two bounds for the residual norm of the method, which guarantee the convergence. As expected, the experimental results showed that the number of iterations (Iter) of the PGl-GPBiCG method is less than that of the PGl-BiCGSTAB method. The CPU time (CPU) of the PGl-GPBiCG method is sometimes more than that of the PGl-BiCGSTAB method because of the existence of parameters ζ_j and η_j. In addition, the PGl-GPBiCG method is more effective and smoother than the PGl-BiCGSTAB method.

Acknowledgements

We would like to thank the referees for their valuable remarks and helpful suggestions that improved the presentation.

References

1. Badahmane, A., Bentbib, A.H., and Sadok, H. *Preconditioned global Krylov subspace methods for solving saddle point problems with multiple right-hand sides*, Electron. Trans. Numer. Anal. 51 (2019), 495–511.

2. Badahmane, A., Bentbib, A.H. and Sadok, H. *Preconditioned Krylov subspace and GMRHSS iteration methods for solving the nonsymmetric saddle point problems,* Numer. Algorithms, 84 (2020), 1295–1312.

3. Bai Z.-Z. and Benzi, M. *Regularized HSS iteration methods for saddle-point linear systems,* BIT Numerical Mathematics, 57 (2017), 287–311.

4. Bai, Z.-Z., Golub, G.H., Lu, L.-Z. and Yin, J.-F. *Block triangular skew-Hermitian splitting methods for positive-definite linear systems,* SIAM J. Sci. Comput. 26 (2004), 844–863.

5. Bai, Z.-Z., Golub, G.H. and Pan, J.-Y. *Preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive semidefinite linear systems,* Numer. Math. 98 (2004), 1–32.

6. Bai, Z.-Z., Ng, M.K. and Wang, Z.-Q. *Constraint preconditioners for symmetric indefinite matrices,* SIAM J. Matrix Anal. Appl. 31 (2009), 410–433.

7. Bai, Z.-Z., Parlett, B.N. and Wang, Z.-Q. *On generalized successive overrelaxation methods for augmented linear systems,* Numer. Math. 102 (2005), 1–38.

8. Bellalij, M., Jbilou, K. and Sadok, H. *New convergence results on the global GMRES method for diagonalizable matrices,* J. Comput. Appl. Math. 219 (2008), 350–358.

9. Benzi M. and Golub, G.H. *A preconditioner for generalized saddle point problems,*SIAM J. Matrix Anal. Appl. 26 (2004), 20–41.

10. Benzi, M. and Wathen, A.J. S*ome preconditioning techniques for saddle point problems. Model order reduction: theory, research aspects and applications*, 195–211, Math. Ind., 13, Eur. Consort. Math. Ind. (Berl.), Springer, Berlin, 2008.

11. Bramble, J.H., Pasciak, J.E. and Vassilev, A.T. *Analysis of the inexact Uzawa algorithm for saddle point problems,* SIAM J. Numer. Anal. 34 (1997), 1072–1092.

12. Bramble, J.H., Pasciak, J.E. and Vassilev, A.T. *Uzawa type algorithms for nonsymmetric saddle point problems,* Math. Comp. 69 (2000), 667–689.

13. Brezzi, F. and Fortin, M. *Mixed and hybrid finite element methods,* Springer Series in Computational Mathematics, 15. Springer-Verlag, New York, 1991.

14. Cao, Z.-H. *Positive stable block triangular preconditioners for symmetric saddle point problems,* Appl. Numer. Math 57 (2007), 899–910.

15. Cao, Y., Du, J. and Niu, Q. *Shift-splitting preconditioners for saddle point problems,* J. Comput. Appl. Math. 272 (2014), 239–250.

16. Chen, C. and Ma, C. *A generalized shift-splitting preconditioner for saddle point problems,* Appl. Math. Lett. 43 (2015), 49–55.

17. Davis, T. and Hu, Y. *The university of Florida sparse matrix collection*, ACM Trans. Math. Software 38 (2011), no. 1, Art. 1, 25 pp.

18. Elman, H.C., Silvester, D.J. and Wathen, A.J. *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics,* Numerical Mathematics and Scientific Computation. Oxford University Press, New York, 2005.

19. Feng, T.T., Chen, G.L. and Guo, X.P. *An accelerated SOR-Like method for generalized saddle point problems,* East Asian J. Appl. Math. 8 (2018), 70–81.

20. Golub, G.H., Wu, X. and Yuan, J.-Y. *SOR-like methods for augmented systems*, BIT 41 (2001), 71–85.

21. Gould, N., Orban D. and Rees, T. *Projected Krylov methods for saddle point systems,* SIAM J. Matrix Anal. Appl. 35 (2014), 1329–1343.

22. Guo, C. and Li, J., *A new preconditioner for solving weighted Toeplitz least squares problems,* Math. Appl. (Wuhan) 33 (2020), no. 1, 172–185.

23. Jbilou, K., Sadok, H. and Tinzefte, A. *Oblique projection methods for linear systems with multiple right-hand sides,* Electron Trans. Numer. Anal. 20 (2005), 119–138.

24. Jiang, M.-Q., Cao, Y. and Yao, L.-Q. *On parametrized block triangular preconditioners for generalized saddle point problems,* Appl. Math. Comput. 216 (2010), 1777–1789.

25. Keller, C., Gould, N.I.M. and Wathen, A.J. *Constraint preconditioning for indefinite linear systems,* SIAM J. Matrix Anal. Appl. 21 (2000), 1300–1317.

26. Li, C., Li, Z., Evans, D.J. and Zhang, T. *A note on an SOR-like method for augmented systems,* IMA J. Numer. Anal. 23 (2003), 581–592.

27. Lukšan, L. and Vlčcek, J. *Indefinitely preconditioned inexact Newton method for large sparse equality constrained non-linear programming problems,* Numer. Linear Algebra Appl., 5 (1998), 219–247.

28. Murphy, M.F., Golub, G.H. and Wathen, G.H. *A note on preconditioning for indefinite linear systems,* SIAM J. Sci. Comput 21 (2000), 1969–1972.

29. Nocedal, J. and Wright, S.J. *Numerical optimization,* Springer Series in Operations Research, Springer, Berlin 1999.

30. Pan, J.-Y., Ng, M.K. and Bai, Z.-Z. *New preconditioners for saddle point problems,* Appl. Math. Comput. 172 (2006), 762–771.

31. Perugia, I. and Simoncini, V. *Block-diagonal and indefinite symmetric preconditioners for mixed finite element formulations,* Numer. Linear Algebra Appl. 7 (2000), 585–616.

32. Perugia, I., Simoncini, V. and Arioli, M. *Linear algebra methods in a mixed approximation of magnetostatic problems,* SIAM J. Sci. Comput. 21 (1999), 1085–1101.

33. Rozložník, M. and Simoncini, V. *Krylov subspace methods for saddle point problems with indefinite preconditioning,* SIAM J. Matrix Anal. Appl. 24 (2002), 368–391.

34. Saad, Y. *Iterative Methods for Sparse Linear Systems,* SIAM, Philadelphia, 2003.

35. Salkuyeh, D.K. and Masoudi, M. *A new relaxed HSS preconditioner for saddle point problems,* Numer. Algor. 74 (2017), 781–795.

36. Simoncini, V. *Block triangular preconditioners for symmetric saddle-point problems,* Appl. Numer. Math. 49 (2004), 63–80.

37. Simoncini, V. and Benzi, D.M. *Spectral properties of the Hermitian and skew-Hermitian splitting preconditioner for saddle point problems,* SIAM J. Matrix Anal. Appl. 26 (2004/05), 377–389.

38. Sturler, E.D. and Liesen, J. *Block-diagonal and constraint preconditioners for nonsymmetric indefinite linear systems,* SIAM J. Sci. Comput 26 (2005), 598–619.

39. Taherian, A. and Toutounian, F. *Block GPBi-CG method for solving nonsymmetric linear systems with multiple right-hand sides and its convergence analysis,* J Numer. Algor. (2021), https://doi.org/10.1007/s11075-021-01097-7.

40. Wen, R., Wu R. and Guan, J. *Some generalizations of the new SOR-like method for solving symmetric Saddle point problems,* J. Inequal. Appl. 2018, Paper No. 145, 12 pp.

41. Wesseling, P. *Principles of computational fluid dynamics,* Vol. 29 of Springer Series in Computational Mathematics, Springer, Berlin 2001.

42. Wright, M.H. *Interior methods for constrained optimization,* Acta numerica, 1992, 341–407, Acta Numer., Cambridge Univ. Press, Cambridge, 1992.

43. Wright, S.J. *Primal-dual interior point methods,* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

44. Wu, X.-N., Golub, G.H., Cuminato, J.A. and Yuan, J.-Y. *Symmetric-triangular decomposition and its applications Part II: Preconditioners for indefinite systems.* BIT 48(2008), 139–162.

45. Yun, J.H. *Variants of the Uzawa method for saddle point problem,* Comput. Math. Appl. 65(2013), 1037–1046.

46. Zhang, J. and Dai, H. *Global GPBiCG method for complex non-Hermitian linear systems with multiple right-hand sides,* Comput. Appl. Math. 35 (2016), 171–185.

47. Zheng, Q. and Ma, C. *A new SOR-Like method for the saddle point problems,* Appl. Math. Comput. 233 (2014), 421–429.

How to cite this article

Taherian, A., Toutounian, F. Preconditioned global GPBiCG method for solving saddle point problems with multiple right-hand sides and its convergence analysis. *Iranian Journal of Numerical Analysis and Optimization*, 2022; 12(1): 111-129. doi: 10.22067/ijnao.2021.71036.1044