Recycling Krylov Subspaces and Truncating Deflation Subspaces for Solving Sequence of Linear Systems

Hussam Al Daas, Laura Grigori, Pascal Hénon, Philippe Ricoux

Recyclage de Sous-Espaces de Krylov et Troncature de Sous-Espaces de Déflation pour Résoudre Séquence de Systèmes Linéaires

Hussam Al Daas, Laura Grigori, Pascal Hénon, Philippe Ricoux

Recyclage de Sous-Espaces de Krylov et Troncature de Sous-Espaces de Déflation pour Résoudre Séquence de Systèmes Linéaires

Hussam Al Daas [*], Laura Grigori[*], Pascal Hénon [†], Philippe Ricoux [‡]

Équipe-Projet ALPINES

Résumé :   Ce papier présente plusieures stratégies de déflation liées aux méthodes de recyclage de sous-espaces de Krylov pour résoudre une séquence de systèmes linéaires. À côté de stratégies de déflation très connues qui sont basées sur la déflation des vecteurs de Ritz et Ritz harmonique, on introduit une technique de déflation basée sur la décomposition en valeurs singulières. On considère deux contextes du recyclage, le recyclage de l'espace de Krylov entre les cycles de resart et le recylcage de l'espaces de déflation quand la matrice change dans la séquence. L'efficacité de la méthode proposée est étudiée sur des séquence de systèmes linéaires issues de la modélisation de réservoirs.

Mots-clés :   espaces de Krylov, solveur linéaires itératifs, recyclage, déflation, réduction, svd

[*] ALPINE, INRIA-Paris, Laboratoire J.L. Lions, Sorbonne University, Paris, France
[†] TOTAL EP, Pau, France
[‡] MIT, Department of Mathematics, USA

Recycling Krylov Subspaces and Truncating Deflation Subspaces for Solving Sequence of Linear Systems

Abstract: This paper presents deflation strategies related to recycling Krylov subspace methods for solving one or a sequence of linear systems of equations. Besides well-known strategies of deflation, Ritz- and harmonic Ritz-based deflation, we introduce an SVD-based deflation technique. We consider the recycling in two contexts, recycling the Krylov subspace between the cycles of restarts and recycling a deflation subspace when the matrix changes in a sequence of linear systems. Numerical experiments on real-life reservoir simulations demonstrate the impact of our proposed strategy.

Table des matières

1   Introduction

We consider the system

$$Ax = b \tag{1}$$

where $A \in \mathbb{C}^{n \times n}$ is non-singular, $b \in \mathbb{C}^n$ is a right-hand side vector and $x \in \mathbb{C}^n$ is the vector of unknowns. We are interested in solving (1) by using a Krylov iterative method. GMRES [33], CG [17], GCR [8], and BiCG [26] are widely used Krylov subspace methods. Several works have studied the impact of the spectrum of the matrix A on the convergence rate of Krylov methods [26, 31, 24, 23, 40, 34]. It was noticed that the small and nearly isolated eigenvalues induce a slow rate of convergence. In order to have robustness and fast convergence, a deflation subspace correction can annihilate the impact of this part of the spectrum. Different types of deflation techniques depending on the Hermitian structure of the matrix A are described in the literature [25, 11, 10, 40, 32, 24, 9, 23, 12]. Some of these strategies construct a deflation subspace during the iterative solve of (1).

In several applications, we have to solve successive linear systems $A_i x = b_i$ where $A_i \approx A_{i+1}$, e.g., large-scale eigenvalue problems and Newton's method for solving nonlinear systems. Furthermore, Newton's method combined with a Constrained Pressure Residual (CPR [41, 35]) solver have both cases, $A_i \approx A_{i+1}$ and $A_i = A_{i+1}$. Thus, the deflation subspace that is built during the solution of $A_i x = b_i$ can be useful to solve $A_{i+1} x = b_{i+1}$. However, the deflation subspaces introduced in the previously mentioned work may not be directly used to solve successive linear systems with different right-hand sides even if the matrix does not change.

In [30], the authors introduced a recycling strategy based on Rayleigh-Ritz preconditioner to solve sequence of linear systems of equations with symmetric positive definite matrices. In the same time, De Sturler presented in [6] a truncation strategy based on orthogonality constraints to select vectors from the search subspace and used them to accelerate convergence after restarting the method or when solving a new linear system in a sequence. Later, Parks et al. introduced the method GCRO-DR in [28]. This method is a variant of the restarted GMRES in which an abstract subspace can be used as a deflation subspace. Several papers followed based on this work [3, 16, 12, 39, 1, 2, 19].

Usually, deflation subspaces in recycling methods are approximations of eigenspaces. In [9], the authors proposed an approximation of the eigenspace by using the Ritz vectors. In order to preserve the structure of the Arnoldi procedure, Morgan presented the deflation of harmonic Ritz vectors [24]. A recent survey on recycling subspaces covers most developments in this area [38]. Few papers discuss the impact of singular vectors on the convergence of Krylov methods [36, 22, 21]. In [36], Simoncini shows the impact of very small singular values on the convergence of several restarted variants of Krylov methods when the right-hand side has a large coefficient in the direction of the left singular vectors associated with the smallest singular values. Simoncini's work inspired us to consider deflation of approximate singular vectors rather than eigenvectors.

In the Hermetian case, Vuik et al. [40] prove that for any deflation subspace the effective condition number of the deflated matrix is smaller than the effective condition number of the matrix without deflation. However, this is not true for the non-symmetric case. Moreover, we will see in the numerical experiments 8 how the deflation subspace, in several cases might deteriorate the convergence of the method. Thus, when considering a new matrix in the sequence of linear systems, it is important to check whether the deflation subspace is appropriate or not. For this aim, we introduce a new strategy to truncate the deflation subspace based on the approximation type of the deflation. This truncation strategy allows to select the useful part of the deflation subspace and removes the rest.

We introduce the deflation based on singular vectors. We derive the deflation formulation of singular vectors for a general linear systems. It turns out to be a special case of the general

deflation formula presented previously in [12].

Then we derive the approximation strategy for deflating the singular vectors. We compare the combination of the truncation of the deflation subspace and the deflation of singular vectors to the existing method GCRO-DR on a sequence of linear systems arising from reservoir simulation. Since we are interested in solving a sequence of linear systems, we follow the presentation of Parks et al. in [28].

The paper is organized as follows. In the following section, we give a brief review about the Krylov subspace and the Arnoldi procedure, and then we present the deflation subspace strategy. Afterwards, we review the method GCRO-DR [28] with an abstract deflation subspace related to the previous search subspace. To motivate the deflation of the singular vectors, an extension of a result obtained in [36] on the influence of the singular values on the convergence of GMRES is extended to GCRO-DR and presented in Section 3 In Section 4, we introduce the deflation of singular vectors. We derive a formulation of deflation of exact singular vectors. Section 5 presents strategies of computing deflation vectors for GCRO-DR. At first, we recall the Ritz pairs and the harmonic Ritz pairs [27], and then we introduce a strategy of deflating approximate singular vectors. In Section 6, we present the truncation of the deflation subspace in recycling Krylov subspace methods. For each type of recycled vectors we propose a truncation strategy. Section 7 details the parallel implementation of the method and analyzes its computation and communication cost. In Section 8, we compare the three variants of deflation techniques combined with the truncation of the deflation subspace on a sequence of linear systems arising from reservoir simulation.

2 Background

In this section, we review the Krylov subspaces, the Arnoldi procedure, and the deflation of eigenvectors. We review the GCRO-DR algorithm to which we propose a modification in the following sections.

Notation

For an integer $m > 0$ we refer to the $m \times m$ identity matrix as I_m. Let the integers $m > 0$, $p > 0$, and let $h_{ij} \in \mathbb{C}$, for $i = 1, \ldots, m$, $j = 1, \ldots, p$, we refer to the matrix whose (i, j) coefficient is $h_{i,j}$ for $i = 1, \ldots, m$, $j = 1, \ldots, p$ as $(h_{ij})_{\substack{1 \le i \le m \\ 1 \le j \le p}}$. If for some pair (i, j) such that $1 \le i \le m$ and $1 \le j \le p$ the element $h_{i,j}$ is not defined, it is set to zero. Let $H = (h_{i,j})_{\substack{1 \le i \le m \\ 1 \le j \le p}} \in \mathbb{C}^{m \times p}$. we refer to the matrix $(h_{i,j})_{\substack{1 \le i \le m-1 \\ 1 \le j \le p}} \in \mathbb{C}^{(m-1) \times p}$ as $\overline{H}$. Unless otherwise stated, the matrix $(h_{i,j})_{\substack{2 \le i \le m \\ 1 \le j \le p}} \in \mathbb{C}^{(m-1) \times p}$ is referred to as $\underline{H}$. The $m \times p$ matrix with zeros elements is referred to as $0_{m,p}$ and the subscript is omitted when the size is clear from the context.

2.1 Krylov subspaces and Arnoldi procedure

The kth Krylov subspace associated to A and b is defined as

$$K_k(A, b) = \text{span}\left\{b, Ab, \ldots, A^{k-1}b\right\}. \tag{2}$$

Let

$$k_{A,b} = \min_{\substack{P \in \mathcal{P} \\ P(0)=1}} \{deg(P) \mid P(A)b = 0\},$$

where $\mathcal{P}$ represents the set of polynomials with complex coefficients. The vectors $b, Ab, \ldots, A^{k-1}b$ for $k \leq k_{A,b}$ form a basis of the Krylov subspace. Due to round-off errors, an orthogonalization procedure is necessary to construct an orthogonal basis of the Krylov subspace. To this aim, the Arnoldi procedure is widely used, see Algorithm 1.

Algorithm 1 Arnoldi (A, v_1, m)

Require: normal vector $v_1 \in \mathbb{C}^n$, matrix $A \in \mathbb{C}^{n \times n}$, number of iterations m
Ensure: orthonormal basis vectors V_m, Hessenberg matrix $H_m \in \mathbb{C}^{(m+1) \times m}$
 1: **for** $j = 1 : m$ **do**
 2:     $w = Av_j$
 3:     **for** $i = 1 : j$ **do**
 4:         $h_{ij} = v_i^H w$
 5:     **end for**
 6:     $w = w - \sum_{i=1}^{j} h_{ij} v_i$
 7:     $h_{j+1,j} = \|w\|_2$
 8:     **if** $h_{j+1,j} = 0$ **then**
 9:         Set $m = j$, $v_{m+1} = 0$
10:     **else**
11:         $v_{j+1} = \frac{w}{h_{j+1,j}}$
12:     **end if**
13:     $V_j = [v_1, \cdots, v_j]$, $V_{j+1} = [V_j, v_{j+1}]$, $H_j = (h_{ik})_{\substack{1 \leq i \leq j+1 \\ 1 \leq k \leq j}}$
14: **end for**

There are multiple orthogonalization procedures that can be used in the Arnoldi procedure. We refer the reader to [20, 13, 14] for analysis on their behavior.

According to the notations in Algorithm 1 the following relation holds at each iteration k

$$AV_k = V_{k+1}H_k, \tag{3}$$

where $V_k = \{v_1, \ldots, v_k\}$, $V_{k+1} = [V_k, v_{k+1}]$, and $H_k = (h_{i,j})_{\substack{1 \leq i \leq k+1 \\ 1 \leq j \leq k}}$.

2.2   Deflation of eigenvectors

Let S be an A-invariant subspace of dimension $k \geq 0$ related to the smallest eigenvalues in magnitude of the matrix A. Let $Z \in \mathbb{C}^{n \times k}$ be a matrix whose columns form an orthonormal basis of the subspace S, and let T be the projection of A on the subspace S, i.e., $T = Z^H A Z$. We note that the following relation holds

$$AZ = ZT.$$

Let $\{\lambda_1, \ldots, \lambda_n\}$ be the eigenvalues of A ordered increasingly by the magnitude of their absolute value, i.e., $|\lambda_1| \leq \ldots \leq |\lambda_n|$. Consider the preconditioned system

$$(I + Z(T^{-1} - I)Z^H)Ax = (I + Z(T^{-1} - I)Z^H)b \tag{4}$$

The following Theorem 1 can be found in [9]. It describes how to deflate the eigenvalues of the matrix A by adding a low-rank correction.

Theorem 1 *The eigenvalues of the matrix $A + Z(T^{-1} - I)Z^H A$ are $\{1, \ldots, 1, \lambda_{k+1}, \ldots, \lambda_n\}$.*

The equation (4) can be rewritten as :

$$\left((A - ZZ^H A) + ZT^{-1}Z^H A \right) x = \left((b - ZZ^H b) + ZT^{-1}Z^H b \right)$$

In other words, to solve (1), we look for the solution in two subspaces, the first is S, the second is the orthogonal subspace to S. The first finds the part of the solution that is generated by the basis vectors V. This is done by solving the deflation subspace system with the matrix T. The second finds the part that is orthogonal to S. This part is left to the Krylov method. Since the eigenvectors of the matrix are not orthogonal in general, we need at each iteration of the Krylov method to correct the new computed basis vector. In the Hermitian case, this correction can be avoided by choosing a specific initial guess, see [40].

2.3 Ritz pairs

In this paragraph we review the definition of Ritz pairs of a matrix related to a subspace.

Definition 1 *Let $B \in \mathbb{C}^{n \times n}$ and let $\mathcal{S}$ be a subspace of dimension $k < n$. We say that the pair $(u, \theta) \in \mathbb{C}^n \times \mathbb{C}$ is a Ritz pair of B related to the subspace $\mathcal{S}$ if*

$$\begin{cases} u \in \mathcal{S}, \\ y^H(Bu - \theta u) = 0, \forall y \in S. \end{cases} \tag{5}$$

In the case where the subspace $\mathcal{S} = K_k(A, b)$, the Ritz pairs of the matrix A correspond to

$$\mathcal{R} = \{(V_k u, \theta), \overline{H}_k u = \theta u\},$$

see Algorithm 1. If we consider the subspace $\mathcal{S} = AK_k(A, b)$, the Ritz pairs of the matrix A^{-1} correspond to

$$\mathcal{R} = \{(V_k u, \theta), \overline{H}_k^H u = \theta H_k^H H_k u\}.$$

Lemma 1 *Let $B \in \mathbb{C}^{m \times m}$ and let S be a subspace of dimension $k < m$ that contains an eigenpair (λ, u) of the matrix B. Let $V \in \mathbb{C}^{m \times k}$ be a matrix whose columns form an orthonormal basis of S. Then, there exists a vector $w \in \mathbb{C}^k$ such that (λ, w) is an eigenpair of $V^H B V$, the restriction of B to the subspace S. Furthermore, $w = V^H u$.*

Proof 1 *The proof is simple and direct.*

In Section 5, we will review in detail the derivation of these Ritz pairs. Furthermore, we derive the Ritz pairs relative to the singular vectors and values of A.

In the following, we review the GCRO-DR method and the construction of an abstract deflation subspace related to it.

2.4 GCRO-DR

We recall that GCRO-DR is an iterative method for solving a sequence of linear systems which takes advantage of the history of previous search subspaces by performing recycling. In [28], the authors present the algorithm with a deflation subspace related to the harmonic Ritz vectors [27] as well as another deflation subspace that was previously proposed in [6]. In the following, we explain briefly the method. We note that most of the following algebraic results can be found in [5]. Given the sequence of linear systems $A_i x = b_i$, for $i = 1, \ldots, p$, where $p > 1$, the method GCRO-DR(m, k) works as the following. To solve the linear system $A_1 x = b_1$, the

first cycle, which is a sequence of iterations between two restarts, is performed as a usual GMRES cycle. The subspace relation that holds at the end of the cycle is the usual Arnoldi relation :

$$A_1 V_m = V_{m+1} H_m.$$

The solution x_1 and the residual r_1 are computed as usual. Based on a given strategy, a column matrix $P_k \in \mathbb{C}^{m \times k}$ is computed (Section 5 details the computation of this matrix following different strategies.) This matrix is expanded by V_m to form the basis of the deflation subspace $\tilde{Y}_k = V_m P_k$. The image of the deflation subspace by A_1 is computed

$$A_1 V_m P_k = V_{m+1} H_m P_k,$$

then orthonormalized,

$$C_k = V_{m+1} Q,$$

where Q stands for the unitary factor of the QR factorization of $H_m P_k = QR$. The relation that holds at this stage is :

$$A_1 \tilde{Y}_k R^{-1} = C_k.$$

We set $U_k = \tilde{Y}_k R^{-1}$. In order to avoid round-off errors, the vectors of U_k are normalized. We define the diagonal matrix D_k such that the following holds

$$\tilde{U}_k = U_k D_k^{-1},$$
$$A_1 \tilde{U}_k = C_k D_k.$$

At that moment, the method is ready to carry on the next cycle, $m - k$ Arnoldi iterations are performed with $(I - C_k C_k^H) A_1$ as an operator and the normalized residual as the starting vector. The following relation holds

$$A_1 [\tilde{U}_k, V_{m-k}] = [C_k, V_{m-k+1}] G_m,$$

where V_{m-k+1} is the resulting orthonormal basis constructed by the Arnoldi procedure, $G_m = \begin{pmatrix} D_k & B_{m-k} \\ 0 & H_{m-k} \end{pmatrix}$, $B_{m-k} = C_k^H A_1 V_{m-k}$ and H_{m-k} is the resulting Hessenberg matrix of the Arnoldi procedure. The solution x and the residual r are computed following the constraint $\|r_2\|_2 = \|r_1 - A_1 x\|_2$ has a minimal norm over the subspace spanned by the columns of $[\tilde{U}_k, V_{m-k}]$. The extraction of the deflation subspace is performed as before by replacing the matrix H_m by G_m. This cycle is repeated until the convergence is achieved. In order to solve the second linear system, all what is necessary is to update the image of the deflation subspace, i.e., C_k, by the image of the deflation subspace by the new matrix A_2. Algorithm 2 presents the GCRO-DR method with an abstract deflation subspace. We will discuss the deflation variants in Section 5.

Considering the notations in Algorithm 2, if the matrix $\tilde{Y}_k$ is not defined, then (3) holds. Otherwise, the following relation holds

$$A[\tilde{U}_k, V_{m-k}] = [C_k, V_{m-k+1}] \begin{pmatrix} D_k & B_{m-k} \\ 0 & H_{m-k} \end{pmatrix}. \tag{6}$$

The matrix $[C_k, V_{m-k+1}]$ is unitary, but this is not true, in general, for the matrix $[\tilde{U}_k, V_{m-k}]$. In this case the subspace generated by the columns of $\tilde{U}_k, V_{m-k}$ is not a Krylov subspace.

Algorithm 2 GCRO-DR

Require: the maximum dimension of the search subspace m, the dimension of the deflation subspace k, the convergence tolerance ε, the initial guess x_0.

Ensure: approximate solution $\tilde{x}$ of $Ax = b$.

1: initialization $r_0 = b - Ax_0$, $i = 1$
2: **if** $\tilde{Y}_k$ is defined (from solving a previous linear system) **then**
3:     let $[Q, R]$ be the QR factorization of $A\tilde{Y}_k$
4:     $C_k = Q$
5:     $U_k = \tilde{Y}_k R^{-1}$
6:     $x_1 = x_0 + U_k C_k^H r_0$
7:     $r_1 = r_0 - C_k C_k^H r_0$
8: **else**
9:     $v_1 = r_0 / \|r_0\|$
10:     $c = \|r_0\|_2 e_1$
11:     perform m steps of GMRES, solving $\min \|c - H_m y\|_2$ for y and generating V_{m+1} and H_m
12:     $x_1 = x_0 + V_m y$
13:     $r_1 = V_{m+1}(c - H_m y)$
14:     compute k deflation vectors P_k
15:     $\tilde{Y}_k = V_m P_k$
16:     let $[Q, R]$ be the QR factorization of $H_m P_k$
17:     $C_k = V_{m+1} Q$
18:     $U_k = \tilde{Y}_k R^{-1}$
19: **end if**
20: **while** $\|r_i\|_2 > \varepsilon$ **do**
21:     $i = i + 1$
22:     perform $m - k$ Arnoldi steps with the linear operator $(I - C_k C_k^H)A$, letting $v_1 = r_{i-1}/\|r_{i-1}\|_2$ and generating V_{m-k+1}, H_{m-k}, and $B_{m-k} = C_k^H A V_{m-k}$
23:     let D_k be a diagonal scaling matrix such that $\tilde{U}_k = U_k D_k$, where the columns of $\tilde{U}_k$ have a unit norm
24:     $\hat{V}_m = [\tilde{U}_k, V_{m-k}]$, $\hat{W}_{m+1} = [C_k, V_{m-k+1}]$
25:     $G_m = \begin{pmatrix} D_k & B_{m-k} \\ 0 & H_{m-k} \end{pmatrix}$
26:     solve $min \|G_m y - \hat{W}_{m+1}^H r_{i-1}\|_2$ for y
27:     $x_i = x_{i-1} + \hat{V}_m y$
28:     $r_i = r_{i-1} - \hat{W}_{m+1} G_m y$
29:     compute k deflation vectors P_k
30:     $\tilde{Y}_k = \hat{V}_m P_k$
31:     let $[Q, R]$ be the QR factorization of $G_m P_k$
32:     $C_k = \hat{W}_{m+1} Q$
33:     $U_k = \tilde{Y}_k R^{-1}$
34: **end while**
35: let $\tilde{Y}_k = U_k$

---

3   On the convergence of GCRO-DR

Here, we extend a theoretical result obtained in [36] on the convergence of GMRES to GCRO-DR with an abstract recycling subspace. We suppose in the following that the recycling subspace is defined. Without loss of generality, we suppose that the residual is normalized. At iteration j of the GCRO-DR algorithm, the minimization problem in the search subspace is :

$$y_j = arg \min_{y \in \mathbb{C}^{j+k}} \|e_{k+1} - G_{j+k} y\|_2.$$

The following proposition generalizes [36, Proposition 4.1]

Proposition 1 *At iteration j of the GCRO-DR algorithm (the inner GMRES iteration j with the the matrix coefficient $(I - C_k C_k^H)A$), the following holds :*
 — the solution of the minimization problem y_j is :

$$y_j = (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1} \underline{G}_{j+k}^{-1} \underline{G}_{j+k}^{-H} \tilde{h}_j. \tag{7}$$

 — the projection of r_j on v_1 is

$$v_1^H r_j = (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1} \tag{8}$$

 — the projection of r_j on $[C_k, v_2, \ldots, v_{j+1}]$ is

$$[C_k, v_2, \ldots, v_{j+1}]^H r_j = (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1} \begin{pmatrix} 0 \\ \underline{H}_j^{-H} h_j \end{pmatrix} \tag{9}$$

 — $\cos^2(\theta(v_1, r_j)) = \|r_j\|_2^2 = (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1}$
where $v_1, \ldots, v_{j+1}$ are the basis vectors obtained by the Arnoldi procedure, $\tilde{h}_j^\top = [0, h_j^\top] \in \mathbb{R}^{k+j}$, $\underline{G}_{j+k} = \begin{pmatrix} D_k & B_j \\ 0 & \underline{H}_j \end{pmatrix}$, and $\theta(v_1, r_j)$ is the angle between the vectors v_1 and r_j.[1]

Proof 2

$$y_j = (G_{j+k}^H G_{j+k})^{-1} G_{j+k}^H e_{k+1},$$
$$= (\underline{G}_{j+k}^H \underline{G}_{j+k} + \tilde{h}_j \tilde{h}_j^H)^{-1} \tilde{h}_j,$$

Using the Sherman-Morrison formula, we have

$$y_j = (1 + \|\underline{G}_{j+k}^{-H} \tilde{h}_j\|_2^2)^{-1} \underline{G}_{j+k}^{-1} \underline{G}_{j+k}^{-H} \tilde{h}_j,$$

Exploiting the structure of $\tilde{h}_j$ and $\underline{G}_{j+k}$ we can write $(\underline{G}_{j+k}^{-H} \tilde{h}_j)^\top = [0, \underline{H}_j^{-H} h_j]^\top$. Then, we have the relation required,

$$y_j = (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1} \underline{G}_{j+k}^{-1} \underline{G}_{j+k}^{-H} \tilde{h}_j.$$

Furthermore, we can write

$$y_j = (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1} \begin{pmatrix} -D_k^{-1} B_j \underline{H}_j^{-1} \underline{H}_j^{-H} h_j \\ \underline{H}_j^{-1} \underline{H}_j^{-H} h_j \end{pmatrix}.$$

---

1. Note that $\underline{G}_{j+k}$ is not the matrix obtained from G_{j+k} by suppressing the first line as for $\underline{H}_j$.

The residual in GCRO-DR can be expressed as :

$$r_j = b - A[\tilde{U}_k, V_j]y_j,$$

$$= [\tilde{C}_k, V_{j+1}][e_{k+1}, G_{j+k}] \begin{bmatrix} 1 \\ -y_j \end{bmatrix},$$

$$= [C_k, v_1, v_2, \ldots, v_{j+1}] \begin{bmatrix} 0 & D_k & B_j \\ 1 & 0 & h_j^H \\ 0 & 0 & \underline{H}_j \end{bmatrix} \begin{bmatrix} 1 \\ -y_j \end{bmatrix},$$

$$= [v_1, C_k, v_2, \ldots, v_{j+1}] \begin{bmatrix} 1 & 0 & h_j^H \\ 0 & D_k & B_j \\ 0 & 0 & \underline{H}_j \end{bmatrix} \begin{bmatrix} 1 \\ -y_j \end{bmatrix},$$

$$= [v_1, C_k, v_2, \ldots, v_{j+1}] \begin{bmatrix} 1 & \tilde{h}_j^H \\ 0 & \underline{G}_{j+k} \end{bmatrix} \begin{bmatrix} 1 \\ -y_j \end{bmatrix},$$

$$= v_1(1 - \tilde{h}_j^H y_j) - [C_k, v_2, \ldots, v_{j+1}]\underline{G}_{j+k} y_j.$$

Combining the last relation with (7), *we have*

$$v_1^H r_j = 1 - (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1}\|\underline{G}_{j+k}^{-H} \tilde{h}_j\|^2,$$

$$= 1 - (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1}\|\underline{H}_j^{-H} h_j\|^2,$$

$$= (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1}.$$

and

$$[C_k, v_2, \ldots, v_{j+1}]^H r_j = \underline{G}_{j+k} y_j,$$

$$= (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1}\underline{G}_{j+k}^{-H} \tilde{h}_j,$$

$$= (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1} \begin{pmatrix} 0 \\ \underline{H}_j^{-H} h_j \end{pmatrix}.$$

The norm of the residual r_j is given as :

$$\|r_j\|_2^2 = (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-2} + (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-2}\|\underline{H}_j^{-H} h_j\|_2^2$$

$$= (1 + \|\underline{H}_j^{-H} h_j\|_2^2)^{-1}.$$

To prove the last relation, it is sufficient to note that :

$$v_1^H r_j = \cos(\theta(v_1, r_j))\|r_j\|_2$$

Proposition 1 shows that the norm of the residual starts to decrease when the value $\|\underline{H}_j^{-1} h\|_2^2$ is large. Moreover, if $\|h_j\|_2$ is very small and $\|\underline{H}_j^{-1}\|_2$ is not sufficiently large the method will stagnate. As discussed in [36], the value of $\|h_j\|_2$ may depend on different factors. A focus in [36] is on the case when the right-hand side has large components corresponding to the left singular vectors associated with small singular values of the coefficient matrix. For example, let (u_k, σ_k, w_k) be the kth singular triplet of A; if $v_1 = u_n$, then $h_j^H = \sigma_n w_n^H V_j$ and $\|h_j\|_2 \leq \sigma_n$. This can be easily generalized to the case $v_1 = \alpha_n u_n + \ldots + \alpha_k u_k$ for some $k \in [\![1; n-1]\!]$, where $\sum_{i=k}^n \alpha_k^2 = 1$, and $\alpha_j \ll \alpha_n$, for $k \leq j < n$. Suppressing these components by deflating the right singular vectors associated with the smallest singular values avoids the previous situation.

In the following section we discuss the deflation of exact singular vectors.

4   Deflation based on singular vectors

In the previous section we extended a result obtained in [36] on the convergence of GMRES to the GCRO-DR method. As previously mentioned, the work of Simoncini on the convergence of several restarted Krylov methods and its relation with the singular vectors associated to the smallest singular values [36] motivated us to investigate the deflation based on singular vectors approximation. This section introduces the deflation of singular vectors. Given a set of singular vectors (right or left), we show how to deflate these vectors during the Krylov method.

Theorem 2 *Let x_* be the exact solution of* (1). *Let*

$$A = U\Sigma V^H$$

be the singular value decomposition of A such that the singular values are ordered increasingly. Let k_τ be the number of singular values smaller than a given threshold $\tau > 0$. Consider the splitting of the SVD decomposition

$$A = U_1 \Sigma_1 V_1^H + U_2 \Sigma_2 V_2^H,$$

where $\Sigma_2 \in \mathbb{C}^{k_\tau \times k_\tau}$ is a diagonal matrix whose diagonal elements are the singular values smaller than τ. Consider $\tilde{x}$ an approximate solution of the following linear system of equations

$$(I - U_2 U_2^H)Ax = (I - U_2 U_2^H)b, \tag{10}$$

such that $\|\hat{x} - \tilde{x}\|_2 \leq \varepsilon$, where $\hat{x}$ is an exact solution of (10) *and $\varepsilon > 0$. Then, the error norm in the solution of the original system, corresponding to the approximate solution of the associated projected system* (10) *and the correction term, is bounded by the error norm in the solution of the projected system* (10), *i.e., the following inequality holds*

$$\|x_* - (I - V_2 V_2^H)\tilde{x} - V_2 \Sigma_2^{-1} U_2^H b\|_2 \leq \varepsilon.$$

Proof 3 *First, we remark that x_* is a solution of* (10) *and the set of solutions of* (10) *can be written as $\mathcal{S} = \{x = x_* + V_2 u, \ u \in \mathbb{C}^{k_\tau}\}$. Indeed, let x be a solution of* (10). *Then,*

$$Ax = (I - U_2 U_2^H)b + U_2 U_2^H Ax,$$
$$Ax = Ax_* - U_2 U_2^H(b - Ax).$$

We multiply by A^{-1} both sides. We obtain

$$\begin{aligned} x &= x_* + V_2 \Sigma_2^{-1} U_2^H (Ax - b) \\ &= x_* + V_2 \Sigma_2^{-1} (\Sigma_2 V_2^H x - U_2^H b) \\ &= x_* + V_2 V_2^H x - V_2 \Sigma_2^{-1} U_2^H b. \end{aligned}$$

On the other side, each element of $\mathcal{S}$ is a solution of (10).
    We have

$$\begin{aligned} V_2 V_2^H x_* &= V_2 V_2^H A^{-1} b \\ &= V_2 \Sigma_2^{-1} U_2^H b. \end{aligned}$$

Hence, we can write

$$\begin{aligned} x_* &= (I - V_2 V_2^H)x_* + V_2 V_2^H x_*, \\ &= (I - V_2 V_2^H)\hat{x} + V_2 \Sigma_2^{-1} U_2^H b, \end{aligned}$$

thus,

$$x_* - (I - V_2V_2^H)\tilde{x} - V_2\Sigma_2^{-1}U_2^H b = (I - V_2V_2^H)(\hat{x} - \tilde{x}).$$

Finally, we obtain

$$\begin{aligned}\|x_* - (I - V_2V_2^H)\tilde{x} - V_2\Sigma_2^{-1}U_2^H b\|_2 &\leq \|(I - V_2V_2^H)(\hat{x} - \tilde{x})\|_2, \\ &\leq \|\hat{x} - \tilde{x}\|_2, \\ &\leq \varepsilon.\end{aligned}$$

Note that the matrix $(I - U_2U_2^H)A$ is singular. However, solving the linear system of equations (10) with a Krylov method is possible. The approximate solution of (1) can be computed as

$$x = (I - V_2V_2^H)\tilde{x} + V_2x_2,$$

where $x_2 = \Sigma_2^{-1}U_2^H b$ and $\tilde{x}$ is the approximate solution given by the Krylov method. In [12], the authors presented a general formulation of deflation by using an abstract deflation subspace. Theorem 2 is a special case for a subspace associated with singular vectors.

The following section discusses different approaches for computing deflation information from a Krylov subspace. Two approaches are based on approximating eigenvectors of the matrix, and these approaches can be found in literature [24, 28]. A third approach is introduced here and it is based on approximating the right singular vectors of the matrix.

5   Computing deflation vectors

In this section, we derive three different types of deflation subspaces. These subspaces are considered in the context of GCRO-DR [28]. In Section 2, we reviewed the method GCRO-DR for an abstract deflation subspace issued from the last Krylov subspace. Even though any deflation vectors can be chosen for GCRO-DR, this choice has an impact on the convergence of the method. Several types of deflation vectors have been presented in previous works [24, 9, 28]. We focus on two types which are widely used and based on the smallest and largest Ritz values of the matrices A, A^{-1}, respectively. During the GCRO-DR method, one of the relations (3) and (6) holds. In the following, we present how to compute the deflation vectors mentioned above. These strategies were introduced in previous works [24, 9, 28]. We present them for the sake of completeness. Afterwards, we introduce the deflation vectors that are related to the right singular vectors.

We consider the following environment. Let K be a search subspace (Krylov subspace or else) of dimension m associated to the matrix A, let $V \in \mathbb{C}^{n \times m}$ be a matrix whose columns form a basis of K. Let $W \in \mathbb{C}^{n \times (m+1)}$ be a unitary matrix such that the following relation holds

$$AV = WG, \tag{11}$$

where $G \in \mathbb{C}^{(m+1) \times m}$ is a Hessenberg matrix.

We will derive the eigenvalue problem that is solved in order to compute the deflation vectors. For each approximation strategy, we formulate the computation following the notations in Algorithm 2. Thus, we will distinguish two cases in each strategy, depending on whether the deflation subspace already exists or not.

5.1   Approximation based on the smallest Ritz values of A

In this case, the Ritz pairs (u, θ) verify

$$
\begin{cases}
u \in K, \\
y^H(Au - \theta u) = 0, \forall y \in K.
\end{cases}
$$

Since the columns of V span the subspace K, we have

$$
V^H(AVw - \theta Vw) = 0,
$$

where $u = Vw$. This can be written as

$$
V^H W G w = \theta V^H V w.
$$

By adapting the notations in Algorithm 2, the deflation vectors can be computed by solving the following eigenvalue problem

$$
\begin{cases}
\overline{H}_m w = \theta w, & \text{if } \tilde{Y}_k \text{ is not defined} \\
\begin{pmatrix} \tilde{U}_k^H C_k & \tilde{U}_k^H V_{m-k+1} \\ 0 & (I_{m-k}, 0) \end{pmatrix} G_m w = \theta \begin{pmatrix} \tilde{U}_k^H \tilde{U}_k & \tilde{U}_k^H V_{m-k} \\ V_{m-k}^H \tilde{U}_k & I_{m-k} \end{pmatrix} w, & \text{if } \tilde{Y}_k \text{ is defined.}
\end{cases} \tag{12}
$$

5.2   Approximation based on the largest Ritz values of A^{-1}

To approximate the eigenvectors of A^{-1} we look for the approximate vectors in the subspace AK. We write

$$
\begin{cases}
u \in AK, \\
y \perp (A^{-1}u - \theta u), \forall y \in AK,
\end{cases}
$$

The columns of AV span the subspace AK. Hence, we have

$$
V^H A^H (Vw - \theta AVw) = 0,
$$

this can be written as

$$
G^H W^H V w = \theta G^H G w.
$$

By using the notations of Algorithm 2, this requires solving the following generalized eigenvalue problem

$$
\begin{cases}
\overline{H}_m^H w = \theta H_m^H H_m w, & \text{if } \tilde{Y}_k \text{ is not defined} \\
G_m^H \begin{pmatrix} C_k^H \tilde{U}_k & 0 \\ V_{m-k+1}^H \tilde{U}_k & \begin{pmatrix} I_{m-k} \\ 0 \end{pmatrix} \end{pmatrix} w = \theta G_m^H G_m w, & \text{if } \tilde{Y}_k \text{ is defined.}
\end{cases} \tag{13}
$$

We note that Morgan, in his paper [24], refers to the values θ^{-1} as the *harmonic* Ritz values.

5.3   Approximation based on the smallest Ritz values of $A^H A$

We follow the definition of the Ritz pairs given in Definition 1. In order to approximate the right singular vectors of the matrix A, we consider an approximation of the eigenvectors of the matrix $A^H A$. Thus, we are looking for the pair (u, θ) such that

$$u \in K,$$
$$y \perp (A^H A u - \theta u), \forall y \in K$$

The columns of V span the subspace K, the previous relation can be written as

$$V^H (A^H A V w - \theta V w) = 0,$$

where $u = Vw$. This can be written as

$$G^H G w = \theta V^H V w.$$

The formulation of this eigenvalue problem in the context of Algorithm 2 is as follows

$$\begin{cases} H_m^H H_m w = \theta w, & \text{if } \tilde{Y}_k \text{ is not defined} \\ G_m^H G_m w = \theta \begin{pmatrix} \tilde{U}_k^H \tilde{U}_k & 0 \\ 0 & I_{m-k} \end{pmatrix} w, & \text{if } \tilde{Y}_k \text{ is defined.} \end{cases} \tag{14}$$

6   Deflation subspace truncation

In this section, we introduce a strategy to reduce the dimension of the deflation subspace. Consider a sequence of linear systems solved by GCRO-DR. This method builds or updates a deflation subspace during the solution of each linear system in the sequence. This deflation subspace is used to potentially enhance the convergence of the iterative method while solving the following system. However, part of the deflation subspace may not be useful for improving the convergence when the matrix changes. When the matrices in the sequence are Hermitian, the worst case that can happen is that the deflation subspace does not enhance the convergence; see [40]. Nonetheless, this is not the case for non-Hermitian matrices, where an improper choice of the deflation subspace may even deteriorate the convergence. For this reason, one needs a strategy to extract the useful and relevant information from the deflation subspace and removes the improper part of it. Such a strategy should keep in the truncated deflation subspace a vector that is a good approximate of an eigenvector associated with a small eigenvalue in magnitude of $A, A^H A$, or large eigenvalue in magnitude of A^{-1} depending on the deflation strategy.

One approach can be based on the accuracy of the approximated eigenvectors. Nevertheless, it was noticed in [28, 29] that GCRO-DR does not require high accuracy of the approximated eigenvectors in order to improve the convergence. A practical selection criterion for an accuracy-based approach needs the estimation of the residual. This may be a very strong criterion since it requires not only a good approximation of the eigenvectors but the eigenvalues as well. Hence, this can lead to remove the deflation subspace entirely even though part of it might be useful.

The strategy that we propose is based on Ritz values of the deflation strategy matrix ($A, A^H A$, or A^{-1}) with respect to the deflation subspace.

Let B be the matrix which defines the deflation strategy, $B = A, A^H A$, or A^{-1}. Suppose that the deflation subspace contains an exact eigenvector u of B associated with the eigenvalue θ that is smaller in magnitude than τ, τ^2, or larger in magnitude than $\frac{1}{\tau}$, respectively, where $\tau > 0$ is a

predefined value. Lemma 1 proves that the projection of the vector u to the deflation subspace is an eigenvector of the projected matrix to the deflation subspace.

Therefore, on one hand, this strategy allows to keep in the truncated deflation subspace a vector that is a good approximate of an eigenvector associated with a small eigenvalue in magnitude of $A, A^H A$, or large eigenvalue in magnitude of A^{-1} depending on the deflation strategy. On the other hand, this strategy relies only on the Ritz values.

The threshold τ can be chosen to be a relative value to either the magnitude of the largest approximated eigenvalue or the largest approximated singular value depending on the deflation strategy. The truncation strategy has to be associated with the recycling strategy, i.e., if the recycling strategy deflates the Ritz vectors associated with the smallest (resp. largest) Ritz values, the truncation strategy disregards the Ritz vectors associated with the largest (resp. smallest) Ritz values. The truncated subspace is then a subset of the original deflation subspace.

In the following, we derive the generalized eigenvalue problems that are necessary to perform the truncation of the deflation subspace. We suppose that the columns of the unitary matrix Z span the deflation subspace $\mathcal{S}$. Table 1 presents the eigenvalue problems to be solved in order to truncate the deflation subspace corresponding to each strategy of deflation.

Reference matrix	Eigenvalue problem	Disregarded part
A	$Z^H A Z w = \theta w$	largest magnitude
A^{-1}	$Z^H A^H Z w = \theta Z^H A^H A Z w$	smallest magnitude
$A^H A$	$Z^H A^H A Z w = \theta w$	largest magnitude

TABLE 1 – Generalized eigenvalue problems truncating the deflation subspace, *Reference matrix* stands for the matrix whose eigenvectors are approximated, *Eigenvalue problem* refers to the problem to solve in order to truncate the deflation subspace, *Disregarded part* refers to the part of the spectrum to which are associated the disregarded vectors.

We note that for the case of an approximation of singular vectors it is more robust to perform an SVD factorization of AZ rather than solving the eigenvalue problem $Z^H A^H A Z w = \theta w$.

Algorithm 3 Deflation subspace truncation

Require: matrix A, deflation basis vectors $\tilde{Y}$, threshold τ, def the matrix to which the deflation subspace is related

Ensure: truncated deflation basis vectors $\tilde{Y}$

1: compute the QR factorization of $\tilde{Y} = ZR$
2: **if** $def = A$ **then**
3:     solve $Z^H A Z w = \theta w$
4: **else if** $def = A^{-1}$ **then**
5:     solve $Z^H A^H A Z w = \theta Z^H A^H Z w$
6: **else if** $def = A^H A$ **then**
7:     solve $Z^H A^H A Z w = \theta^2 w$
8: **end if**
9: form $\tilde{Y}$ whose columns are $\{Zw \mid w$ is an eigenvector associated with $\theta < \tau\}$

Here, we propose the method Generalized Minimal Residual with Modified Deflated Restarting (GMRES-MDR), Algorithm 4. This method is based on the GCRO-DR method. The main difference is that an estimation of the previous deflation subspace (when the matrix in the sequence changes) is performed in order to keep only relevant vectors.

Algorithm 4 GMRES-MDR

Require: the maximum dimension of the search subspace m, the maximum dimension of the deflation subspace k_{max}, the convergence tolerance ε, the initial guess x_0, the matrix to which the deflation subspace is related def, τ threshold of truncation of the deflation subspace.

Ensure: approximate solution $\tilde{x}$ of $Ax = b$.

1: initialization $r_0 = b - Ax_0$, $i = 1$, $k = k_{max}$
2: **if** $\tilde{Y}_k$ is defined (from solving a previous linear system) **then**
3:      call Algorithms 3 and get $\tilde{Y}_k$, where k is the number of columns in $\tilde{Y}_k$
4:      let $[Q, R]$ be the QR factorization of $A\tilde{Y}_k$
5:      $C_k = Q$
6:      $U_k = \tilde{Y}_k R^{-1}$
7:      $x_1 = x_0 + U_k C_k^H r_0$
8:      $r_1 = r_0 - C_k C_k^H r_0$
9: **else**
10:      $v_1 = r_0 / \|r_0\|$
11:      $c = \|r_0\|_2 e_1$
12:      perform m steps of GMRES, solving $\min \|c - H_m y\|_2$ for y and generating V_{m+1} and H_m
13:      $x_1 = x_0 + V_m y$
14:      $r_1 = V_{m+1}(c - H_m y)$
15:      compute k deflation vectors P_k following def and Section 5
16:      $\tilde{Y}_k = V_m P_k$
17:      let $[Q, R]$ be the QR factorization of $H_m P_k$
18:      $C_k = V_{m+1} Q$
19:      $U_k = \tilde{Y}_k R^{-1}$
20: **end if**
21: **while** $\|r_i\|_2 > \varepsilon$ **do**
22:      $i = i + 1$
23:      perform $m - k$ Arnoldi steps with the linear operator $(I - C_k C_k^H)A$, letting $v_1 = r_{i-1}/\|r_{i-1}\|_2$ and generating V_{m-k+1}, H_{m-k}, and $B_{m-k} = C_k^H A V_{m-k}$
24:      let D_k be a diagonal scaling matrix such that $\tilde{U}_k = U_k D_k$, where the columns of $\tilde{U}_k$ have a unit norm
25:      $\hat{V}_m = [\tilde{U}_k, V_{m-k}]$, $\hat{W}_{m+1} = [C_k, V_{m-k+1}]$
26:      $G_m = \begin{pmatrix} D_k & B_{m-k} \\ 0 & H_{m-k} \end{pmatrix}$
27:      solve $\min \|G_m y - \hat{W}_{m+1}^H r_{i-1}\|_2$ for y
28:      $x_i = x_{i-1} + \hat{V}_m y$
29:      $r_i = r_{i-1} - \hat{W}_{m+1} G_m y$
30:      set $k = k_{max}$ compute k deflation vectors P_k following def and Section 5
31:      $\tilde{Y}_k = \hat{V}_m P_k$
32:      let $[Q, R]$ be the QR factorization of $G_m P_k$
33:      $C_k = \hat{W}_{m+1} Q$
34:      $U_k = \tilde{Y}_k R^{-1}$
35: **end while**
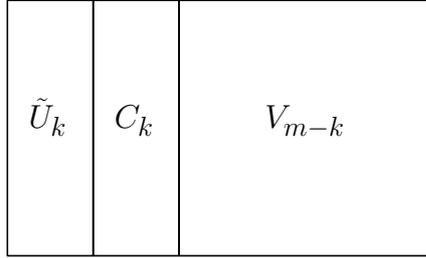36: let $\tilde{Y}_k = U_k$

FIGURE 1 – Deflation and search subspace basis vectors storage for GMRES-MDR method

7   Parallel design of GMRES-MDR

In this section, we describe the parallel design of the method GMRES-MDR. At first, we describe the distribution of data across processors. Then, we analyze the cost of the method and compare the three deflation variants which we use in our implementation.

7.1   Data distribution

Matrices and vectors in our implementation can be separated into two types, distributed and redundant. The matrix A and the search subspace (Krylov and deflation) vectors are distributed in block row wise. All matrices and vectors whose both dimensions are less than or equal to the dimension of the search subspace are stored redundantly on each processor. Using this distribution, the application of the sparse operator requires communication between neighbour processors. Only inner product operations require global communication.

In the following, we describe how data is stored in memory.

7.1.1   Memory management

Dense matrices in iterative solvers are usually stored in one of two ways, either in column major or in row major. We use column major format due to constraints in the reservoir simulator. The vectors of the search subspace are stored in *full storage scheme* since they do not have a special structure. We refer the reader to [18] for more details on storage scheme structures. In our implementation, all dense matrices are stored in full storage scheme. Figure 1 shows how the vectors of the basis are stored in memory. This order $\tilde{U}_k, C_k, V_{m-k}$, in column ordering, has the following advantage. The orthogonalization steps in the Arnoldi procedure and the deflation correction can be done simultaneously. Sparse matrices are stored in the Compressed Sparse Row (CSR) storage scheme.

7.2   Parallel interaction environment and implementation language

We use message passing interface MPI [37] for communication between processors and we use OpenMP for multithreading. These two libraries are compatible with the language C that we chose to use.

Since GMRES-MDR, does not need an explicit coefficient matrix, we use a reverse communication interface (RCI) [7]. Algorithms 5, 6 present the schemes of the RCI loop and the RCI routine, respectively.

Algorithm 5 RCI loop

1: $ido = 0$
2: **while** $ido \,!=\, 99$ **do**
3:   call the reverse communication interface routine 6 and get the value of ido
4:   **if** $ido == 1$ **then**
5:     apply the preconditioned operator
6:   **end if**
7:   **if** $ido == 2$ **then**
8:     apply the preconditioner
9:   **end if**
10: **end while**

Operations	cost (flops)
Preparation of the residual, the solution, and the starting vector	$n(4k + 3)$
Arnoldi orthonormalization	$n(m^2 + 4(m - k) - k^2)$
Hessenberg and residual updates	$2m^2 - 4mk + 2k^2 + 5(m - k)$
Solve the least squares problem and expand the solution	$nm + \frac{1}{2}(m^2 + m)$
Prepare matrices and solve GEVP (Ritz A)	$n(mk + k^2 + k) + 68m^3 + 2m^2$
Prepare matrices and solve GEVP (Ritz A^{-1})	$nmk + 68m^3 + 2m^2$
Prepare matrices and solve GEVP (Ritz $A^H A$)	$nk^2 + 15m^3 + m^2$
Prepare C_k and $\tilde{U}_k$	$nk(2m + k + 3) + m^2 k + m(\frac{5}{2}k^2 + k) - \frac{2}{3}k^3$

TABLE 2 – Detailed computation cost of one cycle of GMRES-MDR supposing that k deflation vector exist. The maximum dimension of the search subspace is m. GEVP stands for generalized eigenvalue problem. Ritz A, Ritz A^{-1}, and Ritz $A^H A$ refer to the deflation strategy.

7.3 Cost analysis

Here, we compare the cost of one cycle of presented variations of GMRES-MDR to the cost of the classical GMRES method. Since the cost of the sparse matrix vector multiplication depends on the sparsity of the matrix A, we do not take it into account in our analysis. In the analysis of the cost of GMRES-MDR, we suppose that the deflation vectors exist i.e., starting from the second cycle. The computation costs consider the real-valued matrices and vectors. The computation costs of the generalized eigenvalue problems can be found in [15]. In Table 2 (resp. 4), we detail the computation (resp. communication) cost of one cycle of GMRES-MDR(m, k). Table 3 (resp. 5) shows a comparison between the computation (resp. communication) costs of the variants of GMRES-MDR and the classical GMRES. It is clear that the computation cost of the SVD variation is the smallest compared to the two other deflation strategies. This cost can be written as :

$$C(n, m, k) = n(m^2 + (2k+5)m + 3k + 3) + 15m^3 + m^2(k + \frac{7}{2}) + m(\frac{5k^2 + 11}{2} - 3k) - \frac{2}{3}k^3 + 2k^2 - 5k.$$

To give an illustration of the difference between the computation cost of the SVD variant of GMRES-MDR and the classical GMRES, we can set $m = 2k$. Then the difference is

$$C(n, 2k, k) - C(n, 2k, 0) = n(5k^2 + 3k) + (9 - \frac{2}{3})k^3 - 4k^2 - 5k.$$

However, it is sufficient that $k > 1$ to make the communication cost of GMRES-MDR smaller than the communication cost of the classical GMRES.

Algorithm 6 RCI routine

Require: ido the interface indicator, parameters of the solver
 1: static variable $status = 0$
 2: **if** $ido == 0$ **then**
 3:     set up the solver using the parameters
 4:     **if** deflation vectors exist **then**
 5:         set $ido = 1$, $status = -1$
 6:         return to apply the preconditioned operator on the deflation vectors
 7:     **else**
 8:         $status = 1$
 9:     **end if**
10: **end if**
11: **if** $status = -1$ **then**
12:     prepare the deflation vectors and matrices
13:     set $ido = 10$, $status = 1$
14:     return
15: **end if**
16: **if** $status = 1$ **then**
17:     prepare for the Arnoldi procedure
18:     set $ido = 1$, $status = 2$ and return
19: **end if**
20: **if** $status = 2$ **then**
21:     orthogonalization of new basis vector and Hessenberg operations and stopping criterion
22:     **if** convergence and max interior iteration are not achieved **then**
23:         $ido = 1$, $status = 2$
24:         return to apply the preconditioned operator on the last basis vector
25:     **else**
26:         solve the least squares problem and compute the solution
27:         compute the deflation subspace
28:         **if** convergence or the max outer iteration is achieved **then**
29:             $ido = 2$, $status = 99$
30:             return to compute the unpreconditioned solution
31:         **else if** max interior iteration is achieved **then**
32:             $ido = 1$, $status = 3$
33:             return to apply the preconditioned operator on the solution
34:         **end if**
35:     **end if**
36: **end if**
37: **if** $status = 3$ **then**
38:     compute the residual
39:     set $ido = 10$, $status = 1$ and return
40: **end if**
41: **if** $status = 99$ **then**
42:     set $ido = 99$, $status = 0$ and return
43: **end if**

Method	cost(flops)
GMRES(m)	$C(n,m,0) - 2n - 15m^3 - m^2$
GMRES-MDR(m,k) (Ritz A)	$C(n,m,k) + nk(m+1) + 53m^3 + m^2$
GMRES-MDR(m,k) (Ritz A^{-1})	$C(n,m,k) + nk(m-k) + 53m^3 + m^2$
GMRES-MDR(m,k) (Ritz $A^H A$)	$C(n,m,k)$

TABLE 3 – Comparison of computation cost of one cycle of GMRES-MDR (supposing that k deflation vector exist) and classical GMRES. The maximum dimension of the search subspace is m. GEVP stands for generalized eigenvalue problem. Ritz A, Ritz A^{-1}, and Ritz $A^H A$ refer to the deflation strategy. $C(n,m,k) = n(m^2 + (2k+5)m + 3k + 3) + 15m^3 + m^2(k + \frac{7}{2}) + m(\frac{5k^2+11}{2} - 3k) - \frac{2}{3}k^3 + 2k^2 - 5k$.

Operations	messages	words
Preparation of the residual, the solution, and the starting vector	2	$(k, 1)$
Arnoldi orthonormalization at iteration j	2	$(j + k, 1)$
Prepare matrices for GEVP (Ritz A)	1	$k(m + k + 1)$
Prepare matrices for GEVP (Ritz A^{-1})	1	$k(m + 1)$
Prepare matrices for GEVP (Ritz $A^H A$)	1	k^2
Prepare $\tilde{U}_k$	1	k

TABLE 4 – Detailed communication costs of one cycle of GMRES-MDR supposing that k deflation vectors exist. The maximum dimension of the search subspace is m. GEVP stands for generalized eigenvalue problem. Ritz A, Ritz A^{-1}, and Ritz $A^H A$ refer to the deflation strategy. When the operation counts 2 messages the number of words is separated into two quantities corresponding to each message. The Arnoldi procedure uses the classical Gram-Schmidt variation.

Method	messages
GMRES(m)	$2m + 1$
GMRES-MDR(m,k)	$2m - 2k + 4$

TABLE 5 – Comparison of communication cost of one cycle of GMRES-MDR (supposing that k deflation vector exist) and classical GMRES. The maximum dimension of the search subspace is m.

8   Numerical experiments

In this section, we illustrate the impact of the proposed method on the convergence of sequences of linear systems arising from reservoir simulation. We present sequential experiments performed in MATLAB as well as parallel experiments. Our set of test cases are obtained from the in-house prototype code at Total, which simulates a complex enhanced oil recovery (EOR) mechanism. This simulator relies on a finite volume discretization and a two-point flux approximation. We have two sequences of linear systems denoted with initials BIGCO24 and BIGP1. BIGCO24 systems arise from the simulation of water and gas injection using a compositional model (8 hydrocarbon components). The permeability field is heterogeneous. The grid has 83587 active cells. BIGP1 systems arise from the simulation of water injection using a black-oil model. The permeability field is heterogeneous (sector model from a real field case). The grid has 42332 active cells.

We are interested in applying recycling techniques in the context of the constrained pressure residual (CPR) preconditioner. At each Newton iteration of the reservoir simulation, a system of linear equations has to be solved. Typical preconditioner is the CPR preconditioner [35, 41]. This preconditioner is based on a two-level approach. For each application of the preconditioner on a vector, the restricted system associated with the pressure variable needs to be solved. Therefore, at each Newton iteration, several systems of linear equations associated with the pressure variable requires the solution. The right-hand side for each of them corresponds to the the restriction of a vector on the global level to the pressure level. This vector can be a Krylov basis vector, a solution, or any vector on which the user wants to apply the CPR preconditioner. On the other hand, the matrix is the same during the whole Newton iteration. It changes, however, from one Newton iteration to the following one. Typically, a low accuracy is sufficient for the restricted pressure system.

In our experiments we use the double classical Gram-Schmidt for the orthogonalization process in the Arnoldi procedure.
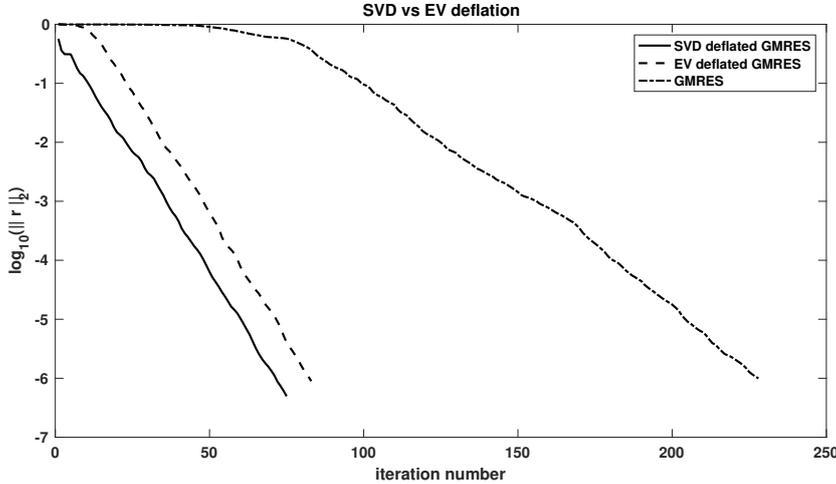
8.1   MATLAB numerical experiments

FIGURE 2 – Convergence history, singular vectors deflated GMRES against eigenvectors deflated GMRES. The convergence history of non-deflated GMRES is also plotted, results for the matrix BIGP1 in MATLAB 2017. The 20 computed approximated vectors (singular vectors and eigenvectors) are approximated up to a tolerance of 10^{-6}.

Figure 2 shows the impact of deflating the singular vectors associated with the smallest singular values. In both variants, 20 vectors are computed approximately to a tolerance of 10^{-6} by using the routines `eigs` and `svds`. The deflation of the eigenvectors corresponds to (4) which we rewrite here,

$$(I + Z(T^{-1} - I)Z^H)Ax = (I + Z(T^{-1} - I)Z^H)b,$$

where Z is a unitary matrix whose columns span the approximated eigenspace. The deflation of the singular vectors follows the Theorem 2. We write the preconditioned linear system

$$(I - U_2 U_2^H)Ax = (I - U_2 U_2^H)b,$$

where $U_2 = AV_2\Sigma_2^{-1}$ stands for the approximated left singular vectors. The solution is computed as $x = (I - V_2 V_2^H)\bar{x} + V_2 x_2$, where $\bar{x}$ is the solution obtained by GMRES and $x_2 = \Sigma_2^{-1}U_2^H b$. We remark that the speed of convergence for both deflated variants are approximately the same with a small advantage for the SVD deflation. Furthermore, in the first iterations the SVD deflated GMRES has a sharp convergence curve while the curve corresponding to the eigenvectors deflated GMRES stagnates for a number of iterations. This phenomenon is important for our application. In each Newton iteration during the simulation of a reservoir, the linearized system is solved by using the CPR solver. An iteration of the latter solves the linear system associated with the pressure variable. This matrix does not change during the CPR solve, only the right-hand side changes. In practice, a large threshold of convergence for the pressure level is sufficient. Thus, having stagnation at the beginning of the method, even for a small number of iterations, can impact the global number of iterations over the entire simulation. The relative tolerance of convergence in the experiments related to the sequences of linear systems is fixed to 10^{-1}.

As it was noted in the thesis of Parks [29], the choice of the deflation subspace has an impact on the convergence of sequences of linear systems. Figure 3 does not only confirm this point but
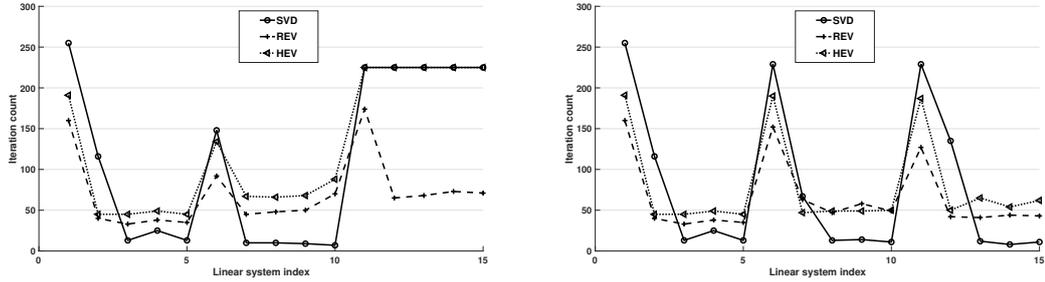
FIGURE 3 – Comparison between three variants of deflation subspaces. SVD, REV, and HEV stand for the deflation methods related to the matrices $A^H A$, A and A^{-1}, respectively. The maximum dimension of the search subspace is 30. The maximum dimension of the deflation subspace is 5. Fifteen linear systems in the sequence BIGP1 are solved, with three matrices, each solved for five different right-hand sides (for index i the three methods share the same right-hand side). In the figure on the left, the iteration counts are obtained by using a solver that does not truncate the deflation subspace. In the figure on the right, the iteration counts are obtained by using a solver that truncates the deflation subspace. The truncation threshold is set to 10^{-3}.

also shows how it might be possible that in some configurations the deflation subspace becomes an issue and leads to the stagnation of the residual norm. For each deflation variant, we solve a sequence of linear systems related to the pressure variable. We have three matrices arising from three contiguous Newton steps. For each matrix, we solve five (resp. ten) linear systems with initial BIGP1 (resp. BIGCO24) with different right-hand sides each given in a time. This mimics the CPR solver. In order to make a fair comparison and avoid eventual large components of the right-hand sides on special eigenvectors or singular vectors, we generate random right-hand sides with seeds corresponding to the index of the linear system. In the same figure we show the impact of the truncation of the deflation subspace. Figure 4 shows how the truncation strategy might lead to lose information that can enhance the convergence. We realize an increase in the number of iterations due to truncating the deflation subspace. One way to avoid such a case is to perform the truncation strategy after the first restart cycle. An activation flag to initiate the truncation strategy can be set if the residual norm reduction during the first cycle is larger than a predefined threshold that the user has to precise. Our numerical experiments lead to two remarks. The first is that deflating the Ritz vectors related to A is efficient when no deflation subspace exists. The second is that deflating the Ritz vectors related to $A^H A$ needs more initial iterations than the one related to A in order to be efficient. To benefit from both subspaces, we propose a simple criterion based on the reduction of the relative residual norm during the restart cycle to choose the deflation subspace adaptively. In terms of the Algorithm 4, if $\frac{\|r_i\|_2}{\|r_{i-1}\|_2} \leq \varepsilon_0$, where ε_0 is a predefined threshold, then the SVD-based deflation strategy is chosen. Otherwise, the Ritz-based deflation strategy is chosen. In our experiments we set the value of ε_0 to 10^{-1}. Figures 5 and 6 present numerical experiments on the same two previous sequences. Figures 7 and 8 present the history of the residual norm corresponding to each strategy of deflation. As we noted previously, we notice that the decrease of the residual norm starting from the first iteration makes the SVD variant have more advantage against other strategies. The total iteration numbers in the BIGCO24 sequence corresponding to the deflation related to A, A^{-1}, and the alternating $(A, A^H A)$ is 418, 374, and 163, respectively. We also notice that using the adaptive alternating deflation between the Ritz pairs of A and $A^H A$ has the benefit of both strategies, quick useful deflation subspace and quick reduction of residual norm. Moreover, in this example we see how
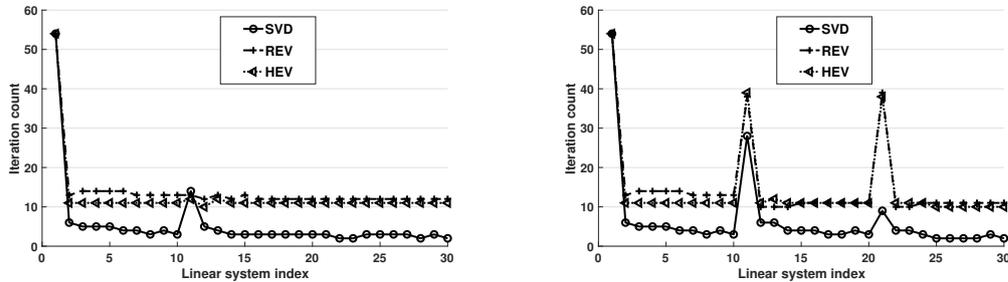
FIGURE 4 – Comparison between three variants of deflation subspaces. SVD, REV, and HEV stand for the deflation methods related to the matrices $A^H A$, A and A^{-1}, respectively. The maximum dimension of the search subspace is 30. The maximum dimension of the deflation subspace is 5. Thirty linear systems in the sequence BIGCO24 are solved, with three matrices, each solved for ten different right-hand sides (for index i the three methods share the same right-hand side). In the figure on the left, the iteration counts are obtained by using a solver that does not truncate the deflation subspace. In the figure on the right, the iteration counts are obtained by using a solver that truncates the deflation subspace. The truncation threshold is set to 10^{-3}.
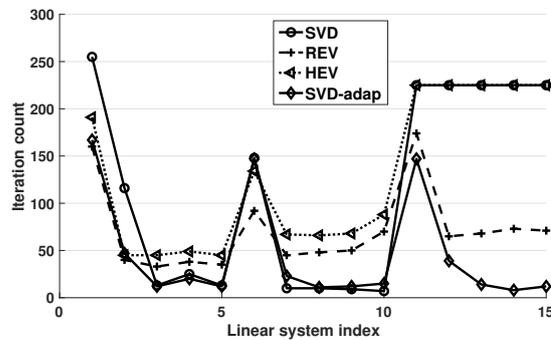

FIGURE 5 – Comparison between four variants of deflation subspaces. SVD, REV, and HEV stand for the deflation method related to the matrices $A^H A$, A, and A^{-1}, respectively. SVD-adap stands for an adaptive strategy that alternates the deflation between SVD and REV based on the relative residual norm during the cycle. Fifteen linear systems in the sequence BIGP1 are solved, with three matrices, each solved for five different right-hand sides (for index i the three methods share the same right-hand side).
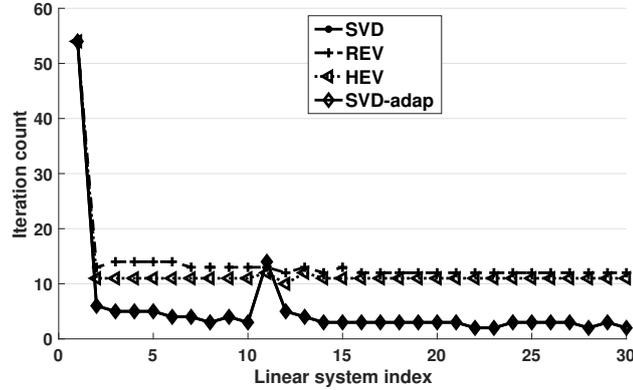
FIGURE 6 – Comparison between four variants of deflation subspaces. SVD, REV, and HEV stand for the deflation method related to the matrices $A^H A$, A, and A^{-1}, respectively. SVD-adap stands for an adaptive strategy that alternates the deflation between SVD and REV based on the relative residual norm during the cycle. The maximum dimension of the search subspace is 30. The maximum dimension of the deflation subspace is 5. Thirty linear systems in the sequence BIGCO24 are solved, with three matrices, each solved for ten different right-hand sides (for index i the three methods share the same right-hand side).

i	$\|V_{svds}(:,i) - QQ^H V_{svds}(:,i)\|_2$	$\|V_{eigs}(:,i) - Q_{HR}Q_{HR}^H V_{eigs}(:,i)\|_2$	$\|V_{eigs}(:,i) - Q_R Q_R^H V_{eigs}(:,i)\|_2$
1	9.8×10^{-2}	7.6×10^{-3}	2.6×10^{-2}
2	2.0×10^{-1}	1.2×10^{-2}	2.3×10^{-2}
3	1.2×10^{-1}	3.8×10^{-2}	5.6×10^{-2}
4	1.5×10^{-1}	9.9×10^{-1}	9.9×10^{-1}
5	2.6×10^{-1}	1.0×10^{-1}	1.2×10^{-1}

TABLE 6 – Approximation measurement of recycling vectors to eigenspaces associated with the operator associated with the recycling strategy. Experiment is run on system with initial BIGCO24. The number of iterations required to reach convergence is 204, 204, and 197 by using the SVD, harmonic Ritz, and Ritz recycling strategy, respectively.

the residual norm stagnates starting from the third matrix systems with the harmonic Ritz-based deflation subspace. The total iteration numbers in the BIGP1 sequence corresponding to the deflation strategies related to A, and the alternating $(A, A^H A)$ is 1062 and 686, respectively.

To measure the effectiveness of approximating the singular vectors and eigenvectors, we perform the following experiment :
— Compute the singular vectors (V_{svds}) of the operator that are associated with the smallest five singular values by using the routine `svds`.
— Solve a linear system by using the SVD recycling strategy with GMRES-MDR(25,5) with convergence threshold 10^{-6}.
— Compute QR decomposition of the recycling subspace basis $\tilde{U}_k = QR,.$
— Measure the error norm $\|v - QQ^H v\|_2$, where v is a column vector of V_{svds}.
We perform the same experiment with the two other variants and measure the error with respect to the eigenvector (V_{eigs}) associated with the smallest five eigenvalues in magnitude of the operator computed by using the routine `eigs`. Table 6 shows the results of this experiment. We observe that low approximation accuracy for each variant is sufficient.
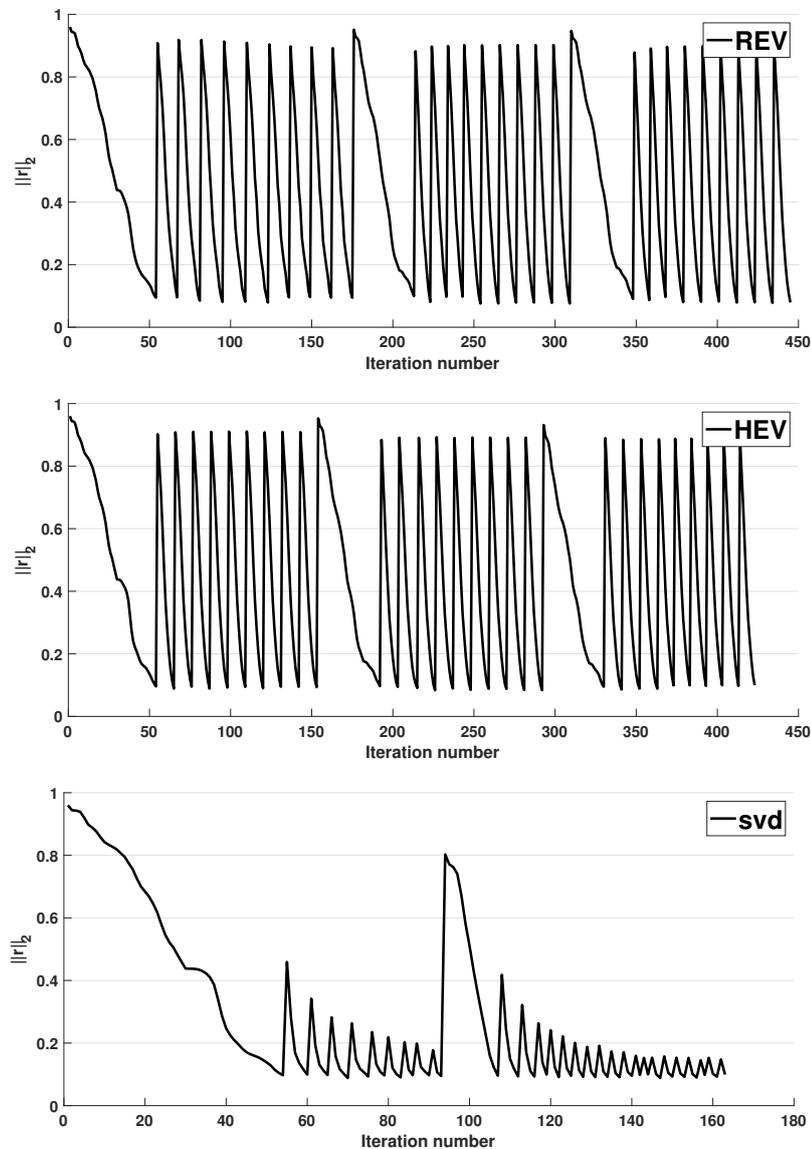
FIGURE 7 – History of residual norm in a sequence of linear systems (a peak corresponds to the beginning of a new linear system). Comparison between three variants of deflation subspaces. SVD, REV, and HEV stand for the deflation method related to the matrices $A^H A$, A, and A^{-1}, respectively. The SVD variant uses an adaptive strategy that alternates the deflation between SVD and REV deflation methods based on the relative residual norm during the cycle. The maximum dimension of the search subspace is 30. The maximum dimension of the deflation subspace is 5. Thirty linear systems in the sequence BIGCO24 are solved, with three matrices, each solved for ten different right-hand sides (for index i the three methods share the same right-hand side).
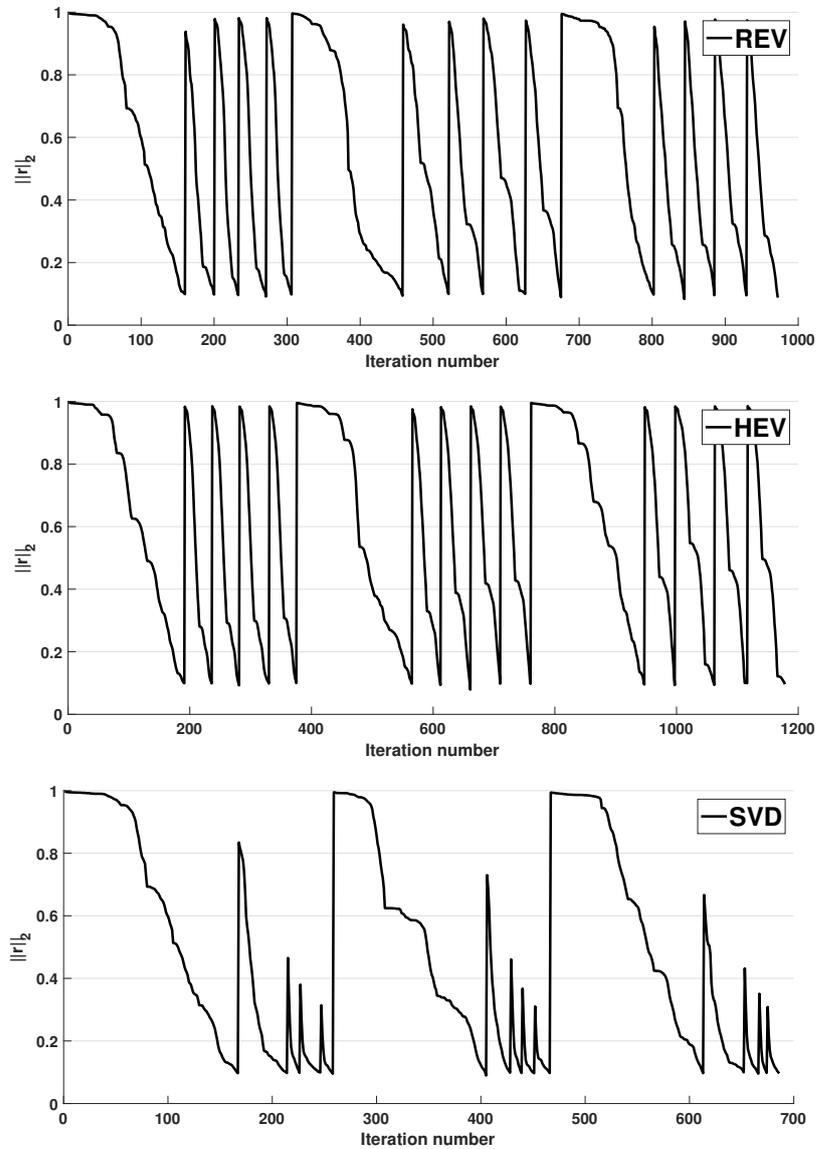
FIGURE 8 – History of residual norm in a sequence of linear systems (a peak corresponds to the beginning of a new linear system). Comparison between three variants of deflation subspaces. SVD, REV, and HEV stand for the deflation method related to the matrices $A^H A$, A, and A^{-1}, respectively. The SVD variant uses an adaptive strategy that alternates the deflation between SVD and REV based on the relative residual norm during the cycle. The maximum dimension of the search subspace is 30. The maximum dimension of the deflation subspace is 5. Fifteen linear systems in the sequence BIGP1 are solved, with three matrices, each solved for five different right-hand sides (for index i the three methods share the same right-hand side).

i	$\|V_{svds}(:,i) - QQ^H V_{svds}(:,i)\|_2$	$\|V_{eigs}(:,i) - Q_{HR}Q_{HR}^H V_{eigs}(:,i)\|_2$	$\|V_{eigs}(:,i) - Q_R Q_R^H V_{eigs}(:,i)\|_2$
1	1.0×10^{-6}	1.2×10^{-2}	2.5×10^{-2}
2	2.6×10^{-1}	3.9×10^{-3}	7.1×10^{-3}
3	8.4×10^{-2}	7.9×10^{-2}	8.7×10^{-2}
4	2.0×10^{-1}	1.0	1.0
5	2.2×10^{-1}	3.3×10^{-2}	4.5×10^{-2}

TABLE 7 – Approximation measurement of recycling vectors to eigenspaces associated with the operator associated with the recycling strategy. Experiment is run on system with initial BIGCO24. The right-hand side is chosen to be the left singular vector associated with the smallest singular value. The number of iterations required to reach convergence is 185, 191, and 180 by using the SVD, harmonic Ritz, and Ritz recycling strategy, respectively.

In Table 7, we perform the same previous experiment where we choose the right-hand side to be the left singular vector associated with the smallest singular value. We note that the SVD-based recycling subspace extracted a good approximation of the right singular vector associated with the smallest singular value.

8.2 Parallel numerical experiments

In this section we present the numerical experiments of our implementation of the method GMRES-MDR. All experiments are performed on the supercomputer PANGEA at TOTAL. We had access to a maximum of 1024 nodes. Each node has 16 cores. Our set of test matrices consists of the matrices SPE10 and Grid2D-2000x2000. We solve two sequences, each of nine linear systems in which the matrix is fixed. The right-hand sides are different for each linear system in the sequence. All variants have the same right-hand side for the ith linear system, $i = 1, \ldots, 9$. The matrix is SPE10 $\in \mathbb{R}^{n \times n}$ where n is the number of unknowns $n = 1094421$. The number of the non-zero elements $nnz = 7478141$. SPE10 corresponds to a simulation of a black oil with a heterogeneous permeability field [4]. The grid has 1094421 active cells. The matrix Grid2D-2000x2000 $\in \mathbb{R}^{n \times n}$, where n is the number of unknowns $n = 4 \times 10^6$. The number of non-zero elements $nnz = 2 \times 10^7$. This matrix corresponds to a discretization of a pressure system in which the permeability field follows a log-normal distribution with standard deviation $\sigma = 3$ and a mean $\mu = log(500)$. The discretized domain is the unit square with 2000×2000 grid points. In all experiments, a linear system is solved to warm up the machine before solving each sequence. Figure 9 shows the scalability of the three variants discussed in Section 5. The sequence of linear systems is related to the test matrix Grid2D-2000x2000. The number of nodes (processors) varies from 16 up to 1024 with 2 cores per processor. A block Jacobi preconditioner with LU factorization in each block is used. The number of blocks is equal to the number of processors. In Listing 1 and Listing 2, we present two post-analysis of GMRES-MDR with SVD deflation variant that is associated with Figure 9. Figure 10 shows the scalability of the three variants discussed in Section 5. The sequence of linear systems is related to the test matrix SPE10. The number of nodes (processors) varies from 16 up to 256, 1 core per MPI processor. A block Jacobi preconditioner with $ILU(0)$ factorization in each block is used. The number of blocks is equal to the number of processors. We notice that the three variants scale well with a smaller runtime for the SVD variant. We note that the runtime for the three variants of deflation does not scale on 512 nodes in the test case SPE10. See Figure 11 and the discussion below for the explanation. Figure 11 shows the percentage of the average time (over all processors) spent in the operations during the solution of a representative linear system of the sequence by using the SVD deflation strategy. This runtime corresponds to a solution of the system on 256 nodes. We
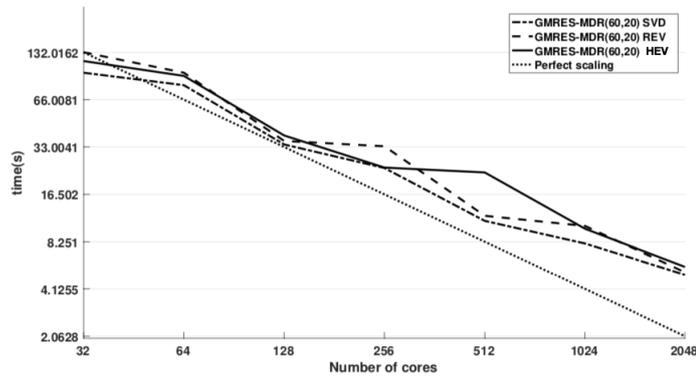
FIGURE 9 – Comparison between runtime of three variants of deflation subspaces. SVD, REV and HEV stand for subspace method related to the matrices $A^H A$, A and A^{-1}, respectively. The maximum dimension of the subspace is 60. The maximum dimension of the deflation subspace is 20. Nine linear systems sharing the same matrix, Grid2D-2000x2000, with different right-hand sides (for index i the three methods share the same right-hand side) are solved.

Listing 1 – Post analysis of GMRES-MDR with SVD deflation variant. First linear system in the sequence Grid2D-2000x2000.

```
Solver information:
Number of processors: 1024
Number of threads: 2
Preconditioner: LU of block Jacobi
Orthogonalization strategy: Double Block Classical Gram–Schmidt
Deflation strategy: SVD values
Max dimension of search subspace: 60
Max dimension of deflation subspace: 20
```

```
Post analysis:
```

```
Number of iterations: 1395
Number of cycles: 36
Residual norm: 9.96438063135672e−02
Relative residual norm: 9.96438063135672e−02
Relative tolerance of convergence: 1.00000000000000e−01
```

Times over processors	:	MAX	MIN	AVERAGE
Total	:	3.237501	3.237501	3.237501 s
Preconditioning	:	1.577664	1.209435	1.294308 s
SPMV	:	0.447073	0.081644	0.166228 s
SPMV Diag	:	0.031328	0.025887	0.026805 s
SPMV Off Diag	:	0.068317	0.016273	0.044505 s
SPMV Comm	:	0.322190	0.036086	0.090636 s
DeflationComputation	:	0.280072	0.275017	0.276312 s
DeflationComm	:	0.021925	0.016195	0.020721 s
Orthogonalization	:	1.539758	1.132159	1.445061 s
Orthogonalization Comm	:	1.094884	0.642479	0.984363 s
HessenbergOperations	:	0.002427	0.002085	0.002221 s

Listing 2 – Post analysis of GMRES-MDR with SVD deflation variant. Ninth linear system in the sequence Grid2D-2000x2000.

```
Solver information:
Number of processors: 1024
Number of threads: 2
Preconditioner: LU of block Jacobi
Orthogonalization strategy: Double Block Classical Gram–Schmidt
Deflation strategy: SVD values
Max dimension of search subspace: 60
Max dimension of deflation subspace: 20
```

```
Post analysis:
```

```
Number of iterations: 39
Number of cycles: 1
Residual norm: 9.99391435927701e−02
Relative residual norm: 9.99391435927701e−02
Relative tolerance of convergence: 1.00000000000000e−01
```

| Times over processors	:	MAX	MIN	AVERAGE	
Total	:	0.069219	0.069219	0.069219	s
Preconditioning	:	0.038246	0.034066	0.036175	s
SPMV	:	0.006581	0.002130	0.004383	s
SPMV Diag	:	0.001015	0.000734	0.000759	s
SPMV Off Diag	:	0.001926	0.000461	0.001252	s
SPMV Comm	:	0.004382	0.000921	0.002392	s
DeflationComputation	:	0.003500	0.003277	0.003379	s
DeflationComm	:	0.000360	0.000173	0.000268	s
Orthogonalization	:	0.025600	0.020788	0.023068	s
Orthogonalization Comm	:	0.013947	0.008824	0.011384	s
HessenbergOperations	:	0.000074	0.000053	0.000063	s
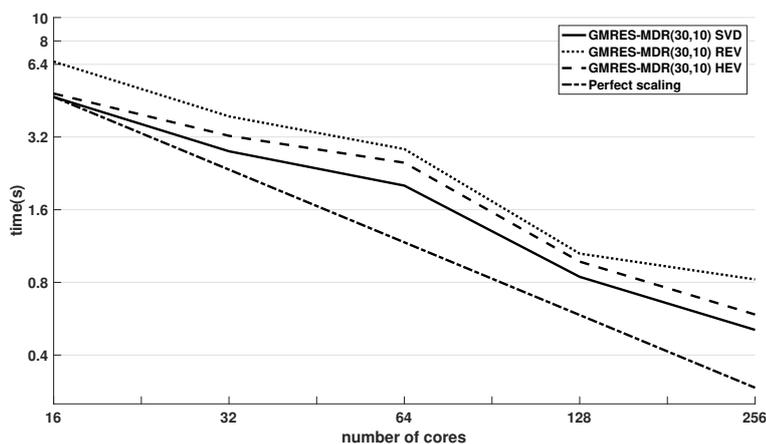
FIGURE 10 – Comparison between runtime of three variants of deflation subspaces. SVD, REV and HEV stand for subspace method related to the matrices $A^H A$, A and A^{-1}, respectively. The maximum dimension of the subspace is 30. The maximum dimension of the deflation subspace is 10. Nine linear systems sharing the same matrix, SPE10, with different right-hand sides (for index i the three methods share the same right-hand side) are solved.
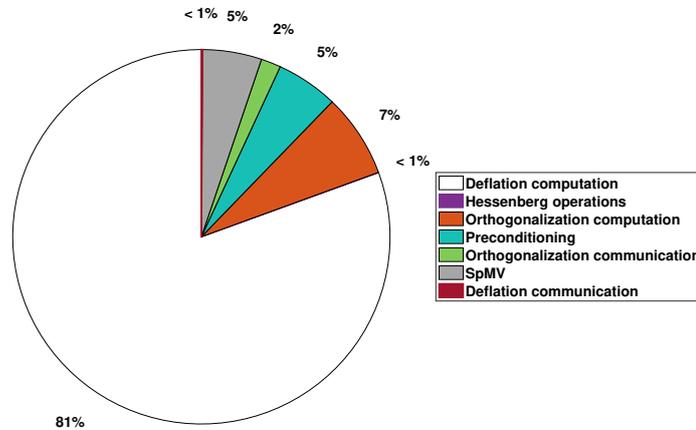
FIGURE 11 – Average time (over processors) of different operations in GMRES-MDR as a percentage of the total runtime. The communication and the computation part of the deflation and the orthogonalization are separated. This runtime corresponds to the solution of a representative linear system during the sequence. Number of nodes = 256. Deflation strategy is the SVD.

remark that the time spent in the computation of the generalized eigenvalue problem becomes much more important than all other operations. This explains why the scaling of the runtime stopped for this sequence after 256 nodes.

9 Conclusion

In this work, we reviewed deflation strategies in Krylov iterative methods. Usual choice of deflation subspaces is related to the approximation of eigenvectors. Based on the study in [36] we introduced the deflation of singular vectors. Since we are interested in solving a sequence of linear systems, we reviewed the method GCRO-DR [28] with an abstract deflation subspace. We briefly presented two deflation subspaces which can be found in the literature. We extended part of the theory presented in [36] on the relation between the residual and the first vector in the Krylov basis sequence to GCRO-DR. Afterwards, we proposed a deflation subspace related to an approximation of singular vectors. To validate the proposed strategy, a comparison between the three variants was made on different sequences arising from reservoir simulation. This validation was performed on a sequential code as well as on a parallel code. The numerical experiments demonstrated the impact of choosing the SVD variant. The gain in the global iteration count is up to 40%. Concerning the parallel implementation, the three variants showed a similar behavior on a short sequence of linear systems. More experiments on larger test cases and longer sequences will be considered in future work.

Références

[1] Kapil. Ahuja, Peter. Benner, Eric. de Sturler, and Lihong. Feng. Recycling bicgstab with an application to parametric model order reduction. *SIAM Journal on Scientific Computing*, 37(5) :S429–S446, 2015.

[2] Kevin. Carlberg, Virginia. Forstall, and Ray. Tuminaro. Krylov-subspace recycling via the pod-augmented conjugate-gradient method. *SIAM Journal on Matrix Analysis and Applications*, 37(3) :1304–1336, 2016.

[3] L. M. Carvalho, S. Gratton, R. Lago, and X. Vasseur. A flexible generalized conjugate residual method with inner orthogonalization and deflated restarting. *SIAM Journal on Matrix Analysis and Applications*, 32(4) :1212–1235, 2011.

[4] M. A. Christie and M. J. Blunt. Tenth spe comparative solution project : A comparison of upscaling techniques. 2001.

[5] E. de Sturler. Nested krylov methods based on gcr. *Journal of Computational and Applied Mathematics*, 67(1) :15–41, 1996.

[6] E. De Sturler. Truncation strategies for optimal krylov subspace methods. *SIAM Journal on Numerical Analysis*, 36(3) :864–889, 1999.

[7] Jack Dongarra, Victor Eijkhout, and Ajay Kalhan. Reverse communication interface for linear algebra templates for iterative methods, 1995.

[8] S. Eisenstat, H. Elman, and M. Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM Journal on Numerical Analysis*, 20(2) :345–357, 1983.

[9] Jocelyne Erhel, Kevin Burrage, and Bert Pohl. Restarted GMRES preconditioned by deflation. *Journal of Computational and Applied Mathematics*, 69(2) :303–318, 1996.

[10] Yogi A. Erlangga and Reinhard. Nabben. Deflation and balancing preconditioners for krylov subspace methods applied to nonsymmetric matrices. *SIAM Journal on Matrix Analysis and Applications*, 30(2) :684–699, 2008.

[11] J. Frank and C. Vuik. On the construction of deflation-based preconditioners. *SIAM Journal on Scientific Computing*, 23(2) :442–462, 2001.

[12] André. Gaul, Martin H. Gutknecht, Jörg. Liesen, and Reinhard. Nabben. A framework for deflated and augmented krylov subspace methods. *SIAM Journal on Matrix Analysis and Applications*, 34(2) :495–518, 2013.

[13] L. Giraud, J. Langou, and M. Rozloznik. The loss of orthogonality in the gram-schmidt orthogonalization process. *Computers & Mathematics with Applications*, 50(7) :1069–1075, 2005. Numerical Methods and Computational Mechanics.

[14] Luc Giraud, Julien Langou, Miroslav Rozlozník, and Jasper van den Eshof. Rounding error analysis of the classical gram-schmidt orthogonalization process. *Numerische Mathematik*, 101(1) :87–100, 2005.

[15] Gene H. Golub and Charles F. Van Loan. *Matrix Computations (3rd Ed.)*. Johns Hopkins University Press, Baltimore, MD, USA, 1996.

[16] P. Gosselet, C. Rey, and J. Pebrel. Total and selective reuse of krylov subspaces for the resolution of sequences of nonlinear structural problems. *International Journal for Numerical Methods in Engineering*, 94(1) :60–83, 2013.

[17] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards.*, 49 :409–436, 1952.

[18] Intel(R). Math kernel library.

[19] Pierre Jolivet and Pierre-Henri Tournier. Block iterative methods and recycling for improved scalability of linear solvers. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '16. IEEE Press, 2016.

[20] J. Langou. *Iterative methods for solving linear systems with multiple right-hand sides*. PhD thesis, CERFACS, 2003.

[21] J. C. Meza. A modification to the GMRES method for ill-conditioned linear systems, 1995.

[22] J. C. Meza and W. W. Symes. Deflated krylov subspace methods for nearly singular linear systems. *Journal of Optimization Theory and Applications*, 72(3) :441–457, 1992.

[23] Ronald B. Morgan. A restarted GMRES method augmented with eigenvectors. *SIAM Journal on Matrix Analysis and Applications*, 16(4) :1154–1171, 1995.

[24] Ronald B. Morgan. GMRES with deflated restarting. *SIAM Journal on Scientific Computing*, 24(1) :20–37, 2002.

[25] R. A. Nicolaides. Deflation of conjugate gradients with applications to boundary value problems. *SIAM J. Numer. Anal.*, 24(2) :355–365, April 1987.

[26] Dianne P. O'Leary. The block conjugate gradient algorithm and related methods. *Linear Algebra and its Applications*, 29 :293–322, 1980.

[27] Chris C. Paige, Beresford N. Parlett, and Henk A. van der Vorst. Approximate solutions and eigenvalue bounds from krylov subspaces. *Numerical Linear Algebra with Applications*, 2(2) :115–133, 1995.

[28] Michael L. Parks, Eric de Sturler, Greg Mackey, Duane D. Johnson, and Spandan Maiti. Recycling krylov subspaces for sequences of linear systems. *SIAM Journal on Scientific Computing*, 28(5) :1651–1674, 2006.

[29] Michael Lawrence Parks. *The Iterative Solution of a Sequence of Linear Systems Arising from Nonlinear Finite Element Analysis*. PhD thesis, 2005.

[30] Christian Rey and Franck Risler. A rayleigh–ritz preconditioner for the iterative solution to large scale nonlinear problems. *Numerical Algorithms*, 17(3) :279–311, Jul 1998.

[31] Y. Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd edition, 2003.

[32] Y. Saad, M. Yeung, J. Erhel, and F. Guyomarc'h. A deflated version of the conjugate gradient algorithm. *SIAM Journal on Scientific Computing*, 21(5) :1909–1926, 2000.

[33] Youcef Saad and Martin H. Schultz. GMRES : A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3) :856–869, 1986.

[34] G. Sacchi and V. Simoncini. A GMRES convergence analysis for localized invariant subspace ill-conditioning. *SIAM Journal on Matrix Analysis and Applications*, 40(2) :542–563, 2019.

[35] H. Cao SPE Schlumberger, H.A. Tchelepi SPE Stanford U., J. Wallis Wallis Consulting Inc., and H. Yardumian SPE Chevron. Parallel scalable unstructured CPR-Type Linear Solver for Resevoir Simulation. *SPE 96809*, 2005.

[36] V. Simoncini. On the convergence of restarted krylov subspace methods. *SIAM Journal on Matrix Analysis and Applications*, 22(2) :430–452, 2000.

[37] Marc Snir, Steve Otto, Steven Huss-Lederman, David Walker, and Jack Dongarra. *MPI-The Complete Reference, Volume 1 : The MPI Core*. MIT Press, Cambridge, MA, USA, 2nd. (revised) edition, 1998.

[38] Kirk M. Soodhalter, Eric de Sturler, and Misha Kilmer. A survey of subspace recycling iterative methods, 2020.

[39] Kirk M. Soodhalter, Daniel B. Szyld, and Fei Xue. Krylov subspace recycling for sequences of shifted linear systems. *Applied Numerical Mathematics*, 81 :105–118, 2014.

[40] J. M. Tang, R. Nabben, C. Vuik, and Y. A. Erlangga. Comparison of two-level preconditioners derived from deflation, domain decomposition and multigrid methods. *Journal of Scientific Computing*, 39(3) :340–370, 2009.

[41] J.R. Wallis, R.P. Kendall, and T.E. Little. Constrained residual acceleration of conjugate residual methods. *SPE 13563*, 1985.