

Research Article

An Enhancement of the Accuracy of the BiCGStab Method for Solving Linear Systems with Single or Multiple Right-Hand Sides

F. Bouyghf ^{1,2}

¹LMPA, University of Littoral Cote d'Opale, Calais, France

²LabMIA-SI, Ecole Normale Supérieure, Mohammed V University in Rabat, Morocco

Correspondence should be addressed to F. Bouyghf; fatimabouyghf3@gmail.com

Received 4 May 2023; Revised 17 August 2023; Accepted 23 August 2023; Published 13 September 2023

Academic Editor: Pedro Alonso

Copyright © 2023 F. Bouyghf. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we present a technique to improve the convergence of the biconjugate gradient stabilized (BiCGStab) method. This method was developed by Van der Vorst for solving nonsymmetric linear systems with a single right-hand side. The global and block versions of the BiCGStab method have been proposed for solving nonsymmetric linear systems with multiple right-hand sides. Using orthogonal projectors to minimize the residual norm in each step, we get an enhancement of the convergence of each version of the BiCGStab method. The considered methods are BiCGStab, global BiCGStab, and block BiCGStab methods, noted, respectively, as GI-BiCGStab and BI-BiCGStab. To show the performance of our enhanced algorithms, we compare them with the standard, global, and block versions of the well-known generalized minimal residual method (GMRES).

1. Introduction

The aim of the BiCGStab method studied in this paper is to solve the following nonsymmetric linear system

$$Ax = b, \quad (1)$$

where A is a nonsingular matrix in $\mathbb{R}^{N \times N}$ and the vectors b and x are in \mathbb{R}^N . Problems such as (1) occur in most applications of scientific computing, engineering applications, and Navier-Stokes equations in computational fluid dynamics and structural mechanic computations based on finite element analysis. If the order N of the matrix A is small, we can solve (1) using direct methods, but if N is large, direct methods can be prohibitively expensive both in terms of memory and time. So, iterative methods become appealing. These methods include the Krylov subspace methods. It is true that current methods give us interesting computational performance, but it is important to realize that there is no single method that can solve every linear system with desirable accuracy. Because it all depends on the conditioning number of the problem matrix A and some characteristics of the matrix, i.e., its symmetric and whether it is positive

definite or not. Many experiments are given in the recent book [1] to show this remark. However, a number of factors can influence the choice of the method, including the conditioning of the matrix A and the number of nonzero values. In literature, there are many problems of linear systems with a sparse matrix which are obtained from real applications (heat transfer, fluid flow, mass transport, etc.) by using a numerical strategy for solving partial differential equations (finite difference, finite volume, and finite element methods) [2, 3]. The difference between all the existing methods for solving linear systems is in the level of accuracy, turnaround time, and storage. For system (1) with a symmetric positive definite matrix, we use the conjugate gradient method (CG). If A is a symmetric matrix, we usually use the MINRES. Finally, the nonsymmetric case can be solved using GMRES, developed as the most popular and the most optimal in terms of precision but suffers from storage problems. The nonsymmetric case can also be solved using some short-recurrence Krylov subspace methods. We find for example the BiCG, CGS, and BiCGStab methods. These methods are derived from the extension of the CG in the nonsymmetric case. In this work, we focus on the nonsymmetric case, especially on the BiCGStab method as the most stable

method compared with the BiCG and CGS methods. For more details about all the Krylov subspace methods, see [4] and the current book [1].

The BiCGStab method has been developed in [5] for solving (1). The global version of the BiCGStab method has been given in [6], and the block version of this method has been given in [7] for solving linear systems with several right-hand sides. Using orthogonal projectors and a technique given in [8, 9] to improve the convergence of the IDR method [10, 11], and some Krylov methods (BiCG and CGS), we give an enhancement of the convergence of the BiCGStab method.

The rest of this paper is organized as follows: In the next section, we recall the algorithm of the BiCGStab method [5]. Then, we propose an improvement in the convergence of this algorithm using orthogonal projectors. A partial and full improvement of the BiCGStab method is proposed and will be called PEnha-BiCGStab(k) and FEnha-BiCGStab, respectively. In Section 3, we focus on the solution of linear systems with multiple right-hand sides. We will recall the global version of the BiCGStab [6], which will be called the global BiCGStab (Gl-BiCGStab) method. We will also propose two improvements to this method, partial and full improvements, which will be called partial and full enhancements of the global BiCGStab and denoted by PEnha-Gl-BiCGStab(k) and FEnha-Gl-BiCGStab, respectively. In Section 4, we will recall the block version of the BiCGStab (Bl-BiCGStab) method. We will also propose two improvements to this method, partial and full improvements, which will be called partial and full enhancements of the block BiCGStab and denoted by PEnha-Bl-BiCGStab(k) and FEnha-Bl-BiCGStab, respectively.

In Section 5, we will present some numerical experiments to compare the proposed algorithms with the well-known GMRES [12], the global GMRES (Gl-GMRES) [13], and the block GMRES (Bl-GMRES) [14] methods.

Throughout this article, all vectors and matrices are assumed to be real, and the following notation is used. First, M^T represents the transpose of any matrix M . For two vectors x and y in \mathbb{R}^N , the inner product is $\langle x, y \rangle_2 = x^T y$, with $\|x\|_2 = \sqrt{\langle x, x \rangle_2}$ the Euclidean norm. In the block and global cases, we consider two matrices X and Y in $\mathbb{R}^{N \times m}$. The inner product is defined by $\langle X, Y \rangle_F = \text{Tr}(X^T Y)$, where $\text{Tr}(Z)$ denotes the trace of a square matrix Z . Moreover, the associated norm is the Frobenius norm noted $\|\cdot\|_F$. We denote by I the identity matrix of order N .

2. BiCGStab Method and Its Enhancement

The biconjugate gradient stabilized (BiCGStab) algorithm has been developed for solving nonsymmetric linear systems (1). This algorithm has been given from the conjugate gradient squared (CGS) algorithm of Sonneveld [15], which is obtained from the biconjugate gradient (BiCG) algorithm (see [4, 16]). This last algorithm was obtained by using the Lanczos biorthogonalization (see [4, 16]). All these methods are Krylov subspace methods for solving linear systems. So, in this section, we give some theoretical background and

some preliminary results. For an initial guess x_0 , we associate a residual vector $r_0 = b - Ax_0$.

Definition 1. We define the Krylov subspace of order k associated to the matrix A and vector r_0 by

$$\mathcal{K}_k^s(A, r_0) = \text{Span}\{r_0, Ar_0, \dots, A^{k-1}r_0\}. \quad (2)$$

Classical Krylov subspace methods compute the approximate solution x_k and its corresponding residual $r_k = b - Ax_k$ such that

$$x_k - x_0 \in \mathcal{K}_k^s(A, r_0), \text{ and } r_k = \Phi_k(A)r_0 \text{ for } k = 1, \dots, m, \quad (3)$$

where Φ_k is a polynomial of degree k . Let

$$\psi_m(\xi) = \sigma_0 + \sigma_1\xi + \dots + \sigma_m\xi^m = \sum_{i=0}^m \sigma_i \xi^i, \text{ with } \psi_m(0) = 1 \quad (4)$$

be the minimal polynomial of the matrix A with respect to the vector r_0 , i.e.,

$$\begin{aligned} \psi_m(A)r_0 &= \sum_{i=0}^m \sigma_i A^i r_0 = 0, \\ m &= \min \left\{ k \text{ such that } \sum_{i=0}^k \sigma_i A^i r_0 = 0, \text{ with } \sigma_0 = 1 \right\}. \end{aligned} \quad (5)$$

If the considered method converges after m iterations, the polynomial matrix $\Phi_k(A)$ can be written as follows:

$$\Phi_k(A) = \phi_k(A)\psi_k(A). \quad (6)$$

The way in which the coefficients of these two polynomials are calculated gives us the different variants of the Krylov subspace methods. For example, in the GMRES method

$$\begin{aligned} \phi_k(A) &= I, \\ \psi_k(A) &= I - W_k W_k^\dagger, \end{aligned} \quad (7)$$

with $W_k = [Ar_0, A^2r_0, \dots, A^k r_0]$ and $W_k^\dagger = (W_k^T W_k)^{-1} W_k^T$ is the pseudoinverse of the matrix W_k . The BiCG method is obtained if we consider the following choice:

$$\begin{aligned} \phi_k(A) &= I, \\ \psi_k(A) &= p_k(A) = I - W_k \left(\tilde{W}_k^T W_k \right)^{-1} \tilde{W}_k^T, \end{aligned} \quad (8)$$

with $\tilde{W} = [r_0, A^T r_0, \dots, (A^T)^{k-1} r_0]$.

We remark that $H_k = I - W_k W_k^\dagger$ is an orthogonal projector. Then, the associated residual vector of the GMRES method is defined by an orthogonal projector, hence the optimal property of this method.

The CGS method is developed to avoid the calculation of the transpose of the matrix A in the BiCG method, and then the residual associated to the CGS method is given as follows:

$$r_k^{\text{CGS}} = p_k^2(A)r_0. \quad (9)$$

The CGS algorithm is based on squaring the residual polynomial, and, in cases of irregular convergence, this may lead to a substantial buildup of rounding errors or possibly even overflow. The BiCGStab algorithm is a variation of the CGS method which was developed to remedy this difficulty. Instead of seeking a method which delivers a residual vector of the form (9), BiCGStab produces iterates whose residual vectors are of the form

$$r_k^{\text{BiCGStab}} = \Psi_k(A)p_k(A)r_0, \quad (10)$$

in which, as before, $p_k(A)$ is the residual polynomial associated with the BiCG algorithm and $\Psi_k(A)$ is a new polynomial which is defined recursively at each step with the goal of “stabilizing” or “smoothing” the convergence behavior of the original algorithm. Specifically, $\Psi_k(A)$ is defined by the simple recurrence

$$\Psi_{k+1}(A) = (I - \omega_k A)\Psi_k(A), \quad (11)$$

with ω_k is determined by minimizing the norm of the residual. Based on this fact of minimization, our idea of improving the convergence of the BiCGStab method comes from the fact that we can give another form to these polynomials to improve convergence, using only the data that we have to keep the same storage and computation time.

The BiCGStab method is summarized by the following algorithm.

We will propose an improvement to the convergence of the BiCGStab method. Two enhancements of this method are studied: the first one will be called partial enhancement, denoted by PENha-BiCGStab(k), and the second one will be called full enhancement, denoted by FENha-BiCGStab. We propose to improve the convergence of the BiCGStab method by using the following well-known result.

Proposition 2. Consider the orthogonal projector

$$Q_i = I - \mathcal{Z}_i \mathcal{Z}_i^\dagger, \quad (12)$$

where the rectangular matrix Z_i is a full rank matrix in $\mathbb{R}^{N \times l}$ and

$$Z_i^\dagger = (Z_i^T Z_i)^{-1} Z_i^T \quad (13)$$

is its pseudoinverse (Moore-Penrose) (for more details of the pseudoinverse, see [17]). Applying the projector Q_i to any vector $r \in \mathbb{R}^N$, we obtain a new vector, which we denote by

$$r^{\text{Enha}} = Q_i r. \quad (14)$$

Then, we have

$$\|r^{\text{Enha}}\|_2 \leq \|r\|_2. \quad (15)$$

The proof of this proposition is given in [4] page 38.

To improve the convergence of an iterative method for solving linear systems, it is necessary to minimize and decrease the norm of its residual in as few iterations as possible. Then, by invoking Proposition 2 with the residual vector r_i , we obtain an improvement in the accuracy and stability of the BiCGStab algorithm. Thus, we will apply an orthogonal projector Q_k to the residual of this method to obtain a new residual with a smaller norm. Furthermore, to avoid a storage problem, we use the k pairs of vectors already calculated in the BiCGStab method to construct the orthogonal projector.

The partial enhancement of the convergence of the BiCGStab method (PENha-BiCGStab(k)) is given by choosing Z_k equal to the k last pairs of vectors $[v_i, t_i]$, and by adding to line 10 in Algorithm 1 the following instructions

- (1) $Z1 = [p_{i-k+1}, s_{i-k+1}, p_{i-k+2}, s_{i-k+2}, \dots, p_i, s_i]$
- (2) $Z_k = AZ1 = [v_{i-k+1}, t_{i-k+1}, v_{i-k+2}, t_{i-k+2}, \dots, v_i, t_i]$
- (3) $Z = Z_k^\dagger r_i$
- (4) $x_i^{\text{PENha}} = x_i + Z1Z$
- (5) $r_i^{\text{PENha}} = r_i - Z_k Z$

The full enhancement of the convergence of the BiCGStab (FENha-BiCGStab) method is defined by choosing \mathcal{Z} equal to the all last pairs of vectors

$$[v_0, t_0, v_1, t_1, \dots, v_i, t_i], \quad (16)$$

and by adding to line 10 in Algorithm 1 the following instructions

- (1) $Z1 = [p_0, s_0, p_1, s_1, \dots, p_i, s_i]$
- (2) $Z = AZ1 = [v_0, t_0, v_1, t_1, \dots, v_i, t_i]$
- (3) $Z2 = Z^\dagger r_i$
- (4) $x_i^{\text{FENha}} = x_i + Z1Z2$
- (5) $r_i^{\text{FENha}} = r_i - Z Z2$

3. Global BiCGStab Method and Its Enhancement

In this section, we consider the solution of large and sparse nonsymmetric systems with multiple right-hand sides of the form

$$AX = B, \quad (17)$$

```

1.  $x_0 \in \mathbb{R}^N$  guess initial vector;
2.  $r_0 = b - Ax_0$ ,  $p_0 = r_0$ ,  $\tilde{r}_0 = r_0$ ;
3. for  $i = 0, 1, 2 \dots$ ;
4.    $v_i = Ap_i$ ;
5.    $\alpha_i = \langle \tilde{r}_0, r_i \rangle_2 / \langle \tilde{r}_0, v_i \rangle_2$ ;
6.    $s_i = r_i - \alpha_i v_i$ ;
7.    $t_i = As_i$ ;
8.    $\omega_i = \langle t_i, s_i \rangle_2 / \langle t_i, t_i \rangle_2$ ;
9.    $x_{i+1} = x_i + \alpha_i p_i + \omega_i s_i$ ;
10.   $r_{i+1} = s_i - \omega_i t_i$ ;
11.   $\beta_i = -\langle \tilde{r}_0, t_i \rangle_2 / \langle \tilde{r}_0, v_i \rangle_2$ ;
12.   $p_{i+1} = r_{i+1} + \beta_i (p_i - \omega_i v_i)$ ;
13. end for.

```

ALGORITHM 1: Biconjugate gradient stabilized (BiCGStab) [5].

where the coefficient matrix A is a nonsingular real matrix of order N ,

$$\begin{aligned} X &= [x_1 \ x_2 \ \dots \ x_m], \\ B &= [b_1 \ b_2 \ \dots \ b_m] \in \mathbb{R}^{N \times m}, \end{aligned} \quad (18)$$

with $m \ll N$. One class of solvers for solving problem (17) are the global methods, which are based on the use of a global projection process onto a matrix (global) Krylov subspace, including global the FOM and GMRES methods [13], global BCG and BiCGStab methods [10], and global Hessenberg and CMRH methods [18].

The other class is that of block solvers which are much more efficient when the matrix A is relatively dense. The first block solvers are the block conjugate gradient (BI-CG) and block biconjugate gradient (BI-BiCG) methods proposed in [19]; for nonsymmetric problems, the block generalized minimal residual (BI-GMRES) algorithm [14], the block quasiminimum residual (BI-QMR) algorithm [20], the block BiCGStab (BI-BiCGStab) algorithm [10], and the block Lanczos method [7] have been developed.

In what follows, we recall the global biconjugate gradient stabilized (GI-BiCGStab) algorithm.

We will propose an improvement of the convergence of the GI-BiCGStab method. Two enhancements of this method are studied: the first one will be called partial global enhancement, denoted by PEnha-GI-BiCGStab(k), and the second one will be called full global enhancement, denoted by FEnha-GI-BiCGStab. We propose to improve the convergence of the GI-BiCGStab method by using the following well-known result.

Proposition 3. Consider the orthogonal projector

$$Q_l = I - \mathcal{Z}_l \mathcal{Z}_l^\dagger, \quad (19)$$

where the rectangular matrix \mathcal{Z}_l is a full rank matrix in $\mathbb{R}^{N \times m}$ and

$$\mathcal{Z}_l^\dagger = (\mathcal{Z}_l^T \mathcal{Z}_l)^{-1} \mathcal{Z}_l^T \quad (20)$$

is its pseudoinverse (Moore-Penrose) (for more details of the pseudoinverse, see [17]). Applying the projector Q_l to any matrix $R \in \mathbb{R}^{N \times m}$, we obtain a new residual, which we denote by

$$R^{\text{Enha}} = Q_l R. \quad (21)$$

Then, we have

$$\|R^{\text{Enha}}\|_F \leq \|R\|_F. \quad (22)$$

The proof of this proposition is similar to that of the proposition in the standard case. By invoking Proposition 3 with the residual matrix R_i , we obtain an improvement of the convergence GI-BiCGStab algorithm. Thus, we will apply an orthogonal projector Q_k to the residual of this method and then to minimize its norm. To avoid a storage problem, we use the k pairs of matrices already calculated in the GI-BiCGStab method to construct the orthogonal projector.

The partial enhancement of the convergence of the GI-BiCGStab method (PEnha-GI-BiCGStab(k)) is given by choosing \mathcal{Z}_k equal to the k last pairs of matrices $[V_i, T_i]$ and by adding to line 10 in Algorithm 2 the following instructions

- (1) $\mathcal{Z}1 = [P_{i-k+1}, S_{i-k+1}, P_{i-k+2}, S_{i-k+2}, \dots, P_i, S_i]$
- (2) $\mathcal{Z}_k = A\mathcal{Z}1 = [V_{i-k+1}, T_{i-k+1}, V_{i-k+2}, T_{i-k+2}, \dots, V_i, T_i]$
- (3) $\mathcal{Z}2 = \mathcal{Z}_k^\dagger R_i$
- (4) $X_i^{\text{PEnha}} = X_i + \mathcal{Z}1\mathcal{Z}2$
- (5) $R_i^{\text{PEnha}} = R_i - \mathcal{Z}_k\mathcal{Z}2$

The full enhancement of the convergence of the GI-BiCGStab (FEnha-GI-BiCGStab) method is defined by choosing \mathcal{Z}_l equal to the all last pairs of matrices

$$[P_0, S_0, P_1, S_1, \dots, P_i, S_i], \quad (23)$$

and by adding to line 10 in Algorithm 1 the following instructions

- (1) $\mathcal{Z}1 = [P_0, S_0, P_1, S_1, \dots, P_i, S_i]$
- (2) $\mathcal{Z} = A\mathcal{Z}1 = [V_0, T_0, V_1, T_1, \dots, V_i, T_i]$
- (3) $\mathcal{Z}2 = \mathcal{Z}^\dagger R_i$
- (4) $X_i^{\text{FEnha}} = X_i + \mathcal{Z}1\mathcal{Z}2$
- (5) $R_i^{\text{FEnha}} = R_i - \mathcal{Z}\mathcal{Z}2$

4. Block BiCGStab Method and Its Enhancement

As for the GI-BiCGStab method, we will propose an improvement of the convergence of the block BiCGStab method by applying Proposition 3. Two enhancements of this method are proposed: the first one will be called the block partial enhancement, denoted by PEnha-BI-BiCGStab(k), and the

```

1.  $X_0 \in \mathbb{R}^{N \times m}$  guess initial matrix;
2.  $R_0 = B - AX_0, P_0 = R_0, \tilde{R}_0 = R_0$ ;
3. for  $i = 0, 1, 2 \dots$ ;
4.    $V_i = AP_i$ ;
5.    $\alpha_i = \langle \tilde{R}_0, R_i \rangle_F / \langle \tilde{R}_0, V_i \rangle_F$ ;
6.    $S_i = R_i - \alpha_i V_i$ ;
7.    $T_i = AS_i$ ;
8.    $\omega_i = \langle T_i, S_i \rangle_F / \langle T_i, T_i \rangle_F$ ;
9.    $X_{i+1} = X_i + \alpha_i P_i + \omega_i S_i$ ;
10.   $R_{i+1} = S_i - \omega_i T_i$ ;
11.   $\beta_i = -\langle \tilde{R}_0, T_i \rangle_F / \langle \tilde{R}_0, V_i \rangle_F$ ;
12.   $P_{i+1} = R_{i+1} + \beta_i (P_i - \omega_i V_i)$ ;
13. end for.

```

ALGORITHM 2: Global biconjugate gradient stabilized (GI-BiCGStab) [6].

second one will be called block full enhancement, denoted by FEnha-BI-BiCGStab.

First, let us recall the block version of BiCGStab (BI-BiCGStab).

By invoking Proposition 3 with the residual vector R_i , we obtain an improvement of the BI-BiCGStab algorithm. Thus, we will apply an orthogonal projector Q_k to the residual of this method. To avoid a storage problem, we use the k pairs of matrices already calculated in the BI-BiCGStab method to construct the orthogonal projector.

The partial enhancement of the convergence of the BI-BiCGStab method (PEnha-BI-BiCGStab(k)) is given by choosing \mathcal{Z}_k equal to the k last pairs of matrices $[P_i, S_i]$ and by adding to line 10 in Algorithm 3 the following instructions

- (1) $\mathcal{Z}_k = [P_{i-k+1}, S_{i-k+1}, P_{i-k+2}, S_{i-k+2}, \dots, P_i, S_i]$
- (2) $\mathcal{Z}1 = A\mathcal{Z}_k$
- (3) $\mathcal{Z}2 = \mathcal{Z}1^\dagger R_i$
- (4) $X_i^{\text{PEnha}} = X_i + \mathcal{Z}_k \mathcal{Z}2$
- (5) $R_i^{\text{PEnha}} = R_i - \mathcal{Z}1 \mathcal{Z}2$

Remark that $\mathcal{Z}1 = A\mathcal{Z}_k$ is given by Algorithm3.

$$\begin{aligned} \mathcal{Z}1 = A\mathcal{Z}_k &= A[P_{i-k+1}, S_{i-k+1}, P_{i-k+2}, S_{i-k+2}, \dots, P_i, S_i] \\ &= [V_{i-k+1}, T_{i-k+1}, V_{i-k+2}, T_{i-k+2}, \dots, V_i, T_i]. \end{aligned} \quad (24)$$

The full enhancement of the convergence of the BI-BiCGStab (FEnha-BI-BiCGStab) method is defined by choosing \mathcal{Z}_l equal to the all last pairs of matrices

$$[P_0, S_0, P_1, S_1, \dots, P_i, S_i], \quad (25)$$

and by adding to line 10 in Algorithm 3 the following instructions

- (1) $\mathcal{Z} = [P_0, S_0, P_1, S_1, \dots, P_i, S_i]$

$$(2) \mathcal{Z}1 = A\mathcal{Z}$$

$$(3) \mathcal{Z}2 = \mathcal{Z}1^\dagger R_i$$

$$(4) X_i^{\text{FEnha}} = X_i + \mathcal{Z} \mathcal{Z}2$$

$$(5) R_i^{\text{FEnha}} = R_i - \mathcal{Z}1 \mathcal{Z}2$$

We notice that at each iteration, we compute two matrices P_i and S_i . The choice of the matrix Z_i is crucial because we have $2km$ vectors in each matrix for the partial enhancement instead of $2k$ like in the standard case. Then, the convergence is clear in these two cases. In other words, if the number of vectors that we use to construct the orthogonal projector is large, we obtain a clearer improvement in accuracy and stability, and then the convergence will be faster.

5. Numerical Examples

In this section, we consider the following convection-diffusion equation

$$\begin{aligned} -\Delta u - \alpha \cdot \nabla u - \beta u &= f, \quad \text{in } \Omega, \\ u &= 0, \quad \text{on } \partial\Omega, \end{aligned} \quad (26)$$

where $\Omega = (0, 1)^3$ and $\alpha = (\alpha_x, \alpha_y, \alpha_z)^T \in \mathbb{R}^3$. The discretization of this equation is done via centered finite differences with the standard 7-point stencil in three dimensions. (For more details about the 7-point stencil, see [21].) The obtained matrix is sparse. Then, we have used an example where the existing methods converge. We have shown numerically that the new methods give an improvement to this convergence. That is what we have shown theoretically. For all the examples, we choose $\alpha = (0.5, 0.5, 0.5)^T$, $\beta = 5$, and

$$\begin{aligned} N_x &= 30, \\ N_y &= 20, \\ N_z &= 20. \end{aligned} \quad (27)$$

The order of the system is $N = N_x \times N_y \times N_z = 12\,000$.

```

1.  $X_0 \in \mathbb{R}^{N \times m}$  guess initial matrix;
2.  $R_0 = B - AX_0$ ,  $P_0 = R_0$ ,  $\tilde{R}_0 = R_0$ ;
3. for  $i = 0, 1, 2 \dots$ ;
4.    $V_i = AP_i$ ;
5.    $\alpha_i = (\tilde{R}_0^T V_i)^{-1} (\tilde{R}_0^T R_i)$ ;
6.    $S_i = R_i - V_i \alpha_i$ ;
7.    $T_i = AS_i$ ;
8.    $\omega_i = \langle T_i, S_i \rangle_F / \langle T_i, T_i \rangle_F$ ;
9.    $X_{i+1} = X_i + \alpha_i P_i + \omega_i S_i$ ;
10.   $R_{i+1} = S_i - \omega_i T_i$ ;
11.   $\beta_i = -(\tilde{R}_0^T V_i)^{-1} (\tilde{R}_0^T T_i)$ ;
12.   $P_{i+1} = R_{i+1} + (P_i - \omega_i V_i) \beta_i$ ;
13. end for.

```

ALGORITHM 3: Block biconjugate gradient stabilized (Bl-BiCGStab) [10].

TABLE 1: Comparison of CPU time in the standard case.

| Method | CPU time (s) |
|--------------------|-----------------------|
| BiCGStab | 5.01×10^{-1} |
| PEنها-BiCGStab(5) | 5.48×10^{-1} |
| PEنها-BiCGStab(12) | 6.15×10^{-1} |
| FENها-BiCGStab | 7.45×10^{-1} |
| GMRES | 5.89×10^{-1} |

TABLE 2: Comparison of CPU time in the global case.

| Method | CPU time (s) |
|-----------------------|--------------------|
| Gl-BiCGStab | 3.69×10^0 |
| PEنها-Gl-BiCGStab(5) | 5.69×10^0 |
| PEنها-Gl-BiCGStab(12) | 7.36×10^0 |
| FENها-Gl-BiCGStab | 8.05×10^0 |
| GMRES | 1.46×10^0 |

TABLE 3: Comparison of CPU time in the block case.

| Method | CPU time (s) |
|-----------------------|-----------------------|
| Bl-BiCGStab | 1.90×10^0 |
| PEنها-Bl-BiCGStab(5) | 4.01×10^0 |
| PEنها-Bl-BiCGStab(12) | 7.01×10^0 |
| FENها-Bl-BiCGStab | 8.25×10^0 |
| GMRES | 5.80×10^{-1} |

In general, to compare two iterative methods in terms of convergence accuracy and stability, we need to compare the history of the norm of the residual and error vectors. Numerically, sometimes, even if the residual norm gives calculation results, the error norm does not. That is why it is necessary to check this also for the new methods too. We define the residual norm $\|r_k\|_2$ and the error norm $\|e_k\|_2$ in the standard case as follows:

$$\|r_k\|_2 = \sqrt{\langle r_k, r_k \rangle} = \sqrt{r_k^T r_k},$$

$$\|e_k\|_2 = \|x_k - \text{xtrue}\|_2 = \sqrt{(x_k - \text{xtrue})^T (x_k - \text{xtrue})}. \quad (28)$$

For the global and block case, we use the following formulas:

$$\|R_k\|_F = \sqrt{\langle R_k, R_k \rangle_F} = \sqrt{\text{trace}(R_k^T R_k)},$$

$$\|E_k\|_F = \|X_k - \text{Xtrue}\|_F = \sqrt{\langle X_k - \text{Xtrue}, X_k - \text{Xtrue} \rangle_F}$$

$$= \sqrt{\text{trace}((X_k - \text{Xtrue})^T (X_k - \text{Xtrue}))}. \quad (29)$$

We would also compare the execution time of each method, but in this case, the difference between the improved and unimproved methods is negligible (see Tables 1–3). To minimize the norm of the residual, we have used the matrices and vectors already computed at each iteration.

To illustrate the efficiency of our technique, we compare the BiCGStab and its enhancement methods for systems with single right-hand sides, given by Algorithm 1, with the GMRES methods. Then, we apply the classical BiCGStab and the new enhanced BiCGStab(k) (partial and full enhancements of BiCGStab), denoted by PEنها-BiCGStab(k) and FENها-BiCGStab for $k = 5$ and $k = 12$, we give the curves of residual norms and error norms. For these methods, the right-hand b of the system is chosen as follows:

$$\text{xtrue} = \text{rand}(N, 1), b = \text{Axs}, \quad (30)$$

where xs is the solution of the considered system, and the rand function creates a random N -vector for xs , with coefficients uniformly distributed in $[0, 1]$, and the initial guess was taken to be zero. In this case, the tests were stopped as soon as $\|r_n\|/\|b\| \leq 10^{-10}$. Figures 1 and 2 illustrate the comparison of these algorithms for residual and error norms, respectively. Remark that the function randn can be also

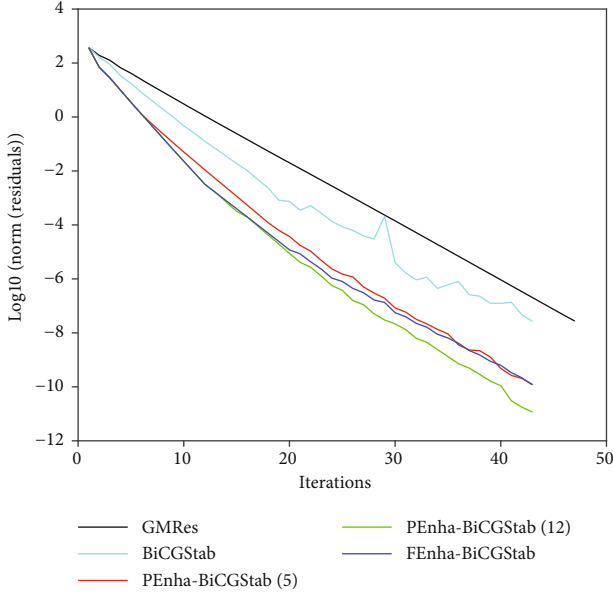


FIGURE 1: The comparison between residual norms of BiCGStab, its enhancements, and GMRES methods for $k = 5$ and $k = 12$.

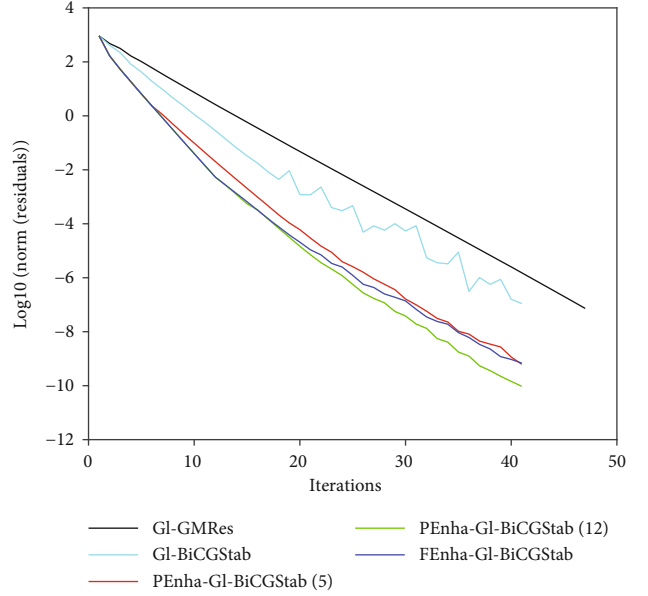


FIGURE 3: Comparison between residual norms of GI-BiCGStab, its enhancements, and GI-GMRES for $k = 5$, $k = 12$, and for $m = 6$.

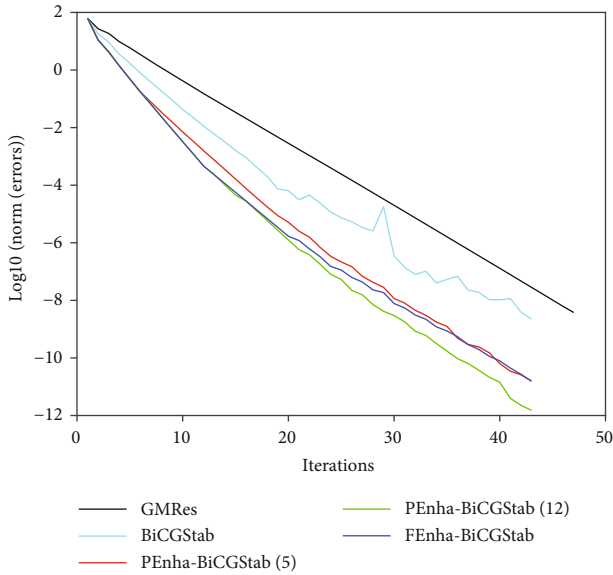


FIGURE 2: The comparison between error norms of BiCGStab, its enhancements, and GMRES methods for $k = 5$ and $k = 12$.

used, which creates a random matrix or vector with real random coefficients.

For global and block methods, the right hand B of (17) is chosen as follows:

$$XS = \text{rand}(N, m) B = AXS, \quad (31)$$

the initial guess matrix equal to zeros(N, m). The tests were stopped as soon as

$$\|R_n\|_F / \|B\|_F \leq 10^{-10}. \quad (32)$$

For the global case, we compare the global BiCGStab (GI-BiCGStab) and its enhancements, partial enhancement of global BiCGStab(k) (PENha-GI-BiCGStab(k)), and full enhancement of global BiCGStab (FENha-GI-BiCGStab) for $k = 5$, $k = 12$, and for $m = 6$, with the GI-GMRES method. Figures 3 and 4 give this comparison of residual and error norms, respectively.

For the block case, we compare the block BiCGStab (BI-BiCGStab) and its enhancements, partial enhancement of block BiCGStab(s) (PENha-BI-BiCGStab(k)), and full enhancement of block BiCGStab (FENha-BI-BiCGStab) for $k = 5$, $k = 12$, and for $m = 6$, with the BI-GMRES method. Figures 5 and 6 show this comparison of residual and error norms, respectively.

5.1. BiCGStab Method. From the curves of Figures 1 and 2, we remark that the residual norm and the error norm of the enhanced algorithms decrease quickly compared with the existing methods. Furthermore, for $k = 12$, the convergence is clearly faster than the case when $k = 5$.

5.2. Global BiCGStab Method. In Figures 3 and 4, we observe that the enhanced solvers FENha-GI-BiCGStab and PENha-GI-BiCGStab(k) give the best result. In this example, we can remark also that for $k = 12$, the PENha-GI-BiCGStab(k) is faster. A slight improvement of stability is also observed. For $k = 5$, the enhanced method is still better than the GI-BiCGStab and GI-GMRES methods.

5.3. Block BiCGStab Method. The curves clearly show that the accuracy of the block-enhanced methods is better than that of the BI-BiCGStab and BI-GMRES. And this is totally normal, as we have in the global and block cases, km vectors to construct an orthogonal projector. The convergence is faster as the number of vectors increases. Hence, the

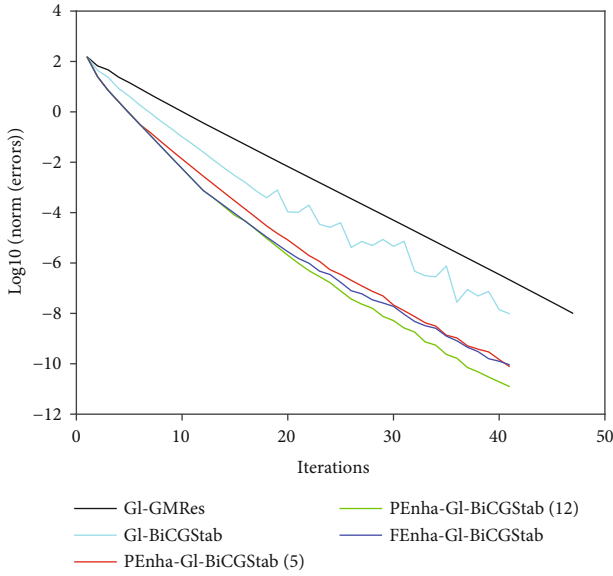


FIGURE 4: Comparison between error norms of GI-BiCGStab, its enhancements, and GI-GMRES for $k = 5$, $k = 12$, and for $m = 6$.

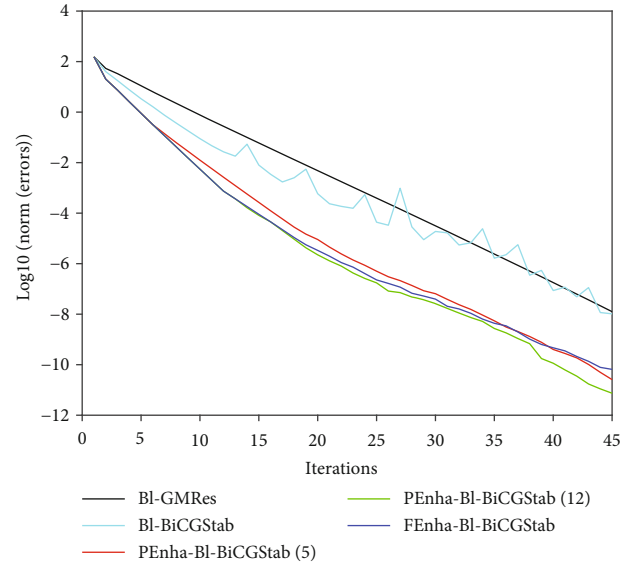


FIGURE 6: Comparison between error norms of BI-BiCGStab, its enhancements, and BI-GMRES for $k = 5$, $k = 12$, and for $m = 6$.

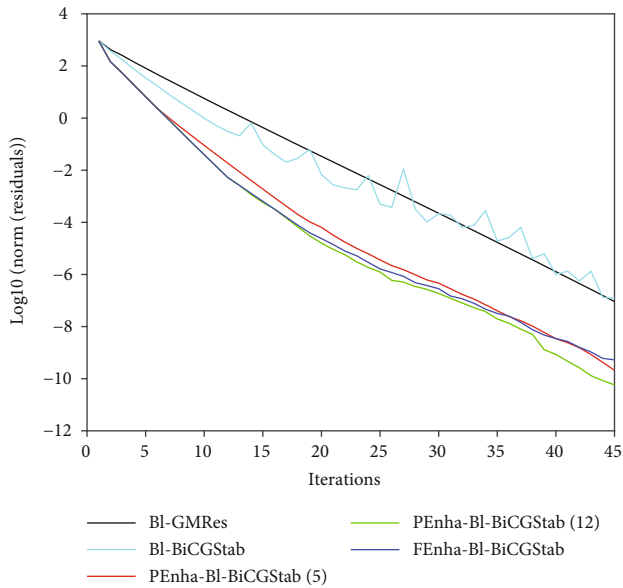


FIGURE 5: Comparison between residual norms of BI-BiCGStab, its enhancements, and BI-GMRES for $k = 5$, $k = 12$, and for $m = 6$.

importance of choosing matrices Z_i for constructing the orthogonal projectors in the global and block cases. The main aim of this paper is to improve the convergence of the BiCGStab method in terms of accuracy and stability while keeping the same storage properties and computation time. Tables show that when using this technique, the difference in computation time is negligible between the BiCGStab method and the improved BiCGStab method in all three cases. Here, we have also included the GMRES method as the most optimal, to show that even though we lose a little in terms of computation time, we still get a significant improvement in accuracy. So, in Tables 1–3, we focus

on the comparison between the BiCGStab method and its improved version.

6. Conclusion

In this paper, we proposed a new technique to improve the convergence behavior of the BiCGStab method for the standard, global, and block cases. Using orthogonal projectors, we have proposed an enhancement of the convergence of the BiCGStab method. The orthogonal projectors are constructed using vectors and matrices already computed in each method to avoid storage problems and then keep the advantage of storage of BiCGStab in all cases. Numerically, we see that for all three cases, the enhanced algorithms of BiCGStab are more efficient, and they converge faster than the BiCGStab and GMRES methods with negligible deference to the turnaround time.

Data Availability

The data that support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that they have no conflicts of interest.

Acknowledgments

This work was supported by the Pure and Applied Mathematics Laboratory at the University of Littoral Côte d'Opale. Open Access funding is enabled and organized by COUPE-RIN CY23.

References

- [1] G. Meurant and J. D. Tebbens, “Krylov methods for nonsymmetric linear systems—from theory to computation,” *Springer Series in Computational Mathematics*, vol. 57, 2020.
- [2] I. S. Duff, R. G. Grimes, and J. G. Lewis, “Sparse matrix test problems,” *ACM Transactions on Mathematical Software*, vol. 15, no. 1, pp. 1–14, 1989.
- [3] M. Market, *Collection of Test Matrices* <http://math.nist.gov/MatrixMarket/index.html>.
- [4] Y. Saad, *Iterative Methods for Sparse Linear System*, SIAM, Philadelphia, Second edition, 2003.
- [5] H. A. Van Der Vorst, “Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems,” *SIAM Journal on Scientific and Statistical Computing*, vol. 13, no. 2, pp. 631–644, 1992.
- [6] K. Jbilou, H. Sadok, and A. Tinzeft, “Oblique projection methods for linear systems with multiple right-hand sides,” *Electronic Transactions on Numerical Analysis*, vol. 20, pp. 119–138, 2005.
- [7] A. El Guennouni, K. Jbilou, and H. Sadok, “The block Lanczos method for linear systems with multiple right-hand sides,” *Applied Numerical Mathematics*, vol. 51, no. 2-3, pp. 243–256, 2004.
- [8] F. Bouyghf, A. Messaoudi, and H. Sadok, “An enhancement of the convergence of IDR method,” *ETNA—Electronic Transactions on Numerical Analysis*, vol. 58, pp. 470–485, 2022.
- [9] F. Bouyghf, A. Messaoudi, and H. Sadok, *A unified approach to Krylov subspace methods for solving linear systems*, Numerical Algorithms, 2023.
- [10] A. El Guennouni, K. Jbilou, and H. Sadok, “A block version of BiCGStab for linear systems with multiple right-hand sides,” *Electronic Transactions on Numerical Analysis*, vol. 16, pp. 129–142, 2003.
- [11] P. Sonneveld and M. B. Van Gijzen, “IDR(s): a family of simple and fast algorithms for solving large nonsymmetric systems of linear equations,” *SIAM Journal on Scientific Computing*, vol. 31, no. 2, pp. 1035–1062, 2009.
- [12] Y. Saad and M. H. Schultz, “GMRES: a generalized minimal residual algorithm for solving non-symmetric linear systems,” *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 3, pp. 856–869, 1986.
- [13] K. Jbilou, A. Messaoudi, and H. Sadok, “Global FOM and GMRES algorithms for matrix equations,” *Applied Numerical Mathematics*, vol. 31, no. 1, pp. 49–63, 1999.
- [14] B. Vital, *Etude de quelques méthodes de résolution de problèmes linéaires de grande taille sur multiprocesseur*, [Ph.D. thesis], Université de Rennes, Rennes, France, 1990.
- [15] P. Sonneveld, “CGS: a fast Lanczos-type solver for nonsymmetric linear systems,” *SIAM Journal on Scientific and Statistical Computing*, vol. 10, no. 1, pp. 36–52, 1989.
- [16] H. A. Van der Vorst, *Iterative Krylov Methods for Large Linear Systems*, Cambridge University Press, Cambridge, 2009.
- [17] A. Ben-Israel and T. N. E. Greville, *Generalized Inverse, Theory and Application*, Canadian Mathematical Society, Second edition, 2003.
- [18] M. El Heyouni, “The global Hessenberg and global CMRH methods for linear systems with multiple right-hand sides,” *Numerical Algorithms*, vol. 26, no. 4, pp. 317–332, 2001.
- [19] D. O’Leary, “The block conjugate gradient algorithm and related methods,” *Linear Algebra and its Applications*, vol. 29, pp. 293–322, 1980.
- [20] R. W. Freund and M. Malhotra, “A block QMR algorithm for non-Hermitian linear systems with multiple right-hand sides,” *Linear Algebra and its Applications*, vol. 254, no. 1-3, pp. 119–157, 1997.
- [21] R. C. O’Reilly and J. M. Beck, “A family of large-stencil discrete Laplacian approximations in three-dimensions,” *International Journal for Numerical Methods in Engineering*, pp. 1–16, 2006.