Research Article

A Flexible Global GCRO-DR Method for Shifted Linear Systems and General Coupled Matrix Equations

Jing Meng,1,2 Xian-Ming Gu,3, Wei-Hua Luo,4 and Liang Fang1

1School of Mathematics and Statistics, Taishan University, Taian 271021, Shandong, China
2Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing 100000, China
3Institute of Mathematics, School of Economic Mathematics, Southwestern University of Finance and Economics, Chengdu, Sichuan 611130, China
4School of Mathematics and Physics Science, Hunan University of Arts and Science, Changde, Hunan 415000, China

Correspondence should be addressed to Xian-Ming Gu; guxianming@live.cn

Received 7 January 2021; Accepted 22 April 2021; Published 12 May 2021

Academic Editor: Jia-Bao Liu

Copyright © 2021 Jing Meng et al. 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 mainly focus on the development and study of a new global GCRO-DR method that allows both the flexible preconditioning and the subspace recycling for sequences of shifted linear systems. The novel method presented here has two main advantages: firstly, it does not require the right-hand sides to be related, and, secondly, it can also be compatible with the general preconditioning. Meanwhile, we apply the new algorithm to solve the general coupled matrix equations. Moreover, by performing an error analysis, we deduce that a much looser tolerance can be applied to save computation by limiting the flexible preconditioned work without sacrificing the closeness of the computed and the true residuals. Finally, numerical experiments demonstrate that the proposed method illustrated can be more competitive than some other global GMRES-type methods.

1. Introduction

In the current study, we focus on the solution of sequences of shifted linear systems of the form

\[
(A_i - \sigma_i I)x_i^\ell = b_i^\ell, \quad \text{for } i = 1, 2, \ldots, s; \ell = 1, 2, \ldots, p_i,
\]

(1)

where \(A_i \in \mathbb{C}^{n \times n}\) are large nonsingular matrices of order \(n\), the right-hand sides \(b_i^\ell \in \mathbb{C}^n\) are vectors of the length \(n\), and \(\sigma_i^\ell \in \mathbb{C}\) are called shifts. Such shifted linear systems (1) arise in many realistic applications, for example, electromagnetic scattering [1], QCD [2–5], dynamics of structures [6], complex network [7–9], and model reduction [10].

Note that problem (1) is reduced to the solution of a sequence of linear systems with the form

\[
\bar{A}_i x_i = b_i, \quad \text{for } i = 1, 2, \ldots, s,
\]

(2)

when \(\ell = 1\), where the coefficient matrix \(\bar{A}_i = A_i - \sigma_i I\) and the right-hand sides \(b_i\) change from one system to the next. One approach to more efficient solution of such systems is to exploit Krylov subspace recycling; that is, recycling a judiciously selected subspace of the search space typically reduces the number of iterations substantially. There are several algorithms that take advantage of recycling for subsequent systems as well as for restarts of generalized minimal residual (GMRES) type methods, such as the generalized conjugate residual method with inner orthogonalization (GCRO) [11], GCRO with deflated restarting (GCRO-DR) [12], GMRES with deflated restarting (GMRES-DR) [13] (only for a single system), and GCRO with optimal truncation (GCROT) [14].

It is also noted that, for each \(i\), we substantially solve a family of shifted linear systems with the form

\[
(A - \sigma I)x^\ell = b^\ell, \quad \text{for } \ell = 1, 2, \ldots, p_i.
\]

(3)
Krylov subspace solvers [15] are particularly attractive for the solutions of (3) because of its shift-invariance property [16]. That is, for any shift \(\sigma\), the \(m\)-th Krylov subspace satisfies
\[
\mathcal{K}_m(A, b) = \mathcal{K}_m(A + \sigma I, \bar{b}),
\]
as long as \(\bar{b} = \beta b\). This property has led to many approaches for solving such linear systems (3) simultaneously by generating only one subspace. Some solvers are built upon the (Bi-)Lanczos recurrences [17–21] and some others use the Arnoldi-like iteration [16, 22–28]. However, the shift-invariance property will be lost when using the general preconditioning, such as Jacobi preconditioning, SSOR preconditioning, ILU preconditioning, AMG preconditioning, and AINV preconditioning. Furthermore, when \([b_1^T]^T\) are unrelated, we cannot employ the shift-invariance property. It means that all residuals are not simultaneously minimized while maintaining collinearity, for example, the GMRES-type methods [16, 23, 29].

Therefore, we propose an alternative without exploiting the shift-invariance. In fact, the solution of (3) is equivalent to solve a Sylvester equation [30],
\[
AX + XT = B,
\]
where \(T = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_p) \in \mathbb{C}^{p \times p}\), and \(B = [b_1^T, b_2^T, \ldots, b_p^T] \in \mathbb{C}^{mp^2}\). Therefore, (1) can also be reformulated as a family of Sylvester equations with the slowly changing coefficient matrices
\[
A_iX_i + X_iT_i = B_i, \quad i = 1, 2, \ldots, s.
\]

For simplicity of the presentation, a linear operator \(\mathcal{A}\) is defined as follows:
\[
\mathcal{A} : X \rightarrow AX + XT.
\]
Applying the linear operator \(\mathcal{A}\), we can rewrite (1) as
\[
\mathcal{A}(X_i) = B_i.
\]

Moreover, we also assume that the linear operator \(\mathcal{A}\) is invertible; that is,
\[
\mathcal{A}^{-1}(B_i) = X_i.
\]

Obviously, the linear systems (8) can be seen as generalized version of (2). Therefore, based on the special structure of (1), we propose a new generalized global version of the GCRO-DR method that not only combines with any type of the preconditioning but also allows one to retain important spectral information generated during the solution of the \(i\)-th linear system and then to exploit such information to accelerate convergence of the iterations when solving the subsequent \(i + 1\)-th system. Moreover, the proposed method can be combined with any form of preconditioning and does not require the right-hand sides to maintain collinearity.

For simplicity, we denote \(\mathcal{A}(X)\) by \(\mathcal{A}X\). In the following, we utilize the operator \(\mathcal{A}\) to present our flexible global GCRO-DR algorithms for solving (1).

The paper is organized as follows. A global version of GCRO-DR to solve (1) is derived in detail in Section 2. We also apply the flexible preconditioning operator at each iteration to improve the global GCRO-DR algorithm. By performing an error analysis, we deduce that a much looser tolerance can be applied to save computation by limiting the flexible preconditioned work without sacrificing the closeness of the computed and the true residuals. Meanwhile, we also propose a global version of the FGMRES-DR method and then analyze the relationship between two flexible global methods. In Section 2.5, the flexible global GCRO-DR method is applied to solve the general coupled matrix equations. Section 3 numerically demonstrates the effectiveness of the novel method. Finally, some conclusions and remarks are summarized in Section 4.

2. Flexible Global GCRO-DR Method

GCRO-DR is a well-established Krylov subspace solver that paired with a harmonic Ritz vector recycling strategy [12]. GCRO [11] is a nested Krylov method, which uses GCR as the outer solver and the restarted GMRES as an inner method. In [12], Parks et al. propose GCRO-DR that exploits deflated restarting within the framework of GCRO for a family of slowly changing linear equations. Because of its superior numerical behavior, we introduce a global version of GCRO-DR to solve (1). Here, we not only employ the subspace recycling at each cycle but also apply the flexible preconditioning operator at each iteration. The proposed method is referred to as flexible global GCRO-DR (FGGCRO-DR).

2.1. Flexible Global Arnoldi Process. We first present a global Arnoldi process used in the flexible setting to create an \(F\)-orthonormal basis for the global Krylov subspace in Algorithm 1. We define \(\mathcal{M}_j\) as the preconditioning operator used at the \(j\)-th iteration, and the action \(\mathcal{M}_j\) on rectangular matrix \(V\) is denoted by \(\mathcal{M}_j(V)\).

The flexible global Arnoldi process, recursively, generates an \(F\)-orthonormal basis \(\mathcal{V}_{m+1} = [V_1, V_2, \ldots, V_{m+1}]\) of the global Krylov subspace spanned by the rectangular matrices \([V_1, \mathcal{A}V_1, \ldots, \mathcal{A}^mV_m]\). At the end of the \(m\)-th iteration, we obtain a matrix relation (flexible global Arnoldi relation); that is,
\[
AX_m = \mathcal{V}_{m+1}(H_m \otimes I_p),
\]
where \(X_m = [\mathcal{M}_1(V_1), \ldots, \mathcal{M}_m(V_m)] \in \mathbb{C}^{m \times p}, \mathcal{V}_{m+1} \in \mathbb{C}^{p \times (m+1)p}\), and
is an upper Hessenberg matrix.

2.2. The FGL-GCRO-DR Algorithm. Suppose that \( \mathcal{Z}_k = [Z_1, \ldots, Z_k] \) and \( \mathcal{V}_k = [V_1, \ldots, V_k] \in \mathbb{C}^{n \times p} \) are given matrices satisfying

\[
\mathcal{A}[\mathcal{Z}_k, Z_{k+1}, \ldots, Z_m] = \mathcal{V}_k (I_k \otimes I_p),
\]

where \( Z_i, i = 1, 2, \ldots, k \) is a search vector, recycled from either the previous linear system or the previous cycle, and \( 0 \leq k \leq m \). By minimizing the residual F-norm,

\[
\min_{\mathbf{X} \in \mathcal{X} \times p} \| R_0 - \mathbf{X} \|_F, \tag{13}
\]

over the search space \( X_0 + \mathcal{R}(\mathcal{Z}_k), X_k \) and \( R_k \in \mathbb{C}^{n \times p} \) satisfy

\[
X_k = X_0 + \mathcal{Z}_k \langle \mathcal{V}_k, R_0 \rangle F \otimes I_p,
\]

\[
R_k = R_0 - \mathcal{V}_k \langle \mathcal{V}_k, R_0 \rangle F \otimes I_p \quad \text{such that} \quad R_k \perp \mathcal{R}(\mathcal{V}_k). \tag{14}
\]

Based upon what is proven in [11, 14] and using the framework discussed in [12, 32], the \( m-k \) steps of the flexible global Arnoldi process are carried out, with the starting rectangular matrix \( V_{k+1} = R_k / \| R_k \| \), while holding the orthogonality to \( \mathcal{V}_k \), to eventually obtain

\[
\mathcal{A}[\mathcal{Z}_k, Z_{k+1}, \ldots, Z_m] = \mathcal{V}_k (I_k \otimes I_p) \langle Z_{k+1}, Z_{k+2}, \ldots, Z_m \rangle \otimes I_p
\]

\[
= [V_{k+1}, V_{k+2}, \ldots, V_{m+1}] (H_{m-k} \otimes I_p). \tag{15}
\]

Since \( R_k \perp \mathcal{R}(\mathcal{V}_k) \), that is, \( V_{k+1} \perp \mathcal{V}_k \), the following flexible global Arnoldi-like relation holds:

\[
\mathcal{A}[\mathcal{Z}_k, Z_{k+1}, \ldots, Z_m] = \mathcal{V}_m (H_m \otimes I_p),
\]

\[
\mathcal{A}[\mathcal{Z}_k, Z_{k+1}, \ldots, Z_m] = \mathcal{V}_m (H_m \otimes I_p), \tag{16}
\]

At the end, we find the approximate solution \( X_m = X_0 + \mathcal{Z}_m (y^* \otimes I_p) \in \mathbb{C}^{n \times p} \) by minimizing the norm \( \| R_0 - A\mathcal{Z}_m (y \otimes I_p) \|_F \), where \( y^* \) satisfies

\[
y^* = \arg \min_{y' \in \mathbb{C}^n} \| R_0 - \mathcal{A}[\mathcal{Z}_m (y' \otimes I_p)] \|_F = \arg \min_{y' \in \mathbb{C}^n} \| \beta_0 - H_m y' \|_2,
\]

with \( \beta_0 = \langle \mathcal{V}_{m+1}, R_0 \rangle_F \).

To improve the convergence of iterative solvers, the smallest magnitude eigenvalues will be deflated. Following the idea of [12, 32], we use harmonic Ritz vectors with those eigenvalues to construct approximate invariant subspace and then the subspace for next cycles is retained. Firstly, we compute \( k \) harmonic Ritz vectors \( \tilde{\mathcal{C}}_k = \mathcal{W}_m (g_k \otimes I_p) \), such that

\[
\mathcal{A}[\mathcal{W}_m \tilde{\mathcal{C}}_i, \mathcal{W}_m (g_i \otimes I_p)] - \theta_i \mathcal{W}_m g_i \perp \mathcal{R}[\mathcal{W}_m], \quad \text{for} \quad 0 \leq i \leq k,
\]

where \( \mathcal{W}_m \in \mathbb{C}^{n \times mp} \), and each harmonic Ritz pair \( (\theta_i, g_i) \) satisfies

\[
H_m H_m^H g_i = 0, H_m H_m^H g_i, \quad 1 \leq i \leq k. \tag{19}
\]

It is noted that the harmonic residual vectors satisfy

\[
\mathcal{A}[\mathcal{W}_m \tilde{\mathcal{C}}_i, \mathcal{W}_m (g_i \otimes I_p)] - \theta_i \mathcal{W}_m g_i \in \mathcal{R}[\mathcal{W}_{m+1}]. \tag{20}
\]
Observe that the residual \( R_m = R_0 - \mathscr{A}_m(y^* \otimes I_p) = \mathscr{V}_{m+1}(y^* \otimes I_p) \in \mathcal{R}(\mathscr{V}_{m+1}) \perp \mathcal{R}(\mathscr{A}_m) \). So, the residual and the harmonic residuals are in the same \( p \)-dimensional

\[
\mathscr{A}_m(G_k \otimes I_p) = [\mathscr{V}_m(G_k \otimes I_p), R_m] \cdot \left[ \begin{array}{c} \operatorname{diag}(	heta_1, \ldots, \theta_k) \\ a_{1:k} \end{array} \right] \otimes I_p),
\]

where \( a(:, i) \) satisfies \( \mathscr{A}_m(g_i \otimes I_p) = 0 \) if \( a(i) R_m \), \( 1 \leq i \leq k \), and \( \theta_k = [g_1, \ldots, g_k] \in \mathbb{C}^{m \times k} \).

Due to the relations (10) and (21) can be also expressed as

\[
\mathscr{V}_m(G_k \otimes I_p) = \mathscr{V}_m(G_k \otimes I_p) = \mathscr{V}_m(G_k \otimes I_p).
\]

Next, we compute the reduced QR-factorization of \( H_m G_k = QR \) with \( Q \in \mathbb{C}^{(m+1) \times k} \) and \( R \in \mathbb{C}^{k \times k} \) and then define \( \mathscr{I}_k, \mathscr{V}_k \in \mathbb{C}^{m \times k} \) as

\[
\mathscr{I}_k = \mathscr{I}_m(G_k R^{-1} \otimes I_p),
\]

\[
\mathscr{V}_k = \mathscr{V}_m(Q \otimes I_p),
\]

such that \( \mathscr{A}_k = \mathscr{V}_k \). Since \( R_m \) is \( F \)-orthogonal to the subspace \( \mathcal{R}(\mathscr{A}_m) \) and \( \mathcal{V}_m \) is \( F \)-orthonormal, we have

\[
\langle \mathscr{V}_k, R_m \rangle_F = \langle \mathscr{V}_m(Q \otimes I_p), R_m \rangle_F = 0,
\]

\[
\langle \mathscr{V}_k, \mathscr{V}_k \rangle_F = \langle \mathscr{V}_m(Q \otimes I_p), \mathscr{V}_m(Q \otimes I_p) \rangle_F = Q^H Q = I_k.
\]

Therefore, at the end of the loop, an update \( \mathscr{I}_k \) is generated, scaled such that \( \mathscr{V}_k = \mathscr{A}_k \) has \( F \)-orthonormal columns; the previous global Krylov subspace basis is dropped, and then we restart. Details of the FGL-GCRO-DR method are summarized in Algorithm 2.

In addition, the major advantage of FGL-GCRO-DR is that it can retain important spectral information generated during the solution of the \( i \)-th linear system and then exploit such information to accelerate convergence of the iterations when solving the subsequent \( i + 1 \)-th system. Then we detail how to recycle subspace when solving sequences of slowly changing linear systems (8) as follows (Algorithm 3).

2.3. Comments on the FGL-GCRO-DR Method. To achieve better efficiency of the linear system solvers, we not only employ the subspace recycling at each cycle but also apply the flexible preconditioning operator at each iteration. As we know, the variable preconditioning can be construed as an inexact subspace method. If the matrix \( \mathscr{A} \) is difficult to invert, we will resort to several steps of iterative method as flexible preconditioning, so we obtain in some way an approximate result,

\[
\mathcal{Z}_j = \mathcal{A}^{-1} V_j,
\]

at \( j \)-th (inner) iteration for the equation \( \mathcal{A} Z = V_j, j = 1, \ldots, m \). In such a situation, what are the properties of the true residual obtained by using the flexible preconditioning? We will address this question in this section. Following [33, 34], we set

\[
P_j = V_j - \mathcal{A} \mathcal{Z}_j, j = 1, \ldots, m,
\]

to be the residual that results after the iterative solver has been terminated; then

\[
\mathcal{Z}_j = \mathcal{A}^{-1} V_j - \mathcal{A}^{-1} P_j, j = 1, \ldots, m.
\]

Therefore, the perturbation in linear operation with \( \mathcal{A} \) can be written explicitly as

\[
\mathcal{A} \mathcal{Z}_j = \mathcal{A} \mathcal{A}^{-1} V_j - \mathcal{A} \mathcal{A}^{-1} P_j
\]

\[
= \left( I + P_j \frac{V_j}{\| V_j \|_F} \right) V_j
\]

\[
= (I - \mathcal{E}_j) V_j.
\]

The perturbation implies that the error computed has the bound

\[
\| \mathcal{E}_j \| \leq \frac{\| P_j \|_F}{\| V_j \|_F}.
\]

In fact, the solution is obtained directly from the approximation subspace \( \mathcal{R}(\mathcal{Z}_m) \); that is,

\[
X_m = \mathcal{Z}_m(\mathcal{Y}^* \otimes I_p),
\]

indicating that the perturbation \( \mathcal{E}_j \) does not affect the computation of the true residual. In other words, in exact arithmetic

\[
R_m = R_0 - \mathcal{A} \mathcal{Z}_m(\mathcal{Y}^* \otimes I_p)
\]

\[
= \mathcal{V}_{m+1}(\beta e_1 \otimes I_p) - \mathcal{V}_{m+1}(H_m \mathcal{Y}^* \otimes I_p)
\]

\[
= \mathcal{V}_{m+1}(\beta e_1 - H_m \mathcal{Y} \otimes I_p),
\]
Input: $A \in \mathbb{C}^{m \times n}$, $B \in \mathbb{C}^{m \times p}$. Choose $m$, $k$, tol and initial guesses $X_0$.

Output: $X_m$ such that $(\|R_m\|_F/\|R_0\|_F) \leq \text{tol}$.

1. $R_0 = B - A X_0$.
2. While $\|R_m\|_F > \|R_0\|_F \times \text{tol}$ do
   3. If $m = 1$ do
      4. $y = \|R_0\|_F$, $V_1 = R_0/y$.
      5. Apply Algorithm 2 to generate $Z_m$, $H_m$, $V_{m+1}$ such that $A Z_m = V_{m+1} (H_m \otimes I_p)$, and set $W_m = V_m$.
   6. Else
      7. Apply $m - k$ steps of the Algorithm 2 with the starting rectangular matrix $V_{k+1} = R_k/\|R_k\|_F$, such that $\beta Z_m = V_{m+1} (H_m \otimes I_p)$, and set $W_m = V_m$.
   8. Endif
9. Set $X_m = X_0 + \sum_{i=1}^{m} \left( V_i \otimes I_p \right)$.
10. Compute the $k$ smallest eigenpairs $\left( \beta_i, g_i \right)$ of $H_m^H H_m g_i = \theta_i H_m^H g_i$, $1 \leq i \leq k$.
11. Choose $\beta_i$ from $\left\{ \beta_i \mid \beta_i \leq \|A X_i - V_i \|_F \right\}$, $1 \leq i \leq k$.
12. $Q_R = H_m G_k$.
13. $V_k = V_{m+1} Q$.
14. $Z_k = Z_m G_k R^{-1}$.
15. $W_k = W_m G_k R^{-1}$.
16. Endwhile

Algorithm 1: The FGL-GCRO-DR method.

\[ y = \|R_0\|_F \text{ and set } V_1 = R_0/y; \]

for $j = 1, \ldots, m$, do
   \[ Z_j = \mathcal{M}_j(V_j); \]
   \[ \overline{V} = A Z_j; \]
   for $i = 1, \ldots, j$, do
      \[ h_{ij} = (V_i, \overline{V})_p, i = 1, \ldots, j; \]
      \[ \overline{V} = \overline{V} - V_i h_{ij}; \]
   endfor
endfor

Algorithm 2: Flexible global Arnoldi process.

1. Suppose that $\mathcal{V}_i/A$ and $w_i/A$ are defined from solving $i$-th linear system linear and satisfy $A Z_i = V_i$ with $\langle \mathcal{V}_i, V_i \rangle = I_k$.
2. $R_{i+1}^k = B_{i+1} - A_{i+1} X_{i+1}$.
3. $Q_R = A_{i+1} Z_{i+1}$.
4. $\mathcal{V}_{i+1} = Q$
5. $\mathcal{Z}_{i+1} = \mathcal{Z}_i (R^{-1} \otimes I_p)$
6. $W_{i+1} = W_i (R^{-1} \otimes I_p)$
7. $X_{i+1} = X_i + \mathcal{V}_{i+1} \langle \mathcal{V}_{i+1}^*, (\mathcal{Z}_{i+1}(R_{i+1}^k) \otimes I_p) \rangle$
8. $R_{i+1}^k = R_i^k - \mathcal{V}_{i+1} \langle \mathcal{V}_{i+1}^*, (\mathcal{Z}_{i+1}(R_{i+1}^k) \otimes I_p) \rangle$
9. Apply $m-k$ steps of Algorithm 2 with the starting rectangular matrix $V_{i+1} = R_k/\|R_k\|_F$, such that $A_{i+1} Z_{i+1} = V_{i+1} (H_{i+1} \otimes I_p)$, and set $W_{i+1} = V_{i+1}
10. y^* = \arg \min_{y \in \mathcal{C}} \| - H_w y \|_2$ with $\beta = \langle \mathcal{V}_{i+1}, R_{i+1}^k \rangle_F$
11. $X_{i+1} = X_{i+1} + \mathcal{V}_{i+1} \langle \mathcal{V}_{i+1}^*, \beta \rangle$
12. $R_{i+1}^k = B_{i+1} - A_{i+1} X_{i+1}$
13. $W_m = [W_k^k, W_m^m (1: n, (k+1)p: mp)]$

Algorithm 3: FGL-GCRO-DR for solving sequences of slowly changing linear systems.
and is orthogonal to \( \mathcal{R}(\mathcal{F}_{m_{1}}) \). As a consequence, the perturbation of the linear operations only influences the goodness of \( \mathcal{R}(\mathcal{F}_{m}) \) and the accuracy of the approximate solution. Therefore, we deduce that, in the flexible setting, a dynamic stopping criterion or a much looser tolerance can be employed to save computation by limiting the flexible preconditioning work without sacrificing the closeness of the computed and the true residuals.

2.4. The Relationship between Two Flexible Global Methods with Subspace Recycling. As we know, FGCRO-DR [32] and FGMRES-DR [35] are two important methods in the general category of augmented (deflated) GMRES-type methods, which choose harmonic Ritz vectors as augmented (deflated) subspace. They emphasize on the solution of linear system \( Ax = b \) which has only one equation. The relationship between FGCRO-DR and FGMRES-DR has been well described by [32]. It is shown that they are algebraically equivalent as long as a collinearity condition is satisfied. In fact, FGL-GCRO-DR is a general globalization of FGCRO-DR. For the sake of descriptive integrality, we also present a global generalization of FGMRES-DR algorithm, referred to as FGL-GMRES-DR, in Algorithm 4. As for global versions, the FGL-GCRO-DR method is also closely related to the FGL-GMRES-DR approach in algebraical analysis, but their numerical behavior will be slightly different. Details will be shown in Section 3.

Furthermore, we compare the main calculation cost and storage capacity for each cycle of FGL-GCRO-DR and FGL-GMRES-DR in Table 1. In practical applications, \( n \gg m, k \). Here, \( opM \) and \( opA \) denote floating-point calculation counts for the preconditioning application and the linear operations, respectively.

The computational cost of harmonic Ritz vectors and \( \min[y - H_{n}y]_{i}^{2} \) have not been induced in Table 1 since the cost is approximately \( O(p^{3}) \), much more less than \( c = (opM + opA)(m - k) + (2(m + 1)kp + p + 2mkp + (m - k)(2m + 2k + 6))np \) is referred to as the total cost of FGL-GCRO-DR. One sees that the time complexities of two global algorithms are the same, that is, \( O((m - k)n^{2}p) \), but FGL-GMRES-DR requires slightly more computation per cycle than FGL-GCRO-DR in terms of computational cost.

2.5. Application to the General Coupled Matrix Equations. FGL-GCRO-DR substantially belongs to the family of the global Krylov subspace method, which can be also applied to solve the general coupled matrix equations of the form

\[
\sum_{j=1}^{l} \left( A_{i1,j} X_{1} B_{1,j} + A_{i2,j} X_{2} B_{2,j} + \cdots + A_{ij,j} X_{j} B_{ij,j} \right) = D_{i}, \quad i = 1, 2, \ldots, l,
\]

including the (coupled) Sylvester and the (coupled) Lyapunov matrix equations. The literature on solving (32) is particularly rich and still growing [36–53], due to the prominent role of such matrix equations in stability theory [54], perturbation analysis [55], control theory [56, 57], and so forth. In particular, the global subspace solvers perform better than other approaches for solving (32), especially for the coefficient matrices which are large and sparse; for more details, see [44–53, 58]. Therefore, we are interested in exploiting the FGL-GCRO-DR method for solving (32). To do so, the linear operator can be defined as follows:

\[
\mathcal{F}: \mathbb{R}^{mn \times \cdots \times mn} \rightarrow \mathbb{R}^{np \times np},
\]

\[
X = (X_{1}, X_{2}, \ldots, X_{l}) \rightarrow \mathcal{F}(X) = (\mathcal{A}_{1}(X), \mathcal{A}_{2}(X), \ldots, \mathcal{A}_{l}(X)),
\]

where \( \mathcal{A}_{i} = \sum_{k=1}^{l} \sum_{j=1}^{l} A_{ik,j} X_{k} B_{kj} \), \( i = 1, 2, \ldots, l \). Exploiting the operator \( \mathcal{F} \), we can rewrite (32) as

\[
\mathcal{F}(X) = D,
\]

where \( D = (D_{1}, D_{2}, \ldots, D_{l}) \). The effectiveness of FGL-GCRO-DR for the solution of (32) is shown in Section 3.

3. Numerical Experiments

This section presents some numerical examples to illustrate the effectiveness of our algorithm for solving shifted linear system (1) and matrix equations (8) and compare the performance with the global GMRES \( (m) \) approach (referred to as GL-GMRES \( (m) \)) [31, 50, 59] and GL-GMRES-DR [60], with or without preconditioning. In each example, the flexible preconditioning is dominated by 5 steps of the global full GMRES [50].

We consider three numerical illustrations here. The first two experiments are sequences of linear systems arising from lattice QCD [2–4, 61] and the finite difference scheme of the convection-diffusion-reaction equation [62], respectively. The last one is the general coupled Sylvester matrix equations. The comparisons are mainly made in two aspects: the execution time in seconds \( (T \, s) \), for short) and number of matrix-vector products (mvp, for short).

All the numerical experiments are carried out on MATLAB 2017b. We choose \( X_{0} = 0_{np}, m = 20, \) and \( k = 10 \). The stopping criteria are \( \|B_{i} - \mathcal{A}_{i}X_{n}\|_{\mathcal{F}}/\|B_{i}\|_{\mathcal{F}} \leq 10^{-8}, \) \( i = 1, 2, \ldots, s \) for all the approaches, or when mvp exceed the set maximum of matrix-vector multiplications (MAXIT, for short). Moreover, the symbol \( \dagger \) represents that the solver does not achieve the stopping criteria within MAXIT and RHSs = rand \( (n, p) \).

Example 1. We first consider two groups of complex Lattice QCD matrices from Matrix Market (please refer to http://math.nist.gov/Matrix-Market for details). One group has seven 3172 \( \times \) 3172 matrices and the other has seven 49152 \( \times \) 49152 matrices. For each matrix \( D_{i} \), a critical parameter \( k_{i} \) is provided such that \( I - kD \) with \( 0 \leq k \leq k_{i} \) is positive-real. For
Input: $\mathcal{A} \in \mathbb{C}^{mn}$, $B \in \mathbb{C}^{np}$. Choose $m$, $k$, $\text{tol}$ and initial guesses $X_0$.

Output: $X_m$ such that $(\|R_m\|_2/\|R_0\|_2) \leq \text{tol}$

1. $R_0 = B - \mathcal{A}X_0$, $y = \|R_0\|_2$, $V_1 = R_0 / y$
2. Apply Algorithm 2 to generate $\tilde{F}_{m+1}, \tilde{V}_{m+1}, H_m$ such that $\mathcal{A} \tilde{F}_m = \tilde{V}_{m+1}(H_m \otimes I_p)$
3. Solve $\min_{y} ||y - H_m y ||_2$ for $y^*$. Set $X_0 = X_0 + \tilde{F}_m\tilde{V}_m(y^* \otimes I_m)$, $R_0 = B - \mathcal{A}X_0$
4. while $||R_m||_2 > ||R_0||_2 \times \text{tol}$ do
5. Compute the $k$ smallest eigenpairs $(\theta_i, g_i)$ of $(H_m + h_{m+1,m}h_{m+1,m}H_m \otimes \mathcal{A}^*)_m$
6. $QR = \left[ \begin{array}{c} P_k \\ 0_{1 \times np} \end{array} \right] y - H_m y^*$
7. $\tilde{V}_k = \tilde{V}_{m+1} Q$
8. $\tilde{F}_k = \tilde{F}_{m+1} Q$
9. $H_k = Q^H H_k Q (1 : m, 1 : k)$
10. Apply $m - k$ steps of Algorithm 2 with the starting rectangular matrix $V_{k+1} = R_k / ||R_k||_2$, such that $\mathcal{A} \tilde{F}_{m-k} = \tilde{V}_{m+1}(H_{m-k} \otimes I_p)$
11. $y^* = \arg \min \{y ||y - H_k y ||_1 \}$ such that $y = \langle \tilde{V}_{m+1}, R_0 \rangle$
12. $X_m = X_{m-1} + \tilde{F}_m(y^* \otimes I_m)$
13. $R_m = B - \mathcal{A}X_m$
14. endwhile

Algorithm 4: FGL-GMRES-DR.

Table 1: The main calculation cost and storage capacity of FGL-GCRO-DR and FGL-GMRES-DR.

<table>
<thead>
<tr>
<th>Computation of</th>
<th>FGL-GMRES-DR $(m, k)$</th>
<th>FGL-GCRO-DR $(m, k)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tilde{V}_{k+1}(:, 1 : (k + 1)p)$</td>
<td>$2(m + 1)(k + 1)p$</td>
<td>$2n(m + 1)kp + np$</td>
</tr>
<tr>
<td>$\tilde{F}_k(:, 1 : kp)$</td>
<td>$2mnp$</td>
<td>$2mnkp$</td>
</tr>
<tr>
<td>$\tilde{V}_m(:, (k + 2)p + 1 : (m + 1)p)$</td>
<td>$\text{op}(m - k) + (m - k)(2n + 2k + 5)np$</td>
<td>$\text{op}(m - k) + (m - k)(2n + 2k + 6)np$</td>
</tr>
<tr>
<td>$\tilde{F}_m(:, (k + 1)p + 1 : mp)$</td>
<td>$\text{op}(m - k)$</td>
<td>$\text{op}(m - k)$</td>
</tr>
<tr>
<td>Total cost</td>
<td>$C + (m + k - 1)np$</td>
<td>$C$</td>
</tr>
<tr>
<td>Memory requirement</td>
<td>$(2n + 3)np$</td>
<td>$(3m + 3)np$</td>
</tr>
</tbody>
</table>

Each set, we take $A_k = (k_c + 10^{-3})I - D_1$ as the sequence matrix, and $T_k = [0.01, 0.02, 0.03, 1, 2, 3, 4]$. The numerical computation is carried out with MAXIT = 1000. The corresponding numerical results are reported in Tables 2 and 3. In addition, we also depict the residual histories for linear systems solved using iterative algorithms with or without flexible preconditioning.

As seen from Tables 2 and 3 and Figures 1 and 2, it is found that using deflation and flexible preconditioning is conducive to improving the convergence rate. FGL-GCRO-DR has a significant decrease in terms of mvs and $T(s)$, which are similar to FGL-GMRES-DR, but performs slightly better. The GL-MRES method is less effective and does not converge in some test problems. The above results show that FGL-GCRO-DR has the potential to improve the convergence, compared with other global Krylov subspace solvers.

Example 2. For the second case, we assess the performance of global solvers on three-dimensional separable model convection-diffusion problem in the unit cube with the homogeneous Dirichlet boundary conditions [62]:

$$-\varepsilon \Delta u + \nu \nabla u - ru = f, \quad \text{in} \Omega, \quad (35)$$

where $\varepsilon = 1$ and $r = [0, 100, 400, 600, 800, 1000]$. For obtaining the sequence matrices, we set $v = [0.1 \ast k, 0.5 \ast i \ast k, k]$ with $k = 1, 2, \ldots, 5$ and $i = \sqrt{-1}$ and then use the central finite difference scheme to discrete the convection-diffusion equation with the mesh size $h = 0.025$ or $h = 0.02$. Thus, two groups of complex matrices are obtained. One group has five 59319 $\times$ 59319 matrices and the other has five 117649 $\times$ 117649 matrices. As the definition of MAXIT, we choose MAXIT = 500.

The numerical results reported in Tables 4 and 5 and Figures 3 and 4 show that FGL-GMRES is slightly superior to FGL-GCRO-DR when the mesh size $h = 0.025$. However, as the number of grid points increases, FGL-GCRO-DR enjoys the significant decrease in the number of mvs and $T(s)$. Meanwhile, according to Tables 4 and 5, we can see that the convergence is speeded up remarkably by considering the flexible preconditioning. Moreover, even the size of grid points increases, and the average number of mvs for FGL-GCRO-DR does not increase.

Example 3. For the last test case, we are interested in solving the general coupled Sylvester matrix equations [45, 47]:

$$\begin{align*}
AX - YB &= E, \\
CX - YD &= F,
\end{align*} \quad (36)$$

where
Table 2: Performance comparisons for 3072 × 3072 QCD matrices.

<table>
<thead>
<tr>
<th>Problems</th>
<th>GL-GCRO-DR</th>
<th>FGL-GCRO-DR</th>
<th>GL-GMRES (m)</th>
<th>FGL-GMRES (m)</th>
<th>GL-GMRES-DR</th>
<th>FGL-GMRES-DR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matrix 1</td>
<td>mvs 780</td>
<td>135</td>
<td>940</td>
<td>180</td>
<td>†</td>
<td>150</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 6.431</td>
<td>2.166</td>
<td>4.166</td>
<td>3.426</td>
<td>†</td>
<td>2.993</td>
</tr>
<tr>
<td>Matrix 2</td>
<td>mvs 431</td>
<td>83</td>
<td>480</td>
<td>100</td>
<td>490</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 3.328</td>
<td>1.618</td>
<td>2.129</td>
<td>1.754</td>
<td>3.464</td>
<td>1.741</td>
</tr>
<tr>
<td>Matrix 3</td>
<td>mvs 501</td>
<td>99</td>
<td>800</td>
<td>140</td>
<td>710</td>
<td>110</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 4.462</td>
<td>2.108</td>
<td>3.733</td>
<td>2.523</td>
<td>4.060</td>
<td>2.329</td>
</tr>
<tr>
<td>Matrix 4</td>
<td>mvs 541</td>
<td>100</td>
<td>620</td>
<td>140</td>
<td>530</td>
<td>110</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 4.090</td>
<td>2.115</td>
<td>2.722</td>
<td>2.551</td>
<td>3.525</td>
<td>2.127</td>
</tr>
<tr>
<td>Matrix 5</td>
<td>mvs 265</td>
<td>46</td>
<td>340</td>
<td>60</td>
<td>270</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 1.573</td>
<td>1.005</td>
<td>1.560</td>
<td>1.040</td>
<td>1.618</td>
<td>1.223</td>
</tr>
<tr>
<td>Matrix 6</td>
<td>mvs 803</td>
<td>148</td>
<td>†</td>
<td>240</td>
<td>820</td>
<td>170</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 5.033</td>
<td>3.463</td>
<td>†</td>
<td>4.414</td>
<td>5.468</td>
<td>3.521</td>
</tr>
<tr>
<td>Matrix 7</td>
<td>mvs 258</td>
<td>50</td>
<td>300</td>
<td>60</td>
<td>260</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 2.024</td>
<td>1.004</td>
<td>1.335</td>
<td>1.064</td>
<td>1.706</td>
<td>1.138</td>
</tr>
</tbody>
</table>

Table 3: Performance comparisons for 49152 × 49152 QCD matrices.

<table>
<thead>
<tr>
<th>Problems</th>
<th>GL-GCRO-DR</th>
<th>FGL-GCRO-DR</th>
<th>GL-GMRES (m)</th>
<th>FGL-GMRES (m)</th>
<th>GL-GMRES-DR</th>
<th>FGL-GMRES-DR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matrix 1</td>
<td>mvs 856</td>
<td>162</td>
<td>†</td>
<td>220</td>
<td>†</td>
<td>180</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 1.306e + 02</td>
<td>8.009e + 01</td>
<td>†</td>
<td>9.015e + 01</td>
<td>†</td>
<td>8.440e + 01</td>
</tr>
<tr>
<td>Matrix 2</td>
<td>mvs 685</td>
<td>139</td>
<td>860</td>
<td>180</td>
<td>720</td>
<td>150</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 8.797e + 01</td>
<td>6.363e + 01</td>
<td>8.556e + 01</td>
<td>7.391e + 01</td>
<td>9.145e + 01</td>
<td>6.868e + 01</td>
</tr>
<tr>
<td>Matrix 3</td>
<td>mvs 706</td>
<td>131</td>
<td>760</td>
<td>180</td>
<td>650</td>
<td>140</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 9.972e + 01</td>
<td>6.186e + 01</td>
<td>7.722e + 01</td>
<td>7.368e + 01</td>
<td>9.033e + 01</td>
<td>6.373e + 01</td>
</tr>
<tr>
<td>Matrix 4</td>
<td>mvs 695</td>
<td>150</td>
<td>740</td>
<td>180</td>
<td>720</td>
<td>160</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 7.931e + 01</td>
<td>7.016e + 01</td>
<td>7.570e + 01</td>
<td>7.427e + 01</td>
<td>8.130e + 01</td>
<td>7.271e + 01</td>
</tr>
<tr>
<td>Matrix 5</td>
<td>mvs 798</td>
<td>110</td>
<td>†</td>
<td>140</td>
<td>800</td>
<td>120</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 9.341e + 01</td>
<td>5.027e + 01</td>
<td>†</td>
<td>5.734e + 01</td>
<td>9.255e + 01</td>
<td>6.274e + 01</td>
</tr>
<tr>
<td>Matrix 6</td>
<td>mvs 838</td>
<td>115</td>
<td>†</td>
<td>180</td>
<td>870</td>
<td>140</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 8.668e + 01</td>
<td>5.480e + 01</td>
<td>†</td>
<td>7.211e + 01</td>
<td>9.357e + 01</td>
<td>6.274e + 01</td>
</tr>
<tr>
<td>Matrix 7</td>
<td>mvs 976</td>
<td>115</td>
<td>†</td>
<td>160</td>
<td>980</td>
<td>130</td>
</tr>
<tr>
<td></td>
<td>(T (s)) 1.104e + 02</td>
<td>5.682e + 01</td>
<td>†</td>
<td>6.585e + 01</td>
<td>1.203e + 02</td>
<td>5.845e + 01</td>
</tr>
</tbody>
</table>

A = diag (diag (rand \((n)\)) + 2) + triu (rand \((n, n)\), 1),
B = diag (diag (rand \((n)\)) + 1.75) + triu (rand \((n, n)\), 1),
C = −diag (diag (rand \((n)\)) + 2) + triu (rand \((n, n)\), 1),
D = diag (diag (rand \((n)\)) + 1.5) + triu (rand \((n, n)\), 1),
E = rand \((n, n)\),
F = rand \((n, n)\).

(37)

Here, we consider \(n = 50, 60, 70, 80, 90, 100\) different orders. As the definition of MAXIT, we choose MAXIT = 500. The corresponding numerical results of mvs and \(T(s)\) are reported in Table 6 and the residual histories are depicted in Figures 5 and 6.

As seen from Table 5 and Figures 5 and 6, it is noted that the flexible deflated GMRES-type methods (FGL-GCRO-DR and FGL-GMRES-DR) are found to be efficient in terms of performance test problems of different orders, while GLGMRES \((m)\) and FGL-GMRES \((m)\) fail to converge in the given steps with or without variable preconditioning, while the parameter \(n\) is greater than 80. The convergence is improved remarkably by considering the eigenvalue deflation technique and the flexible preconditioning. One should note that FGLGCRO-DR enjoys the significant decrease in the number of mvs and \(T(s)\) and performs slightly better than FGL-GMRES-DR method. In addition, when \(n\) increases, the average number of mvs and \(T(s)\) for FGL-GCRO-DR do not increase.
Figure 1: The “−” line and the “+−” line denote the convergence curves for the global method and the flexible global method for solving the linear equations with \( n = 3072 \) in sequence, respectively.
Figure 2: The “-” line and the “+” line denote the convergence curves for the global method and the flexible global method for solving the $s$ linear equations with $n = 49152$ in sequence, respectively.

Table 4: Performance comparisons for 59319 convection-diffusion problem.

<table>
<thead>
<tr>
<th>Problems</th>
<th>Method</th>
<th>Matrix 1</th>
<th>Matrix 2</th>
<th>Matrix 3</th>
<th>Matrix 4</th>
<th>Matrix 5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>GL-GCRO-DR</td>
<td>FGL-GCRO-DR</td>
<td>GL-GMRES (m)</td>
<td>FGL-GMRES (m)</td>
<td>GL-GMRES-DR</td>
</tr>
<tr>
<td>Matrix 1</td>
<td>mvs</td>
<td>2.781e+01</td>
<td>1.735e+01</td>
<td>3.693e+01</td>
<td>1.693e+01</td>
<td>2.811e+01</td>
</tr>
<tr>
<td>T (s)</td>
<td></td>
<td>165</td>
<td>29</td>
<td>340</td>
<td>40</td>
<td>170</td>
</tr>
<tr>
<td>Matrix 2</td>
<td>mvs</td>
<td>2.655e+01</td>
<td>1.766e+01</td>
<td>3.664e+01</td>
<td>1.686e+01</td>
<td>3.115e+01</td>
</tr>
<tr>
<td>T (s)</td>
<td></td>
<td>172</td>
<td>32</td>
<td>340</td>
<td>40</td>
<td>190</td>
</tr>
<tr>
<td>Matrix 3</td>
<td>mvs</td>
<td>2.907e+01</td>
<td>1.713e+01</td>
<td>3.718e+01</td>
<td>1.688e+01</td>
<td>2.946e+01</td>
</tr>
<tr>
<td>T (s)</td>
<td></td>
<td>178</td>
<td>31</td>
<td>340</td>
<td>40</td>
<td>180</td>
</tr>
<tr>
<td>Matrix 4</td>
<td>mvs</td>
<td>3.095e+01</td>
<td>1.818e+01</td>
<td>3.646e+01</td>
<td>1.680e+01</td>
<td>3.120e+01</td>
</tr>
<tr>
<td>T (s)</td>
<td></td>
<td>182</td>
<td>34</td>
<td>340</td>
<td>40</td>
<td>190</td>
</tr>
<tr>
<td>Matrix 5</td>
<td>mvs</td>
<td>2.747e+01</td>
<td>1.747e+01</td>
<td>3.658e+01</td>
<td>1.675e+01</td>
<td>3.271e+01</td>
</tr>
<tr>
<td>T (s)</td>
<td></td>
<td>179</td>
<td>32</td>
<td>340</td>
<td>40</td>
<td>200</td>
</tr>
</tbody>
</table>
Table 5: Performance comparisons for 117649 convection-diffusion problem.

<table>
<thead>
<tr>
<th>Problems</th>
<th>GL-GCRO-DR</th>
<th>FGL-GCRO-DR</th>
<th>GL-GMRES (m)</th>
<th>FGL-GMRES (m)</th>
<th>GL-GMRES-DR</th>
<th>FGL-GMRES-DR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matrix 1</td>
<td>m</td>
<td>209</td>
<td>35</td>
<td>480</td>
<td>60</td>
<td>250</td>
</tr>
<tr>
<td>T (s)</td>
<td>7.486e + 01</td>
<td></td>
<td>4.630e + 01</td>
<td></td>
<td>4.998e + 01</td>
<td></td>
</tr>
<tr>
<td>Matrix 2</td>
<td>m</td>
<td>218</td>
<td>36</td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>T (s)</td>
<td>7.660e + 01</td>
<td></td>
<td>4.006e + 01</td>
<td></td>
<td>4.988e + 01</td>
<td></td>
</tr>
<tr>
<td>Matrix 3</td>
<td>m</td>
<td>219</td>
<td>42</td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>T (s)</td>
<td>7.665e + 01</td>
<td></td>
<td>4.490e + 01</td>
<td></td>
<td>5.013e + 01</td>
<td></td>
</tr>
<tr>
<td>Matrix 4</td>
<td>m</td>
<td>242</td>
<td>41</td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>T (s)</td>
<td>7.664e + 01</td>
<td></td>
<td>4.396e + 01</td>
<td></td>
<td>4.977e + 01</td>
<td></td>
</tr>
<tr>
<td>Matrix 5</td>
<td>m</td>
<td>264</td>
<td>44</td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>T (s)</td>
<td>8.102e + 01</td>
<td></td>
<td>4.351e + 01</td>
<td></td>
<td>9.859e + 01</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3: The "-" line and the "+--" line denote the convergence curves for the global method and the flexible global method for solving the s linear equations with \( n = 59319 \) in sequence, respectively.
Figure 4: The “−” line and the “+−” line denote the convergence curves for the global method and the flexible global method for solving the linear equations with $n = 117649$ in sequence, respectively.

Table 6: Performance comparisons for the general coupled Sylvester matrix equations.

<table>
<thead>
<tr>
<th>Problems</th>
<th>GL-GCRO-DR</th>
<th>FGL-GCRO-DR</th>
<th>GL-GMRES ($m$)</th>
<th>FGL-GMRES ($m$)</th>
<th>GL-GMRES-DR</th>
<th>FGL-GMRES-DR</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 50$</td>
<td>mips</td>
<td>134</td>
<td>37</td>
<td>239</td>
<td>37</td>
<td>130</td>
</tr>
<tr>
<td>$T(s)$</td>
<td>4.060e-01</td>
<td>1.336e-01</td>
<td>2.988e-01</td>
<td>1.396e-01</td>
<td>9.266e-01</td>
<td>3.421e-01</td>
</tr>
<tr>
<td>$n = 60$</td>
<td>mips</td>
<td>217</td>
<td>41</td>
<td>†</td>
<td>59</td>
<td>173</td>
</tr>
<tr>
<td>$T(s)$</td>
<td>6.551e-01</td>
<td>2.165e-01</td>
<td>†</td>
<td>2.195e-01</td>
<td>1.668e+00</td>
<td>7.521e-01</td>
</tr>
<tr>
<td>$n = 70$</td>
<td>mips</td>
<td>243</td>
<td>53</td>
<td>†</td>
<td>115</td>
<td>232</td>
</tr>
<tr>
<td>$T(s)$</td>
<td>1.042e+00</td>
<td>3.929e-01</td>
<td>†</td>
<td>6.168e-01</td>
<td>3.112e+00</td>
<td>1.354e+00</td>
</tr>
<tr>
<td>$n = 80$</td>
<td>mips</td>
<td>507</td>
<td>100</td>
<td>†</td>
<td>507</td>
<td>334</td>
</tr>
<tr>
<td>$T(s)$</td>
<td>2.661e+00</td>
<td>8.868e-01</td>
<td>†</td>
<td>2.661e+00</td>
<td>6.143e+00</td>
<td>3.894e+00</td>
</tr>
<tr>
<td>$n = 90$</td>
<td>mips</td>
<td>507</td>
<td>100</td>
<td>†</td>
<td>†</td>
<td>503</td>
</tr>
<tr>
<td>$T(s)$</td>
<td>3.294e+00</td>
<td>1.109e+00</td>
<td>†</td>
<td>†</td>
<td>1.206e+01</td>
<td>6.228e+00</td>
</tr>
<tr>
<td>$n = 100$</td>
<td>mips</td>
<td>†</td>
<td>166</td>
<td>†</td>
<td>†</td>
<td>†</td>
</tr>
<tr>
<td>$T(s)$</td>
<td>†</td>
<td>2.326e+00</td>
<td>†</td>
<td>†</td>
<td>†</td>
<td>9.962e+00</td>
</tr>
</tbody>
</table>
Figure 5: The convergence curves for the global methods and the flexible global methods for solving general coupled Sylvester matrix equations.
Figure 6: The convergence curves for the global methods and the flexible global methods for solving general coupled Sylvester matrix equations.
4. Conclusions

In this study, a new flexible global GCRO-DR method is proposed for addressing sequences of shifted linear systems and general coupled matrix equations. The proposed method does not require the right-hand sides to be related and can also be compatible with general preconditioning. By performing an error analysis, we deduce that a much looser tolerance can be applied to save computation by limiting the flexible preconditioned work without sacrificing the closeness of the computed and the true residuals. Meanwhile, we also propose a global version of the FGMRES-DR method and then analyze the relationship between two flexible global methods. Finally, numerical experiments show that FGL-GCRO-DR significantly reduces the mvps, saves the runtime, and enjoys faster convergence than some other global GMRES-type approaches on tough problems.

As future work, we plan to investigate the numerical properties of FGL-GCRO-DR on large-scale realistic problems. Of interest are applications related to, for example, model adaptation in Gaussian process models [63] or stochastic finite element methods [64] in three dimensions where variable preconditioning using inexact solvers has to be usually considered. Another extension of this work is a distributed-memory parallel implementation and testing using MPI+OpenMP and MPI+CUDA. Investigation will focus on implementing, testing, and comparing the runtime of the introduced FGL-GCRO-DR on CPUs and GPUs with respect to other existent similar methods.

Data Availability

The data will be made available upon reasonable request via e-mail to the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors’ Contributions

All authors collaborated in carrying out the study and all of them read and approved the final manuscript.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 11601365 and 11801463), Project of Shandong Science and Technology Department Foundation (no. J18L06), Foundation of National Key Laboratory of Computational Physics (no. 6142A05180203), Project of Natural Science Foundation of Shandong Province (no. ZR2016AM06), and the Scientific Research Fund of Hunan Provincial Education Department (no. 19B381).

References


