



HAL
open science

Randomized Flexible GMRES with Deflated Restarting

Yongseok Jang, Laura Grigori, Emeric Martin, Cédric Content

► **To cite this version:**

Yongseok Jang, Laura Grigori, Emeric Martin, Cédric Content. Randomized Flexible GMRES with Deflated Restarting. 2023. hal-04072873

HAL Id: hal-04072873

<https://hal.science/hal-04072873>

Preprint submitted on 18 Apr 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Randomized Flexible GMRES with Deflated Restarting

Yongseok Jang^{a,*}, Laura Grigori^b, Emeric Martin^a, Cédric Content^a

^aDAAA, ONERA, The French Aerospace Lab, Université Paris Saclay, Châtillon, France

^bAlpines, Inria, Sorbonne Université, CNRS Laboratoire Jacques-Louis Lions, Paris, France

Abstract

For high dimensional spaces, a *randomized Gram-Schmidt* (RGS) algorithm is beneficial in computational costs as well as numerical stability. We apply this dimension reduction technique by random sketching to Krylov subspace methods, e.g. to the *generalized minimal residual method* (GMRES). We propose a flexible variant of GMRES with the randomized Gram-Schmidt based Arnoldi iteration to produce a set of basis vectors of the Krylov subspace. Even though the Krylov basis is no longer l_2 orthonormal, its random projection onto the low dimensional space shows l_2 orthogonality. As a result, it is observed the numerical stability which turns out to be independent of the dimension of the problem even in extreme scale problems. On the other hand, as the Harmonic Ritz values are commonly used in GMRES with deflated restarting to improve convergence, we consider another deflation strategy, for instance disregarding the singular vectors associated with the smallest singular values. We thus introduce a new algorithm of the randomized flexible GMRES with *singular value decomposition* (SVD) based deflated restarting. At the end, we carry out numerical experiments in the context of compressible turbulent flow simulations. Our proposed approach exhibits a quite competitive numerical behaviour to existing methods while reducing computational costs.

Keywords: Randomized Gram-Schmidt orthogonalization algorithm, Random sketching, Krylov subspace methods, Implicit restarting with deflation

2020 MSC: 15A06, 65F10, 65F15, 65N22, 68W20

1. Introduction

The Krylov subspace methods are commonly used to solve large, sparse linear systems using iterative methods. In particular, Saad and Schultz [1] proposed the *generalized minimal residual method* (GMRES) for solving non-symmetric problems, resulting in approximate solutions within Krylov subspaces with minimal residual norms. This method is an improvement and generalization of the *minimal residual method* (MINRES) [2]. It is often observed that the convergence of Krylov subspace methods depends on the distribution of eigenvalues, the field of values, or the pseudospectra with respect to the normality of a matrix [3, 4]. To address this, Morgan [5] introduced GMRES with *deflated restarting* (GMRES-DR), which disregards or deflates eigenvectors associated with the smallest eigenvalues in magnitude. For more flexible variants and improvements of GMRES, see Giraud and Rollin [6, 7]. Simoncini's work [8], which explores the impact of small singular values close to zero on the convergence of restarted Krylov subspace methods, and Park et al.'s work [9], specifically the *generalized conjugate residual with inner orthogonalization method with deflated restarting* (GCRO-DR), inspired Al Daas et al. [10] to develop GMRES with *modified DR* (GMRES-MDR) using a *singular value decomposition* (SVD) based deflation technique.

As a variant of the restarted GMRES with recycling subspaces, GCRO-DR was developed to solve sequences of linear systems [9]. It was noted that both GMRES-DR and GCRO-DR produce the same iterations in exact arithmetic, making them algebraically equivalent. Carvalho et al. [11] presented a comparison

*Corresponding author.

Email addresses: yongseok.jang@onera.fr (Yongseok Jang), laura.grigori@inria.fr (Laura Grigori), emeric.martin@onera.fr (Emeric Martin), cedric.content@onera.fr (Cédric Content)

of FGMRES-DR and FGCRO-DR in a flexible manner, for example, regarding variable preconditioning. It was demonstrated that both methods are algebraically equivalent if a collinearity condition is satisfied, but FGCRO-DR is more beneficial when solving sequences of linear systems.

For practical applications of Krylov subspace methods, it is essential to construct Krylov basis vectors. For instance, the *Arnoldi* process generates a sequence of orthonormal vectors spanning the Krylov subspace through the use of *Gram-Schmidt* (GS) process. Since GMRES is based on the Arnoldi iterations, it is crucial to use a computationally stable GS process to ensure the quality of the Krylov basis. Balabanov and Grigori [12] presented the *randomized Gram-Schmidt* (RGS) process, which uses dimension reduction technique through random sketching, as a means of improving both numerical stability and computational efficiency. This process has been applied to the development of randomized GMRES method. Recently, Balabanov and Grigori [13] introduced a block version of the RGS algorithm, which allows for unit round-off independence from the dominant dimension of the matrix and offers an advantage for solving large-scale problems on low-precision arithmetic architectures. In addition, Balabanov and Grigori presented the randomized *Rayleigh-Ritz* method which utilizes the improvement of the sketched Galerkin projection to solve eigenvalue problems, along with a theoretical analysis of the method. Nakatsukasa and Tropp [14] developed another variant of fast randomized algorithms for GMRES and Rayleigh-Ritz methods, called *sGMRES* and *sRR*, respectively. While the works of Balabanov and Grigori are based on the full Arnoldi process, sGMRES and sRR utilize a truncated version of the Arnoldi process, which results in asymptotically faster computational costs but less stability in the Krylov basis vectors.

In this paper, we propose new variants of randomized GMRES algorithms. More specifically, to support variable preconditioning, we develop a flexible version of the Arnoldi algorithm for constructing a set of Krylov basis vectors in Krylov subspace methods, such as GMRES. By incorporating randomization, our algorithm achieves higher efficiency, improved stability, and reduced computational complexity. Accordingly, we introduce the *randomized flexible GMRES with restart* (RGS-FGMRES) and apply Morgan’s deflation strategy [5], which includes the deflation of harmonic eigenvectors, resulting in the *randomized flexible GMRES with deflated restart* (RGS-FGMRES-DR). Furthermore, motivated by singular vectors deflation techniques [10], we introduce the *randomized flexible GMRES with modified deflated restart* (RGS-FGMRES-MDR), which relies on the GCR method with inner sketched orthogonalization for SVD-based deflation. Whereas GCRO-DR is based on l_2 orthogonal projections, our method consider sketched orthogonal projections. For each of the proposed variants, we present a simplified form of minimizing the residual in the sketched manner.

We would like to highlight that our research is the first to use random sketching for dimension reduction in flexible GMRES. This includes both the well-known deflation of harmonic Ritz vectors and the recent deflation method using singular vectors. Our proposed algorithms show improved stability and convergence, even with reduced computational costs. Numerical examples support the benefits of our developments in solving ill-conditioned non-symmetric linear systems arising from CFD compressible turbulent flow simulations.

This paper is organized as follows: in Section 1, we cover preliminary information. Section 2 presents the basic concept of randomization in GMRES, as well as the algorithms for RGS flexible Arnoldi process and RGS-FGMRES-DR, which is based on harmonic Ritz pairs. Section 3 combines it with approximate singular vectors based deflation in RGS-FGMRES-MDR. To highlight the benefits of our method, we describe numerical stability and computational costs in Section 4. In Section 5, we provide numerical examples to validate our proposed algorithms. Finally, we conclude with remarks in Section 6.

1.1. Preliminary

We use standard notations for the sake of generality. We consider complex valued systems and denote a vector (*resp.* a matrix) by a bold lower letter (*resp.* an upper letter), e.g. we denote a n length complex vector by $\mathbf{x} \in \mathbb{C}^n$ for $n \in \mathbb{N}$. Let $A \in \mathbb{C}^{n \times n}$ be a non-singular complex matrix and $\mathbf{b} \in \mathbb{C}^n$ be a vector. We denote the number of nonzero in A by $nnz(A)$. We want to solve the linear system $A\mathbf{x} = \mathbf{b}$ by Krylov subspace methods. For any vector \mathbf{x} and matrix X , we denote $\|\mathbf{x}\|$ the l_2 norm of \mathbf{x} with l_2 inner product (\cdot, \cdot) , X^T the transpose of X and X^H the Hermitian transpose of X . A pseudo-inverse of X is denoted by X^\dagger . $\kappa(X)$ refers to the condition number of X . The identity matrix of the dimension n and the zero rectangular matrix of i by j are denoted by I_n and $O_{i \times j}$, respectively. For the readability of algorithms, we

adopt MATLAB expression, for example, $X(1:i, 1:j)$ denotes the submatrix consisting of first i rows and first j columns of X .

Gram-Schmidt process is most commonly used for orthonormalizing a set of vectors or computing QR factorization of a matrix. It proceeds iteratively based on the projection operators. For instance, let linearly independent $V_m = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m]$ and the projection operator $P_{\mathbf{u}}$ defined by

$$P_{\mathbf{u}}(\mathbf{v}) = \frac{(\mathbf{u}, \mathbf{v})}{(\mathbf{u}, \mathbf{u})} \mathbf{u} \quad \text{for any } \mathbf{u} \text{ and } \mathbf{v}.$$

Then Gram-Schmidt process leads a set of orthogonal vectors $U_m = [\mathbf{u}_1, \dots, \mathbf{u}_m]$ where

$$\mathbf{u}_1 = \mathbf{v}_1 \quad \text{and} \quad \mathbf{u}_i = \mathbf{v}_i - \sum_{j=1}^{i-1} P_{\mathbf{u}_j}(\mathbf{v}_i) \quad \text{for } i = 2, \dots, m.$$

Hence it can also result in an orthonormal set by normalizing each vector. For instance, we can obtain the orthonormal Q_m such that

$$Q_m = [\mathbf{q}_1, \dots, \mathbf{q}_m] \quad \text{where} \quad \mathbf{q}_i = \mathbf{u}_i / \|\mathbf{u}_i\| \quad \forall i = 1, \dots, m.$$

In practice, several variants of Gram-Schmidt process have been proposed and analyzed, for example the *classical Gram-Schmidt* algorithm (CGS) [15], the *modified Gram-Schmidt* algorithm (MGS) [15, 16] and re-orthogonalized algorithms (namely CGS2 and MGS2, respectively) [17]. The latter methods improve computational orthogonality for practical implementation. Depending on the choice of Gram-Schmidt algorithms, the projection can be defined with respect to iterations by

$$\begin{aligned} \text{CGS:} \quad & P^{(i)} = I_n - Q_i Q_i^H, \\ \text{CGS2:} \quad & P^{(i)} = (I_n - Q_i Q_i^H)(I_n - Q_i Q_i^H), \\ \text{MGS:} \quad & P^{(i)} = (I_n - \mathbf{q}_i \mathbf{q}_i^H)(I_n - \mathbf{q}_{i-1} \mathbf{q}_{i-1}^H) \cdots (I_n - \mathbf{q}_1 \mathbf{q}_1^H), \\ \text{MGS2:} \quad & P^{(i)} = (I_n - \mathbf{q}_i \mathbf{q}_i^H) \cdots (I_n - \mathbf{q}_1 \mathbf{q}_1^H)(I_n - \mathbf{q}_i \mathbf{q}_i^H) \cdots (I_n - \mathbf{q}_1 \mathbf{q}_1^H), \end{aligned}$$

where $Q_i = [\mathbf{q}_1, \dots, \mathbf{q}_i]$ for each $i \in \{1, \dots, m\}$. These projectors approximate the l_2 orthogonal projector $I_n - Q_i Q_i^\dagger$ onto the orthogonal complement of $\text{span}(Q_i)$. The quality of computational orthogonality depends on the condition number of Q_m .

On the other hand, the RGS algorithm [12] is based on random sketching which is a technique of dimension reduction. To be specific, it is motivated by approximating inner products of high dimension vectors with inner products of their low dimensional random projections, so called sketch. Thus, the approximate orthogonality depends on the random projection rather than the l_2 norm sense. For more theoretical proofs and details of RGS, we refer to [12] and the references therein.

Arnoldi iteration is required to construct an orthonormal basis of Krylov subspace in order to approximate eigenpairs and approximate solutions of linear systems. For instance, let us consider $A \in \mathbb{C}^{n \times n}$ and $\mathbf{b} \in \mathbb{C}^n$, then the Krylov subspace $\mathcal{K}_m(A, \mathbf{b})$ is defined by

$$\mathcal{K}_m(A, \mathbf{b}) = \text{span}\{\mathbf{b}, A\mathbf{b}, A^2\mathbf{b}, \dots, A^{m-1}\mathbf{b}\}.$$

Then its orthonormal basis can be computed by Gram-Schmidt algorithms to give

$$AV_m = V_{m+1}H_m, \tag{1.1}$$

called *Arnoldi identity* with l_2 orthogonality of V_{m+1} such as $V_{m+1}^H V_{m+1} = I_{m+1}$ where $H_m \in \mathbb{C}^{(m+1) \times m}$ is the upper Hessenberg matrix. In this manner, we have also a flexible Arnoldi relation such that

$$AZ_m = V_{m+1}H_m, \tag{1.2}$$

where $Z_m = [\mathbf{z}_1, \dots, \mathbf{z}_m]$ is defined with variable preconditioning, for instance $\mathbf{z}_j = M_j^{-1}\mathbf{v}_j$ with M_j is the preconditioner for the j -th Arnoldi iteration for $j = 1, \dots, m$.

2. Randomized GMRES

Random sketching is a technique used in numerical linear algebra to reduce the dimensionality of large data sets. The basic idea is to represent a high-dimensional matrix or vector by a much smaller, randomized low-rank approximation. This is achieved by multiplying the original matrix or vector with a random sketching matrix. Random sketching can be used to speed up computations and reduce the memory requirements of algorithms that operate on large matrices and vectors. Defining a suitable random sketching matrix is an essential aspect of this technique.

In practice, several variants of random sketching have been proposed and analyzed, such as Gaussian distributions, Rademacher distributions, and sub-sampled randomized Hadamard transform (SRHT). The quality of the approximation depends on the choice of sketching matrix. For more details on theoretical estimations and specific examples of suitable sketching matrices, we refer to literature such as [18, 19, 20].

Additionally, it is important to note that in order to ensure the effectiveness of the dimensionality reduction, the sketching matrix should have certain properties such as ϵ -embedding and (ϵ, δ, d) oblivious l_2 -subspace embedding. The size of the sketching matrix should also be chosen accordingly. For more details on these properties and specific examples of suitable sketching matrix sizes, we refer to literature such as [12, 21].

Let $\Theta \in \mathbb{C}^{t \times n}$ be a sketching matrix with $t \ll n$. The sketching matrix leads to the sketched product and its induced norm, called Θ -norm or *sketched norm*, to approximate l_2 inner product and l_2 norm as embedding of subspaces. For instance, we can define the random sketched product and the sketched norm by

$$(\mathbf{v}, \mathbf{w})_{\Theta} = (\Theta \mathbf{v}, \Theta \mathbf{w}) \quad \text{and} \quad \|\mathbf{v}\|_{\Theta} = \|\Theta \mathbf{v}\|,$$

for any $\mathbf{v}, \mathbf{w} \in \mathbb{C}^n$, respectively. Then, there exists a l_2 -subspace embedding of subspaces of \mathbb{C}^n into subspaces of \mathbb{C}^t . The quality of embedding property will follow the sketching matrix Θ . For more details of theoretical estimations, we refer to [12]. Thus, we need to define an appropriate sketching matrix such as Gaussian distributions, Rademacher distributions, SRHT, etc., e.g. see [18, 19, 20].

Definition 1 (ϵ -embedding). *For $0 < \epsilon < 1$, the sketching matrix Θ is an ϵ -embedding for subspace $V \subset \mathbb{C}^n$ if*

$$\forall \mathbf{x}, \mathbf{y} \in V, \quad |(\mathbf{x}, \mathbf{y}) - (\mathbf{x}, \mathbf{y})_{\Theta}| \leq \epsilon \|\mathbf{x}\| \|\mathbf{y}\|.$$

Definition 2 ((ϵ, δ, d) oblivious l_2 -subspace embedding). *The random sketching matrix Θ is a (ϵ, δ, d) oblivious l_2 -subspace embedding for $V \subset \mathbb{C}^n$ of dimension d if*

$$\text{probability}(\Theta \text{ is an } \epsilon \text{ embedding for } V) \geq 1 - \delta.$$

In this paper, we refer to [12] for a reasonable choice of random sketching matrices with certain quality of l_2 -subspaces embedding. For example, our Θ is given by either rescaled Rademacher distribution or partial SRHT (P-SRHT). Furthermore, we define the size of sketching matrix for (ϵ, δ, d) oblivious l_2 -subspace embeddings by

$$t \geq 7.87\epsilon^{-2}(6.9d + \log(1/\delta)) \quad \text{and} \quad t \geq 2(\epsilon^2 - \epsilon^3/3)^{-1} \left(\sqrt{d} + \sqrt{8 \log(6n/\delta)} \right)^2 \log(3d/\delta)$$

for rescaled Rademacher distribution and P-SRHT, respectively [21].

By the manner of sketching, we mainly concern Θ *orthogonality* to construct a set of basis of Krylov subspace. To be specific, we present applications of randomized Gram-Schmidt with the sketching matrix Θ , e.g. randomized flexible Arnoldi process and randomized FGMRES based on the sketched orthonormality in the following section. Hereafter, we assume Θ is l_2 -subspace embedding for Krylov subspace.

Remark (l_2 norm and Θ norm). *For a vector $\mathbf{v} \in \mathbb{C}^n$, we have the following bounds.*

- $\|\mathbf{v}\|_{\Theta} = \|\Theta \mathbf{v}\| \leq \|\Theta\| \|\mathbf{v}\|.$
- $\|\mathbf{v}\| = \|\Theta^{\dagger} \Theta \mathbf{v}\| \leq \|\Theta^{\dagger}\| \|\mathbf{v}\|_{\Theta}.$

- $|\|\mathbf{v}\| - \|\mathbf{v}\|_{\Theta}| \leq \sqrt{\epsilon}\|\mathbf{v}\|$ if Θ is an ϵ -embedding for $V \subset \mathbb{C}^n$ and $\mathbf{v} \in V$.
- $\|\Theta\|_F \leq \sqrt{(1+\epsilon)n}$ with probability at least $1-\delta$ if Θ is an $(\epsilon, \delta/n, 1)$ oblivious l_2 -subspace embedding, where $\|\cdot\|_F$ is the Frobenius norm.

We refer to [12] for more details of third and last bounds. First and second inequalities lead the norm equivalence. To be specific, we have

$$\frac{1}{\|\Theta^\dagger\|} \|\mathbf{v}\| \leq \|\mathbf{v}\|_{\Theta} \leq \|\Theta\| \|\mathbf{v}\|,$$

for any \mathbf{v} . For sufficiently good Θ , e.g. full rank Θ , it implies $\|\Theta^\dagger\| = 1/\sigma_{\min}(\Theta)$ and

$$\sigma_{\min}(\Theta) \|\mathbf{v}\| \leq \|\mathbf{v}\|_{\Theta} \leq \|\Theta\| \|\mathbf{v}\|,$$

where $\sigma_{\min}(\Theta)$ is the minimum singular value of Θ . Since Frobenius norm of a matrix is equivalent to l_2 norm of its singular values, we can observe the bound of singular values by the last bound. Note that the norm equivalence is independent of \mathbf{v} but it depends on the parameters of subspace embedding.

2.1. Randomized Arnoldi iteration

Unlike other variants of Gram-Schmidt process, e.g. CGS, MGS, CGS2 and MGS2, the randomized Gram-Schmidt process allows us to have Θ orthogonality rather than l_2 orthogonality. In other words, we will employ Θ orthogonal projector hence we define

$$\text{RGS: } P^{(i)} = I_n - Q_i(\Theta Q_i)^\dagger \Theta,$$

where $Q_i = [\mathbf{q}_1, \dots, \mathbf{q}_i]$ for each $i \in \{1, \dots, m\}$.

In a similar way with the classical Arnoldi iteration, the RGS variant also holds Arnoldi relation (1.1) (or (1.2) if preconditioned). The algorithm of randomized flexible Arnoldi process is depicted as follows.

Algorithm 1: RGS flexible Arnoldi iteration

Data: matrix $A \in \mathbb{C}^{n \times n}$, vector $\mathbf{v} \in \mathbb{C}^n$, sketching matrix $\Theta \in \mathbb{C}^{t \times n}$, number of iterations m , and variable preconditioner M_j for $j = 1, \dots, m$

Result: $V_{m+1} \in \mathbb{C}^{n \times (m+1)}$, $Z_m \in \mathbb{C}^{n \times m}$ and Hessenberg matrix $H_m \in \mathbb{C}^{(m+1) \times m}$ (optionally $S_{m+1} \in \mathbb{C}^{t \times (m+1)}$)

```

1  $\mathbf{w} = \mathbf{v}$ ;
2 Sketch  $\mathbf{w}$ :  $\mathbf{p} = \Theta \mathbf{w}$ ;
3  $\mathbf{q} = \mathbf{w}$ ;  $\mathbf{s} = \mathbf{p}$ ;  $h = \|\mathbf{s}\|$ ;  $\mathbf{s}_1 = \mathbf{s}/h$ ;  $\mathbf{v}_1 = \mathbf{q}/h$ ;  $S_1 = [\mathbf{s}_1]$ ;  $V_1 = [\mathbf{v}_1]$ ;
4 for  $j = 1 : m$  do
5   Apply preconditioner  $M_j$  onto  $\mathbf{v}_j$ :  $\mathbf{z}_j = \text{apply}(M_j, \mathbf{v}_j)$ ;
6    $\mathbf{w} = A \mathbf{z}_j$ ;
7   Sketch  $\mathbf{w}$ :  $\mathbf{p} = \Theta \mathbf{w}$ ;
8   Solve  $t \times j$  least squares problem:  $H_m(1 : j, j) = \arg \min_{\mathbf{y} \in \mathbb{C}^j} \|S_j \mathbf{y} - \mathbf{p}\|$ ;
9    $\mathbf{q} = \mathbf{w} - V_j H_m(1 : j, j)$ ;
10  Sketch  $\mathbf{q}$ :  $\mathbf{s} = \Theta \mathbf{q}$ ;
11   $h = \|\mathbf{s}\|$ ;  $\mathbf{s}_{j+1} = \mathbf{s}/h$ ;  $\mathbf{v}_{j+1} = \mathbf{q}/h$ ;
12   $S_{j+1} = [\mathbf{s}_1, \dots, \mathbf{s}_{j+1}]$ ;  $V_{j+1} = [\mathbf{v}_1, \dots, \mathbf{v}_{j+1}]$ ;  $Z_j = [\mathbf{z}_1, \dots, \mathbf{z}_j]$ ;  $H_m(j+1, j) = h$ ;
13 end
14 (Optional: compute  $\Delta_m := \|S_{m+1}^H S_{m+1} - I_{m+1}\|_F$  to check the sketched orthogonality numerically.)

```

The basic concept of Algorithm 1 is sketching the basis onto the subspace of dimension t from the higher dimension n and solving least squares problems to impose orthogonality with respect to the sketching

matrix Θ . Theoretically, S_j is l_2 orthonormal for a full rank Θ but in practice with including round-off errors, ϵ -embedding property equipped sketching matrix will yield $S_j^H S_j \approx I_j$. Hence we may want to have the sketching matrix satisfying sufficiently small $\Delta_m = \|S_{m+1}^H S_{m+1} - I_{m+1}\|_F$. For more detail of stability analysis, we refer to [12]. Hereafter, we assume that the sketching matrix Θ has sufficiently good l_2 -subspace embedding properties, which resulting $\kappa(Q) = 1 + O(\epsilon)$ where Q is obtained by RGS process.

By disregarding the rounding error, we can derive l_2 orthonormality of S_{m+1} such as $S_{m+1}^H S_{m+1} = I_{m+1}$ as well as the Θ orthonormality of V_{m+1} such as $V_{m+1}^H \Theta^H \Theta V_{m+1} = I_{m+1}$, since $S_{m+1} = \Theta V_{m+1}$. At the same time, the Arnoldi identity holds.

Remark. As in Line 8 of Algorithm 1, we need to solve the least squares problem. It is equivalent to computing $(\Theta V_j)^\dagger \Theta \mathbf{w}$ in the projection $P^{(j)}$ of \mathbf{w} , since we have

$$\arg \min_{y \in \mathbb{C}^j} \|S_j \mathbf{y} - \mathbf{p}\| = (S_j)^\dagger \mathbf{p} = (\Theta V_j)^\dagger \Theta \mathbf{w} \quad \text{for } j = 1, \dots, m.$$

Hence, instead of considering the pseudo-inverse, we can derive $H_m(1 : j, j)$ by solving the least squares problem with some accurate and stable least squares solvers.

Without round-off errors, we have the following theorem.

Theorem 1. With high probability (almost surely), S_j is orthonormal and hence we have $(S_j)^\dagger = S_j^H$ to yield

$$H_m(1 : j, j) = S_j^H \mathbf{p}.$$

Proof. We shall use mathematical induction to show our claim. Let $1 \leq j \leq m$. For $j = 1$, it is trivial that

$$\|\mathbf{s}_1\| = 1 \quad \text{and} \quad \arg \min_{y \in \mathbb{C}} \|\mathbf{s}_1 y - \mathbf{p}\| = S_1^H \mathbf{p} = (\mathbf{p}, \mathbf{s}_1),$$

for the given \mathbf{p} . Assume that the argument holds for $j = i$. Then we want to show it for $j = i + 1$ to complete the proof by induction. Consider \mathbf{s}_{i+1} . By the definition, we have

$$\mathbf{s}_{i+1} = \frac{\Theta \mathbf{q}}{h} = \frac{\Theta(\mathbf{w} - V_i H_m(1 : i, i))}{h} = \frac{\Theta(\mathbf{w} - V_i S_i^H \mathbf{p})}{h} \quad \text{and} \quad \|\mathbf{s}_{i+1}\| = 1,$$

where \mathbf{q} and \mathbf{w} are given before in iteration $j = i$, and $h = \|\Theta \mathbf{q}\|$. After noting that $S_i = \Theta V_i$, we can rewrite it as

$$\mathbf{s}_{i+1} = \frac{\Theta \mathbf{w} - S_i S_i^H \mathbf{p}}{h} = \frac{\mathbf{p} - S_i S_i^H \mathbf{p}}{h} = \frac{1}{h} \left(\mathbf{p} - \sum_{k=1}^i (\mathbf{p}, \mathbf{s}_k) \mathbf{s}_k \right)$$

by the definition of \mathbf{p} . For any $i_0 \leq i$, we can observe that

$$(\mathbf{s}_{i+1}, \mathbf{s}_{i_0}) = \frac{1}{h} \left((\mathbf{p}, \mathbf{s}_{i_0}) - \sum_{k=1}^i (\mathbf{p}, \mathbf{s}_k) (\mathbf{s}_k, \mathbf{s}_{i_0}) \right) = \frac{1}{h} ((\mathbf{p}, \mathbf{s}_{i_0}) - (\mathbf{p}, \mathbf{s}_{i_0}) (\mathbf{s}_{i_0}, \mathbf{s}_{i_0})) = 0$$

since S_i is orthonormal. We thus have the orthonormal S_{i+1} . Furthermore, it implies

$$(S_{i+1})^\dagger = S_{i+1}^H \quad \text{and} \quad H_m(1 : i+1, i+1) = S_{i+1}^H \mathbf{p}.$$

Therefore, the induction completes the proof. \square

Thanks to Theorem 1, Line 8 in Algorithm 1 can be simplified and replaced by

$$H_m(1 : j, j) = S_j^H \mathbf{p} \quad \text{for } j = 1, \dots, m. \quad (2.1)$$

In practice, the quality of H_m depends on the orthogonality of S_m whereas it might have some errors from least squares solvers before. However, if the sizes of the least squares problems are small enough, the errors from the solvers will be negligible. Therefore, the computation of H_m is just matter of choice.

2.2. Randomized FGMRES

Using RGS flexible Arnoldi process, we can construct sets of Θ orthonormal basis vectors for Krylov subspace methods, e.g. (F)GMRES [1, 22]. Based on l_2 -subspace embedding properties, the randomized FGMRES will find the solution in the Krylov subspace with minimal residual in Θ norm sense rather in the standard l_2 norm sense over \mathbb{C}^n . We will approximate a solution \mathbf{x}_m in the space $\mathbf{x}_0 + \text{range}(Z_m)$ by taking into account a minimization of the sketched norm of the residual $\mathbf{r}_m = \mathbf{b} - A\mathbf{x}_m$. More precisely, we have the following minimization problem such that

$$\text{minimize } \|\mathbf{r}_m\|_{\Theta} \Leftrightarrow \text{minimize } \|\mathbf{r}_0 - AZ_m\mathbf{y}\|_{\Theta} \Leftrightarrow \text{minimize } \|\mathbf{r}_0 - V_{m+1}H_m\mathbf{y}\|_{\Theta}$$

for some $\mathbf{y} \in \mathbb{C}^m$, by the Arnoldi relation. Since V_{m+1} is Θ orthonormal (this argument is equivalent to that ΘV_{m+1} is l_2 orthonormal), we have

$$\text{minimize } \|\mathbf{r}_m\|_{\Theta} \Leftrightarrow \text{minimize } \|\mathbf{c} - H_m\mathbf{y}\| \Leftrightarrow \text{solve } \mathbf{y}^* = \arg \min_{\mathbf{y} \in \mathbb{C}^m} \|\mathbf{c} - H_m\mathbf{y}\|,$$

where $\mathbf{c} = V_{m+1}^H \Theta^H \Theta \mathbf{r}_0$.

Theorem 2. *In the randomized FGMRES, we have the minimal residual principle with respect to the sketched norm to yield*

$$(\Theta AZ_m)^H \Theta \mathbf{r}_m = \mathbf{0}.$$

Proof. Let \mathbf{x}_m be the solution of the minimization problem for the sketched norm of the residual \mathbf{r}_m where

$$\mathbf{x}_m = \mathbf{x}_0 + Z_m\mathbf{y}_m, \quad \text{for some } \mathbf{y}_m \in \mathbb{C}^m.$$

Let us define a function $J(\mathbf{y})$ by $J(\mathbf{y}) = \frac{1}{2} \|\mathbf{r}_0 - AZ_m\mathbf{y}\|_{\Theta}^2$ for $\mathbf{y} \in \mathbb{C}^m$. Note that

$$\text{minimize } \|\mathbf{r}_m\|_{\Theta} \Leftrightarrow \text{minimize } J(\mathbf{y}).$$

Then \mathbf{y}_m should satisfy that

$$\nabla J(\mathbf{y}_m) = \mathbf{0}.$$

Hence we have

$$\begin{aligned} \nabla J(\mathbf{y}_m) &= (\Theta AZ_m)^H \Theta AZ_m \mathbf{y}_m - (\Theta AZ_m)^H \Theta \mathbf{r}_0 \\ &= H_m^H V_{m+1}^H \Theta^H \Theta V_{m+1} H_m \mathbf{y}_m - (\Theta AZ_m)^H \Theta \mathbf{r}_0 \\ &= H_m^H H_m \mathbf{y}_m - (\Theta AZ_m)^H \Theta \mathbf{r}_0 \\ &= \mathbf{0}, \end{aligned}$$

by the flexible Arnoldi relation and Θ orthonormality of V_{m+1} . Since $H_m^H H_m$ is non-singular, we have

$$\mathbf{y}_m = (H_m^H H_m)^{-1} (\Theta AZ_m)^H \Theta \mathbf{r}_0. \quad (2.2)$$

Next, we consider $(\Theta AZ_m)^H \Theta \mathbf{r}_m$. By the definition of \mathbf{r}_m and (2.2), we have

$$\mathbf{r}_m = \mathbf{b} - A\mathbf{x}_m = \mathbf{r}_0 - AZ_m (H_m^H H_m)^{-1} (\Theta AZ_m)^H \Theta \mathbf{r}_0.$$

With the Arnoldi relation and Θ orthonormality, this leads us to obtain

$$\begin{aligned} (\Theta AZ_m)^H \Theta \mathbf{r}_m &= (\Theta AZ_m)^H \Theta \mathbf{r}_0 - (\Theta AZ_m)^H \Theta AZ_m (H_m^H H_m)^{-1} (\Theta AZ_m)^H \Theta \mathbf{r}_0 \\ &= (\Theta AZ_m)^H \Theta \mathbf{r}_0 - H_m^H H_m (H_m^H H_m)^{-1} (\Theta AZ_m)^H \Theta \mathbf{r}_0 \\ &= \mathbf{0}. \end{aligned}$$

□

While the approximate solution \mathbf{x}_m is an optimal minimizer of the residual norm in the sketched norm sense, it is also a quasi-optimal minimizer of the l_2 residual norm due to the l_2 -subspace embedding properties [12].

2.3. Randomized FGMRES-DR

The convergence of Krylov subspace methods is often related to the spectrum or pseudo-spectrum of the matrix. If one can remove or deflate eigenvectors associated with eigenvalues close to zero, the convergence rate will be significantly improved [5], especially if A is normal. Despite the approximate eigenvectors, the deflating strategy performs well in enhancing Krylov methods. In practice, the deflation method combines with restarting.

In this section, we introduce the randomized FGMRES with deflated restarting based on GMRES-DR of Morgan [5]. To approximate eigenvectors associated with eigenvalues of the smallest magnitude, the harmonic Ritz pairs are computed. *Rayleigh-Ritz* method has been commonly applied for solving eigenvalue problems. Let us introduce the definition of the (harmonic) Ritz pairs [23]. Furthermore, we refer to [13] for the randomized Rayleigh-Ritz method and its analysis.

Let $B \in \mathbb{C}^{n \times n}$ and \mathcal{S} be a m dimensional subspace of \mathbb{C}^n . Then a Ritz pair $(\mathbf{y}, \lambda) \in \mathbb{C}^n \times \mathbb{C}$ of B with respect to \mathcal{S} satisfies

$$\mathbf{y} \in \mathcal{S} \quad \text{and} \quad B\mathbf{y} - \lambda\mathbf{y} \perp \mathcal{S}.$$

Here, we call \mathbf{y} a *Ritz vector* associated with a *Ritz value* λ . In practice, we can define the subspace \mathcal{S} as the Krylov subspace $\mathcal{K}_m(A, \mathbf{b})$ if $B = A$ or $A\mathcal{K}_m(A, \mathbf{b})$ if $B = A^{-1}$. The latter choice gives the *harmonic* Ritz pairs of A .

In a classical manner, we consider the harmonic Ritz pairs of A with respect to the subspace $\mathcal{S} = A\mathcal{K}_m(A, \mathbf{b})$. Then we have a set of harmonic Ritz pairs \mathcal{R} such that

$$\mathcal{R} = \left\{ (V_m \mathbf{g}, \lambda) \in \mathbb{C}^n \times \mathbb{C} \mid H_m^H H_m \mathbf{g} = \lambda \hat{H}_m^H \mathbf{g}, \mathbf{g} \in \mathbb{C}^m \right\}, \quad (2.3)$$

where V_m and H_m are given by the classical Arnoldi algorithm and $\hat{H}_m = H_m(1:m, 1:m)$. Thus, to compute the harmonic Ritz pairs, we want to solve the generalized eigenvalue problem $H_m^H H_m \mathbf{g} = \lambda \hat{H}_m^H \mathbf{g}$. This problem is equivalent to the following standard eigenvalue problem

$$\left(\hat{H}_m + h_{m+1,m}^2 \hat{H}_m^{-H} \mathbf{e}_m \mathbf{e}_m^T \right) \mathbf{g} = \lambda \mathbf{g}, \quad (2.4)$$

where $h_{m+1,m} = H_m(m+1, m)$ and \mathbf{e}_m is the m -th Cartesian basis vector of \mathbb{C}^m . We refer to [5, 6] and the references therein for more details of the harmonic Ritz formulation.

In a similar way, we can formulate the eigenvalue problem of the harmonic Ritz pair with respect to the sketched inner product. In other words, we consider

$$\mathbf{y} \in \mathcal{S} \quad \text{and} \quad B\mathbf{y} - \mu\mathbf{y} \perp_{\Theta} \mathcal{S}, \quad (2.5)$$

where \perp_{Θ} denotes the sketched orthogonal. As in the harmonic Ritz formulation for the flexible variants [6, 11], we define $B = (AZ_m V_m^\dagger)^{-1}$ and $\mathcal{S} = (AZ_m V_m^\dagger) \mathcal{K}_m(A, \mathbf{b})$ with the flexible Arnoldi relation (1.2) by the RGS algorithm. Clearly, we have $V_m^\dagger = V_m^H \Theta^H \Theta$, since V_m is Θ orthonormal. After noting that $\mathbf{y} \in \text{range}(AZ_m)$, we can write (2.5) as

$$\forall \mathbf{u} \in \mathbb{C}^m, \quad \mathbf{u}^H (\Theta AZ_m)^H (\Theta V_m \mathbf{g} - \mu \Theta AZ_m \mathbf{g}) = 0, \quad (2.6)$$

where $\mathbf{y} = (AZ_m V_m^\dagger) V_m \mathbf{g} = AZ_m \mathbf{g}$ for some \mathbf{g} . Using the Arnoldi relation and the Θ orthonormality of V_{m+1} , we can obtain the eigenvalue problem such that

$$\mu H_m^H H_m \mathbf{g} = H_m^H V_{m+1}^H \Theta^H \Theta V_m \mathbf{g} = \hat{H}_m^H \mathbf{g}$$

which yields the equivalent problem (2.4) with $\lambda = 1/\mu$. Therefore, we have the same harmonic Ritz formulation regardless of GS algorithms. As a result, we can propose the algorithm of the randomized FGMRES-DR as follows. In the case where no deflation is employed, i.e. $k = 0$, this algorithm presents the restarted randomized FGMRES.

In Algorithm 2, once we compute Θ orthonormal basis, we can obtain k harmonic Ritz pairs. We refer to [6, Proposition 2] for the procedures of Lines 7-9. Then, we perform $(m - k)$ RGS flexible Arnoldi iterations

Algorithm 2: Randomized FGMRES with (deflated) restarting: RGS-FGMRES-DR (m, k)

Data: matrix $A \in \mathbb{C}^{n \times n}$, non-zero vector $\mathbf{b} \in \mathbb{C}^n$, sketching matrix $\Theta \in \mathbb{C}^{t \times n}$, size of Krylov subspace m , number of deflated vectors k , variable preconditioner M_j for $j = 1, \dots, m$, tolerance $tol > 0$, and initial vector $\mathbf{x}_0 \in \mathbb{C}^n$

Result: approximate solution \mathbf{x} for $A\mathbf{x} = \mathbf{b}$

- 1 $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$; $\beta = \|\mathbf{r}_0\|$; $\mathbf{c} = [\|\Theta\mathbf{r}_0\|, O_{1 \times m}]^T$; $\mathbf{e}_m = [O_{1 \times (m-1)}, 1]^T$;
- 2 Initial RGS flexible Arnoldi process to get V_{m+1} , Z_m and H_m with the starting vector \mathbf{r}_0 such that satisfying $AZ_m = V_{m+1}H_m$ and $V_{m+1}^H \Theta^H \Theta V_{m+1} = I_{m+1}$;
- 3 $\mathbf{y}^* = \arg \min_{\mathbf{y} \in \mathbb{C}^m} \|\mathbf{c} - H_m \mathbf{y}\|$; $\mathbf{x}_m = \mathbf{x}_0 + Z_m \mathbf{y}^*$; $\mathbf{x}_0 = \mathbf{x}_m$; $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$; $\beta = \|\mathbf{r}_0\|$; $\boldsymbol{\rho} = \mathbf{c} - H_m \mathbf{y}^*$;
- 4 **while** $\beta / \|\mathbf{b}\| > tol$ **do**
- 5 **if** $k > 0$ **then**
- 6 $h = H_m(m+1, m)$; $\hat{H}_m = H_m(1:m, 1:m)$;
- 7 Compute k harmonic Ritz vectors by solving the eigenvalue problem:
$$\left(\hat{H}_m + h^2 \hat{H}_m^{-H} \mathbf{e}_m \mathbf{e}_m^T \right) \mathbf{g}_i = \lambda_i \mathbf{g}_i \quad \text{for } i = 1, \dots, k$$

and set $G_k = [\mathbf{g}_1, \dots, \mathbf{g}_k] \in \mathbb{C}^{m \times k}$;
- 8 $G_{k+1} = \left[\begin{array}{c} G_k \\ O_{1 \times k} \end{array} \right], \boldsymbol{\rho}$;
- 9 Perform QR decomposition of G_{k+1} : $G_{k+1} = Q_{k+1} R_{k+1}$;
- 10 Define $V_{k+1} = V_{m+1} Q_{k+1}$, $Z_k = Z_m Q_{k+1}(1:m, 1:k)$ and $H_k = Q_{k+1}^H H_m Q_{k+1}(1:m, 1:k)$ satisfying $AZ_k = V_{k+1} H_k$;
- 11 **else**
- 12 $\mathbf{v}_1 = \mathbf{r}_0 / \|\mathbf{r}_0\|_{\Theta}$;
- 13 **end**
- 14 Perform $(m-k)$ steps of RGS flexible Arnoldi process to construct V_{m+1} , Z_m and H_m with V_{k+1} , Z_k and H_k such that satisfying $AZ_m = V_{m+1} H_m$ and $V_{m+1}^H \Theta^H \Theta V_{m+1} = I_{m+1}$;
- 15 Update $\mathbf{c} = V_{m+1}^H \Theta^H \Theta \mathbf{r}_0$; $\mathbf{y}^* = \arg \min_{\mathbf{y} \in \mathbb{C}^m} \|\mathbf{c} - H_m \mathbf{y}\|$; $\mathbf{x}_m = \mathbf{x}_0 + Z_m \mathbf{y}^*$; $\mathbf{x}_0 = \mathbf{x}_m$;
- $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$; $\beta = \|\mathbf{r}_0\|$; $\boldsymbol{\rho} = \mathbf{c} - H_m \mathbf{y}^*$;
- 16 **end**
- 17 $\mathbf{x} = \mathbf{x}_m$;

to construct the basis vector \mathbf{v}_i for $i = k+2, \dots, m+1$ such that $\mathbf{v}_i^H \Theta^H \Theta V_{k+1} = O_{1 \times (k+1)}$ resulting Θ orthonormal V_{m+1} . After the construction of the new basis, we can obtain the approximate solution as before.

Proposition 1. *Without deflation in Algorithm 2 ($k = 0$), we can simplify \mathbf{c} as*

$$\mathbf{c} = \beta_{\Theta} \mathbf{e}_1, \quad \text{where } \beta_{\Theta} = \|\mathbf{r}_0\|_{\Theta}.$$

Proof. The Θ orthonormality of V_{m+1} immediately gives $(\mathbf{v}_1, \mathbf{v}_i)_{\Theta} = 0, \forall i = 2, \dots, m+1$ and so we have

$$\mathbf{c} = V_{m+1}^H \Theta^H \Theta \mathbf{r}_0 = V_{m+1}^H \Theta^H \Theta \mathbf{v}_1 \|\mathbf{r}_0\|_{\Theta} = [\|\mathbf{r}_0\|_{\Theta}, 0, \dots, 0]^T = \beta_{\Theta} \mathbf{e}_1,$$

with $\mathbf{v}_1 = \mathbf{r}_0 / \|\mathbf{r}_0\|_{\Theta}$. □

Remark. *As seen in Line 8 of Algorithm 2, QR factorization is required in l_2 norm sense. We can also consider some sketching variants but it will not be beneficial since the size of G_{k+1} is only $(m+1) \times (k+1)$.*

Mostly, we consider $k < m \ll t$. Therefore, we keep using l_2 QR decomposition here. Furthermore, we note that

$$V_{k+1}^H \Theta^H \Theta V_{k+1} = Q_{k+1}^H (V_{m+1}^H \Theta^H \Theta V_{m+1}) Q_{k+1} = Q_{k+1}^H I_{m+1} Q_{k+1} = I_{k+1},$$

since $Q_{k+1} \in \mathbb{C}^{(m+1) \times (k+1)}$ is l_2 orthonormal.

Proposition 2. With $k > 0$, we can rewrite \mathbf{c} in Algorithm 2 as

$$\mathbf{c} = \begin{bmatrix} Q_{k+1}^H \boldsymbol{\rho} \\ O_{(m-k) \times 1} \end{bmatrix},$$

after the first GMRES cycle.

Proof. Let $k > 0$. We consider one cycle of Algorithm 2 after the first cycle. As in the QR decomposition of G_{k+1} , e.g. corresponding to Line 9, we denote the QR factorization of G_k by $G_k = Q_k R_k$. Then, using QR decomposition of G_k yields

$$G_{k+1} = Q_{k+1} R_{k+1} = \left[\begin{bmatrix} Q_k \\ O_{1 \times k} \end{bmatrix}, \mathbf{q}_{k+1} \right] \begin{bmatrix} R_k & \mathbf{u} \\ O_{1 \times k} & \alpha \end{bmatrix},$$

where $\mathbf{u} = [Q_k^H, O_{k \times 1}] \boldsymbol{\rho} \in \mathbb{C}^k$ and α is given with the orthogonal projection of $\boldsymbol{\rho}$ with respect to Q_k , e.g.

$$\alpha = \left\| \left(I - \tilde{Q}_k \tilde{Q}_k^H \right) \boldsymbol{\rho} \right\| \quad \text{for} \quad \tilde{Q}_k = \begin{bmatrix} Q_k \\ O_{k \times 1} \end{bmatrix}.$$

For more details of the reformulation, we refer to [6]. From this decomposition of G_{k+1} , we can define Z_k and H_k in Line 10 by

$$Z_k = Z_m^{\text{old}} Q_k \quad \text{and} \quad H_k = Q_{k+1}^H H_m^{\text{old}} Q_k,$$

respectively, where we denote the matrices defined in the previous cycle with the letter ‘old’ in superscripts. Also, we can derive

$$\boldsymbol{\rho} = Q_{k+1} \begin{bmatrix} \mathbf{u} \\ \alpha \end{bmatrix}. \quad (2.7)$$

On account of the approximate solution in the affine space $\mathbf{x}_0 + \text{range}(Z_m)$, we note that the residual resides in $\text{range}(V_{m+1})$. It implies that

$$\mathbf{r}_0 = V_{m+1}^{\text{old}} \boldsymbol{\rho} = V_{k+1} \begin{bmatrix} \mathbf{u} \\ \alpha \end{bmatrix}, \quad (2.8)$$

by (2.7), where V_{m+1}^{old} is given by the previous cycle, i.e. before performing further $(m-k)$ Arnoldi iterations. When we complete the construction of Krylov basis vectors, e.g. corresponding to Line 14, (2.8) leads us to obtain

$$\mathbf{c} = V_{m+1}^H \Theta^H \Theta \mathbf{r}_0 = V_{m+1}^H \Theta^H \Theta V_{k+1} \begin{bmatrix} \mathbf{u} \\ \alpha \end{bmatrix}.$$

Then Θ orthonormality implies

$$\mathbf{c} = \begin{bmatrix} \mathbf{u} \\ \alpha \\ O_{(m-k) \times 1} \end{bmatrix}. \quad (2.9)$$

Moreover, in practice, we can obtain (2.9) without computing \mathbf{u} and α separately. For instance, using (2.7) and l_2 orthogonality of Q_{k+1} , (2.9) can be rewritten as

$$\mathbf{c} = V_{m+1}^H \Theta^H \Theta V_{k+1} (Q_{k+1}^H Q_{k+1}) \begin{bmatrix} \mathbf{u} \\ \alpha \end{bmatrix} = V_{m+1}^H \Theta^H \Theta V_{k+1} Q_{k+1}^H \boldsymbol{\rho},$$

and Θ orthonormality yields

$$\mathbf{c} = \begin{bmatrix} Q_{k+1}^H \boldsymbol{\rho} \\ O_{(m-k) \times 1} \end{bmatrix}. \quad (2.10)$$

Therefore, we want to solve $\mathbf{y}^* = \arg \min_{\mathbf{y} \in \mathbb{C}^m} \|\mathbf{c} - H_m \mathbf{y}\|$ with (2.10) to approximate the solution for the current cycle. \square

Remark. We note that H_k is no longer an upper Hessenberg matrix, and neither is H_m in Line 10. In general, they are dense full matrices. But it does not matter in practice. Because H_m is the only $m+1$ by m matrix hence the least squares solver will not degrade with respect to H_m .

3. Deflation of singular vectors

From the inspiring work of [8], Al Daas et al. have investigated the deflation based on singular vectors associated with the smallest singular values [10]. To deflate these vectors, another variant of GMRES-DR will be employed, for instance, GCRO-DR [9, 11]. As a result, [10] presents GMRES-MDR and its convergence analysis. Here, we introduce the flexible variant of GMRES-MDR involving RGS.

3.1. Randomized FGMRES-MDR

First of all, we give how to approximate the right singular vectors of A based on computing Ritz pairs. Recall the definition of Ritz pairs. We can also compute approximate singular vectors arising from

$$\text{finding } (\mathbf{y}, \lambda) \quad \text{such that} \quad \mathbf{y} \in \mathcal{S} \quad \text{and} \quad B\mathbf{y} - \lambda\mathbf{y} \perp \mathcal{S},$$

where $B = A^H A$ and $\mathcal{S} = \mathcal{K}_m(A, \mathbf{b})$. In a similar way with (2.3) and (2.4), we can derive the standard eigenvalue problem such that

$$H_m^H H_m \mathbf{g} = \lambda \mathbf{g} \quad \text{for } \mathbf{g} \in \mathbb{C}^m. \quad (3.1)$$

We call the pair $(V_m \mathbf{g}, \lambda)$ a *singular Ritz pair*.

In the randomization sense, we want to compute the approximate right singular vectors of the sketched A . That is, we consider an approximation of the singular vector of the matrix ΘA . In the flexible variant way, taking into account $B = (\Theta A Z_m S_m^\dagger)^H (\Theta A Z_m S_m^\dagger)$ and $\mathcal{S} = \Theta \mathcal{K}_m(A, \mathbf{b}) = \text{range}(S_m)$ where $S_m = \Theta V_m$ given by RGS Arnoldi process, we can obtain the problem of the singular Ritz pairs such that finds a pair of $\mathbf{y} \in \mathcal{S}$ and λ satisfying

$$\forall \mathbf{u} \in \mathbb{C}^m, \quad \mathbf{u}^H S_m^H ((\Theta A Z_m S_m^\dagger)^H (\Theta A Z_m S_m^\dagger) S_m \mathbf{g} - \lambda S_m \mathbf{g}) = 0, \quad (3.2)$$

where $\mathbf{y} = S_m \mathbf{g}$. Using the fact that S_{m+1} is orthonormal and the flexible Arnoldi relation (1.2), (3.2) yields the same eigenvalue problem (3.1), for instance

$$\begin{aligned} & S_m^H ((\Theta A Z_m S_m^\dagger)^H (\Theta A Z_m S_m^\dagger) S_m \mathbf{g} - \lambda S_m \mathbf{g}) \\ &= S_m^H (S_m H_m^H S_{m+1}^H S_{m+1} H_m S_m^H S_m \mathbf{g} - \lambda S_m \mathbf{g}) \\ &= S_m^H (S_m H_m^H (S_{m+1}^H S_{m+1}) H_m (S_m^H S_m) \mathbf{g} - \lambda S_m \mathbf{g}) \\ &= S_m^H (S_m H_m^H H_m \mathbf{g} - \lambda S_m \mathbf{g}) \\ &= S_m^H S_m H_m^H H_m \mathbf{g} - \lambda S_m^H S_m \mathbf{g} \\ &= H_m^H H_m \mathbf{g} - \lambda \mathbf{g} \\ &= 0. \end{aligned}$$

Moreover, as in [11, 10], introducing W_m with $\text{range}(W_m) = \text{range}(S_m)$ leads us to obtain the eigenvalue problem with respect to $B = (\Theta A W_m^\dagger)^H (\Theta A W_m^\dagger)$ and $\mathcal{S} = \text{range}(W_m)$ formed by

$$H_m^H H_m \mathbf{g} = \lambda W_m^H W_m \mathbf{g} \quad \text{for } \mathbf{g} \in \mathbb{C}^m. \quad (3.3)$$

The definition of W_m is specified in Algorithm 3. For more detail of analysis of W_m , we refer to [11, 10].

By solving eigenvalue problems (3.1) or (3.3), we can immediately define G_k . While Morgan [5] introduced the augmented formula G_{k+1} with the harmonic Ritz pairs and the residual vector, the singular Ritz pairs are not applicable for the construction of G_{k+1} . To be specific, when the augmented G_{k+1} is defined in FGMRES-DR, the argument of the harmonic residual vectors such that $A Z_m \mathbf{g} - \lambda V_m \mathbf{g} \in \text{range}(V_{m+1})$ is

used. However, the singular residual vectors do not reside in $\text{range}(V_{m+1})$. Hence, $AZ_m G_k$ in FGMRES-MDR cannot be expressed with V_{m+1} as in FGMRES-DR. As a consequence, we could not derive deflated subspaces satisfying Arnoldi identity such that $AZ_k = V_{k+1}H_k$ in that way of Algorithm 2. Instead, GMRES-MDR was proposed based on GCRO-DR by Al Daas et al. [10]. In this manner but without diagonal scaling (cf. [11]), we can propose the randomized FGMRES-MDR as following.

Algorithm 3: Randomized FGMRES with SVD based deflated restarting: RGS-FGMRES-MDR
(m, k)

Data: matrix $A \in \mathbb{C}^{n \times n}$, non-zero vector $\mathbf{b} \in \mathbb{C}^n$, sketching matrix $\Theta \in \mathbb{C}^{t \times n}$, size of Krylov subspace m , number of deflated vectors k , variable preconditioner M_j for $j = 1, \dots, m$, tolerance $tol > 0$, and initial vector $\mathbf{x}_0 \in \mathbb{C}^n$

Result: approximate solution \mathbf{x} for $A\mathbf{x} = \mathbf{b}$

- 1 $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$; $\beta = \|\mathbf{r}_0\|$; $\mathbf{c} = [\|\Theta\mathbf{r}_0\|, O_{1 \times m}]^T$;
 - 2 Initial RGS flexible Arnoldi process to get V_{m+1} , Z_m and H_m with the starting vector \mathbf{r}_0 such that satisfying $AZ_m = V_{m+1}H_m$ and $V_{m+1}^H \Theta^H \Theta V_{m+1} = I_{m+1}$;
 - 3 Solve $\mathbf{y}^* = \arg \min_{\mathbf{y} \in \mathbb{C}^m} \|\mathbf{c} - H_m \mathbf{y}\|$;
 - 4 $\mathbf{x}_m = \mathbf{x}_0 + Z_m \mathbf{y}^*$; $\mathbf{x}_0 = \mathbf{x}_m$; $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$; $\beta = \|\mathbf{r}_0\|$; $W_m = \Theta V_m$;
 - 5 **while** $\beta / \|\mathbf{b}\| > tol$ **do**
 - 6 Compute k singular vectors by solving the eigenvalue problem:

$$H_m^H H_m \mathbf{g}_i = \lambda_i W_m^H W_m \mathbf{g}_i \quad \text{for } i = 1, \dots, k$$
 and set $G_k = [\mathbf{g}_1, \dots, \mathbf{g}_k] \in \mathbb{C}^{m \times k}$;
 - 7 Perform QR decomposition of $H_m G_k$: $H_m G_k = Q_k R_k$;
 - 8 Define $V_k = V_{m+1} Q_k$ and $Z_k = Z_m G_k R_k^{-1}$ satisfying $AZ_k = V_k$;
 - 9 Define $W_k = W_m G_k R_k^{-1}$;
 - 10 Perform $(m - k)$ steps of RGS flexible Arnoldi process to construct V_{m-k+1} , Z_{m-k} and H_{m-k} with the starting vector \mathbf{r}_0 with respect to the linear operator $(I_n - V_k V_k^H \Theta^H \Theta)A$ such that satisfying

$$(I_n - V_k V_k^H \Theta^H \Theta)AZ_{m-k} = V_{m-k+1}H_{m-k} \quad \text{with } V_{m-k+1}^H \Theta^H \Theta V_{m-k+1} = I_{m-k+1};$$
 - 11 Define $V_{m+1} = [V_k, V_{m-k+1}]$, $Z_m = [Z_k, Z_{m-k}]$ and $H_m = \begin{bmatrix} I_k & V_k^H \Theta^H \Theta AZ_{m-k} \\ O_{(m-k+1) \times k} & H_{m-k} \end{bmatrix}$
 yielding $AZ_m = V_{m+1}H_m$ and $V_{m+1}^H \Theta^H \Theta V_{m+1} = I_{m+1}$;
 - 12 Define $W_m = [W_k, \Theta V_{m+1}(:, k+1:m)]$;
 - 13 Update $\mathbf{c} = V_{m+1}^H \Theta^H \Theta \mathbf{r}_0$; $\mathbf{y}^* = \arg \min_{\mathbf{y} \in \mathbb{C}^m} \|\mathbf{c} - H_m \mathbf{y}\|$; $\mathbf{x}_m = \mathbf{x}_0 + Z_m \mathbf{y}^*$; $\mathbf{x}_0 = \mathbf{x}_m$;
 $\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0$; $\beta = \|\mathbf{r}_0\|$;
 - 14 **end**
 - 15 $\mathbf{x} = \mathbf{x}_m$;
-

Remark. After the harmonic Ritz values based deflation steps and $(m - k)$ iterations of Arnoldi, H_m is not upper Hessenberg, since H_k becomes dense in general. However, in case of singular value based deflation, we have the upper Hessenberg matrix.

Once we compute the sketch in the Arnoldi iteration, it is not necessary to compute ΘV_{m+1} again. Since we have already obtain S_{m+1} in the Arnoldi process, we can use it straightforwardly. For instance, we can define \mathbf{c} by $\mathbf{c} = S_{m+1}^H \Theta \mathbf{r}_0$. Furthermore, in Algorithm 3, if we define $S_k = S_{m+1} Q_k$, $V_k^H \Theta^H$ can be replaced by S_k^H in Lines 10 and 11.

Theorem 3. Let V_{m+1} be the matrix formed by the Krylov vectors in Algorithm 3. Then V_{m+1} is Θ orthonormal for each cycle.

Proof. For the first cycle, it is trivial since RGS Arnoldi process imposes Θ orthonormality. Let us consider

V_{m+1} after deflating. We want to show that $(\Theta V_{m+1})^H \Theta V_{m+1} = I_{m+1}$. Suppose V_{m+1}^{old} is the Θ orthonormal matrix from the previous cycle. Then we have

$$(\Theta V_k)^H \Theta V_k = (\Theta V_{m+1}^{\text{old}} Q_k)^H \Theta V_{m+1}^{\text{old}} Q_k = Q_k^H I_{m+1} Q_k = I_k,$$

by l_2 orthogonality of Q_k and Θ orthogonality of V_{m+1}^{old} . On the other hand, RGS Arnoldi iterations give us Θ orthonormal V_{m-k+1} . Hence we shall show that each column vector in V_{m-k+1} is orthogonal to every column vector in V_k .

By recalling Theorem 2, the minimal residual principle implies that

$$(\Theta AZ_m^{\text{old}})^H \Theta \mathbf{r}_0 = \mathbf{0},$$

where $\mathbf{r}_0 = \mathbf{r}_m^{\text{old}}$. Since $AZ_k = V_k$, we can obtain

$$\begin{aligned} (\Theta V_k)^H \Theta \mathbf{r}_0 &= (\Theta AZ_k)^H \Theta \mathbf{r}_0 \\ &= R_k^{-H} G_k^H (\Theta AZ_m^{\text{old}})^H \Theta \mathbf{r}_0 \\ &= \mathbf{0}. \end{aligned}$$

Thus, we have $(\Theta V_k)^H \Theta \mathbf{v}_{k+1} = (\Theta V_k)^H \Theta \mathbf{r}_0 / \|\mathbf{r}_0\|_{\Theta} = \mathbf{0}$. Before we consider other columns of V_{m-k+1} , we note that for any $\mathbf{v} \in \text{range}((I_n - V_k V_k^H \Theta^H \Theta)A)$, the argument $(\Theta V_k)^H \Theta \mathbf{v} = \mathbf{0}$ always holds. For example, if we denote $\mathbf{v} = (I_n - V_k V_k^H \Theta^H \Theta)A \mathbf{w}$ for some $\mathbf{w} \in \mathbb{C}^n$, the sketched inner product of \mathbf{v} and i -th column of V_k gives

$$\begin{aligned} ((I_n - V_k V_k^H \Theta^H \Theta)A \mathbf{w}, \mathbf{v}_i)_{\Theta} &= \mathbf{v}_i^H \Theta^H \Theta (I_n - V_k V_k^H \Theta^H \Theta)A \mathbf{w} \\ &= \mathbf{v}_i^H \Theta^H \Theta A \mathbf{w} - \mathbf{e}_i^H V_k^H \Theta^H \Theta A \mathbf{w} \\ &= \mathbf{v}_i^H \Theta^H \Theta A \mathbf{w} - \mathbf{v}_i^H \Theta^H \Theta A \mathbf{w} \\ &= 0, \end{aligned}$$

where \mathbf{e}_i is the i -th standard basis of \mathbb{C}^k for $i = 1, \dots, k$. Since \mathbf{v}_j for any $j = k+2, \dots, m+1$ resides in $\text{range}((I_n - V_k V_k^H \Theta^H \Theta)A)$, every column in V_{m-k+1} is Θ orthogonal to V_k . Consequently, V_{m+1} is Θ orthonormal. \square

As shown in Theorem 3, the basis vectors of the Krylov subspace are well-constructed with Θ orthonormality and the flexible Arnoldi relation. Next, we present how the deflation of singular vectors of ΘA works. Thanks to Theorem 2, we can extend theoretical results on the convergence of the restarted Krylov methods, which addressing that the smallest singular value play an important role at restart [8], in the sketched norm sense. Moreover, as [10, Theorem 4.1] showed the way of singular vector based deflation in the Krylov methods, we can prove the approximate theorem for deflation of singular vectors associated with the k smallest singular values in the sketched low-dimensional space.

Theorem 4. *Let \mathbf{x}_* be the exact solution of $A \mathbf{x} = \mathbf{b}$. Consider the singular value decomposition of ΘA with splitting such that*

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

where the diagonal of Σ_2 consists of k smallest singular values of ΘA and Σ_1 is the diagonal matrix containing other singular values. Let $\hat{\mathbf{x}}$ and $\tilde{\mathbf{x}}$ be an exact solution and an approximate solution of the linear system

$$(I_t - U_2 U_2^H) \Theta A \mathbf{x} = (I_t - U_2 U_2^H) \Theta \mathbf{b}, \quad (3.4)$$

with $\|\hat{\mathbf{x}} - \tilde{\mathbf{x}}\| \leq \eta$ and $\eta > 0$, respectively. Then it holds that

$$\left\| \mathbf{x}_* - (I_n - V_2 V_2^H) \tilde{\mathbf{x}} - V_2 \Sigma_2^\dagger U_2^H \Theta \mathbf{b} \right\| \leq \eta. \quad (3.5)$$

Proof. The proof follows the similar arguments in the proof of [10, Theorem 4.1]. Since \mathbf{x}_* is the solution of the original system, \mathbf{x}_* is a solution of (3.4). Let \mathbf{x} be a solution of (3.4). Then we have

$$\Theta A \mathbf{x} = (I_t - U_2 U_2^H) \Theta \mathbf{b} + U_2 U_2^H \Theta A \mathbf{x} = \Theta A \mathbf{x}_* - U_2 U_2^H (\Theta \mathbf{b} - \Theta A \mathbf{x}).$$

Taking the pseudo inverse of ΘA implies that

$$\mathbf{x} = \mathbf{x}_* - V_2 \Sigma_2^\dagger U_2^H (\Theta \mathbf{b} - \Theta A \mathbf{x}) = \mathbf{x}_* + V_2 (V_2^H \mathbf{x} - \Sigma_2^\dagger U_2^H \Theta \mathbf{b}).$$

Moreover, for any $\mathbf{u} \in \mathbb{C}^k$, $\mathbf{x}, \mathbf{x}_* + V_2 \mathbf{u}$ is a solution of (3.4), since

$$(I_t - U_2 U_2^H) \Theta A V_2 \mathbf{u} = (I_t - U_2 U_2^H) U_2 \Sigma_2 \mathbf{u} = U_2 \Sigma_2 \mathbf{u} - U_2 \Sigma_2 \mathbf{u} = 0.$$

Thus, any solution of (3.4) can be expressed as $\mathbf{x}_* + V_2 \mathbf{u}$ for some $\mathbf{u} \in \mathbb{C}^k$ and hence we can write $\hat{\mathbf{x}} = \mathbf{x}_* + V_2 \hat{\mathbf{u}}$ for some $\hat{\mathbf{u}}$.

On the other hand, using the pseudo inverse of ΘA , we have

$$V_2 V_2^H \mathbf{x}_* = V_2 V_2^H (\Theta A)^\dagger \Theta \mathbf{b} = V_2 \Sigma_2^\dagger U_2^H \Theta \mathbf{b}. \quad (3.6)$$

With (3.6), we can decompose \mathbf{x}_* by

$$\mathbf{x}_* = (I_n - V_2 V_2^H) \mathbf{x}_* + V_2 V_2^H \mathbf{x}_* = (I_n - V_2 V_2^H) \hat{\mathbf{x}} + V_2 \Sigma_2^\dagger U_2^H \Theta \mathbf{b}, \quad (3.7)$$

since $\hat{\mathbf{x}} = \mathbf{x}_* + V_2 \hat{\mathbf{u}}$ and $(I_n - V_2 V_2^H) V_2 \hat{\mathbf{u}} = \mathbf{0}$.

Turning back to our main argument, let us consider

$$\mathbf{x}_* - (I_n - V_2 V_2^H) \tilde{\mathbf{x}} - V_2 \Sigma_2^\dagger U_2^H \Theta \mathbf{b}.$$

The substitution of (3.7) into this, we can obtain

$$\mathbf{x}_* - (I_n - V_2 V_2^H) \tilde{\mathbf{x}} - V_2 \Sigma_2^\dagger U_2^H \Theta \mathbf{b} = (I_n - V_2 V_2^H) (\hat{\mathbf{x}} - \tilde{\mathbf{x}}),$$

thus we have

$$\left\| \mathbf{x}_* - (I_n - V_2 V_2^H) \tilde{\mathbf{x}} - V_2 \Sigma_2^\dagger U_2^H \Theta \mathbf{b} \right\| = \left\| (I_n - V_2 V_2^H) (\hat{\mathbf{x}} - \tilde{\mathbf{x}}) \right\| \leq \|\hat{\mathbf{x}} - \tilde{\mathbf{x}}\| \leq \eta,$$

by the property of the projection. □

Theorem 4 is the sketched analogue of [10, Theorem 4.1]. In Theorem 4, (3.5) describes that the error norm in the solution \mathbf{x}_* of the original system corresponding to an approximated solution $\tilde{\mathbf{x}}$ of the sketched projection problem (3.4) and the correction term is bounded by the error norm in the solution of (3.4). For more convergence analysis of (F)GCRO-DR and SVD based deflation, we refer to [8, 10, 11] and the references therein.

Remark. After noting that $S_k = \Theta V_k$ and $S_{m-k+1} = \Theta V_{m-k+1}$, in the sketched orthogonal projection, we have

$$\Theta (I_n - V_k V_k^H \Theta^H \Theta) A Z_{m-k} = (I_t - \Theta V_k V_k^H \Theta^H) \Theta A Z_{m-k} = (I_t - S_k S_k^H) \Theta A Z_{m-k} = S_{m-k+1} H_{m-k}$$

by sketching the result of Line 10 of Algorithm 3.

By Theorems 3 and 4, and the above remark, we can immediately extend the convergence results obtained in [8, 10] with respect to the sketched manner. In a similar way to the previous section, we can simplify \mathbf{c} for the least squares problem for the randomized FGMRES-MDR as in Propositions 1 and 2.

Proposition 3. *With $k > 0$, we can rewrite \mathbf{c} in Algorithm 3 as*

$$\mathbf{c} = \beta_{\Theta} \mathbf{e}_{k+1},$$

where $\beta_{\Theta} = \|\mathbf{r}_0\|_{\Theta}$ and \mathbf{e}_{k+1} is the $(k+1)$ -th standard basis of \mathbb{C}^{m+1} after the first GMRES cycle.

Proof. After noting that $\mathbf{v}_{k+1} = \mathbf{r}_0 / \|\mathbf{r}_0\|_{\Theta}$, the Θ orthonormality leads us to get

$$\mathbf{c} = V_{m+1}^H \Theta^H \Theta \mathbf{r}_0 = V_{m+1}^H \Theta^H \Theta \mathbf{v}_{k+1} \|\mathbf{r}_0\|_{\Theta} = \beta_{\Theta} \mathbf{e}_{k+1},$$

where $\beta_{\Theta} = \|\mathbf{r}_0\|_{\Theta}$ and \mathbf{e}_{k+1} is the $(k+1)$ -th standard basis of \mathbb{C}^{m+1} . \square

While we can choose the restart parameter for its effectiveness with the adaptive strategy, we have no clue to define the deflation parameter for the most optimal setting. Only experimentally, larger k would contain more deflated vectors associated with eigenvalues/singular values which are clustered at the origin. Accordingly, we can think of the adaptive choice of deflation parameter k for each cycle when k approximate eigenvalues/singular values are sufficiently close to zero. More precisely, in [10, Algorithm 4], the authors introduced a spectral radius criterion $0 < \gamma < 1$ to take the maximal $k_i \leq m$ for each i -th cycle satisfying that

$$0 < |\lambda_1| \leq |\lambda_2| \leq \dots \leq |\lambda_{k_i}| < \gamma,$$

where λ_j is the approximate eigenvalues/singular values for $j = 1, \dots, k_i$. Therefore, how much deflation improves convergence will be controlled by the choice of γ . We give remarks that too small γ implies no deflation, i.e. making $k_i = 0$, and too large γ forces not to perform Arnoldi process after deflating if $k_i = m$. However, in this paper, we consider the fixed k for simplicity.

3.2. Usual orthonormality

We can additionally impose l_2 orthogonality on the given Θ orthonormal V_{m+1} by QR factorization. Applying l_2 QR decomposition to V_{m+1} leads to the new l_2 orthonormal \bar{V}_{m+1} satisfying the Arnoldi identity. More precisely, when we have $V_{m+1} = \bar{V}_{m+1} U_m$ by QR factorization, we can define the new Krylov subspace basis such that

$$AZ_m = \bar{V}_{m+1} \bar{H}_m \quad \text{with} \quad \bar{V}_{m+1}^H \bar{V}_{m+1} = I_{m+1} \text{ and } \bar{H}_m = U_m H_m,$$

from the previous RGS flexible Arnoldi iteration.

Since the Krylov basis after the QR factorization procedure is l_2 orthonormal, the linear operator becomes simpler, i.e. employing l_2 orthogonal projection rather than the sketched orthogonal projection. Except for performing additional QR decomposition, the randomized FGMRES-MDR involving l_2 orthogonality consists of the same progress as Algorithm 3. This method gives some benefits in orthogonality and stability of the construction of Krylov basis vectors. From the fact that the RGS provides well-conditioned matrices whose columns are Krylov basis vectors, e.g. $\kappa(V_{m+1}) \approx 1$, we ensure the stability of QR decomposition for $V_{m+1} = \bar{V}_{m+1} U_m$. Because the algorithms of QR depend on the condition number of a matrix. It also allows us to impose better quality of orthonormal basis vectors. It seems very similar to *reorthogonalization* strategy such as CGS2 and MGS2 [17] to avoid the loss of orthogonality in V_{m+1} . Thus, we call the RGS algorithm imposing l_2 orthogonality the *RGS2* algorithm.

Remark. *This reorthogonalization is also applicable to other randomized GMRES. Hence, we can easily implement the randomized FGMRES-DR with l_2 orthogonal basis vectors. In this paper, we focus on RGS implementations without reorthogonalization but present some numerical results of RGS2 with FGMRES-MDR.*

4. Advantages of random sketching

In this section, we describe the main benefits of our algorithms: (i) stable and (ii) efficiency. We present the stability analysis of RGS algorithm and compare with other variants of the GS process. Additionally, we show computational costs corresponding to Algorithms of RGS procedures, the randomized FGMRES-DR, and the randomized FGMRES-MDR.

4.1. Stable RGS algorithm

We recall the theoretical results from [13] regarding numerical stability bounds for RGS iterations. Let us consider QR decomposition of a matrix $W \in \mathbb{C}^{n \times m}$ to give $W \approx QR$ by the RGS algorithm with finite precision arithmetic. According to [12, Theorem 3.2], we have the stability bound of the RGS process given by

$$\|W - QR\|_F \leq Cm^{3/2}\bar{u}\|W\|_F, \quad (4.1)$$

with $\kappa(Q) = 1 + O(\epsilon)$, where \bar{u} denotes the unit round-off (e.g. $\bar{u} = O(10^{-16})$ for IEEE double precision) and C is a positive constant. Note that (4.1) relies on ϵ embedding property of Θ hence we assume that Θ is l_2 -subspace embedding for W and Q for $\epsilon \leq 1/2$. As remarked in [12], although the proof for the bound (4.1) followed with a multi-precision model, it holds in a unique precision model too. Moreover, the loss of Θ orthogonality of Q (equivalent to l_2 orthogonality of the sketched Q) is bounded by

$$\|I_m - (\Theta Q)^H(\Theta Q)\|_F \leq Cm^2\bar{u}\kappa(W), \quad (4.2)$$

for a positive constant C . Please refer to [12, Theorem 3.3] for more details of the bound (4.2).

On the other hand, we can see the rounding error analysis of CGS and MGS in [24, 25] and [26], respectively. It is shown that the approximation error $\|W - QR\|_F$ is bounded in a similar way with (4.1) involving some low degree polynomials in m and n . While the loss of Θ orthogonality of Q in RGS is proportional to $\kappa(W)$ in (4.2), we have the bounds of the loss of l_2 orthogonality of Q given by

$$\text{CGS: } \|I_m - Q^H Q\| \leq \xi_1(m, n)\bar{u}\kappa(W)^2, \quad (4.3)$$

$$\text{MGS: } \|I_m - Q^H Q\| \leq \xi_2(m, n)\bar{u}\kappa(W), \quad (4.4)$$

with respect to GS process, where $\xi_1(m, n)$ and $\xi_2(m, n)$ are low degree polynomials in m and n . Thus, we can observe that the RGS algorithm is more stable than the CGS algorithm and as stable as the MGS algorithm.

To manage the loss of orthogonality from the presence of the round-off error, reorthogonalization is widely used in CGS and MGS algorithms. For instance, under the assumption $\xi_1(m, n)\bar{u}\kappa(W)^2 < 1$ (*resp.* $\xi_2(m, n)\bar{u}\kappa(W) < 1$) the loss of l_2 orthogonality of Q computed by CGS2 (*resp.* MGS2) can be bounded without the condition number term. For more details, we refer to [17] and the references therein. In case of RGS2, it is easy to show that

$$\|I_m - \tilde{Q}^H \tilde{Q}\| \leq \xi_3(m, n)\bar{u}, \quad (4.5)$$

where $\xi_3(m, n)$ is a low degree polynomials in m and n and \tilde{Q} is computed by the RGS2 algorithm. As seen in Section 3.2, the RGS2 algorithm combines RGS and CGS/MGS algorithms. More precisely, when we apply the RGS2 process to W , we have Q for $W = QR$ by RGS and \tilde{Q} for $Q = \tilde{Q}U$ by either CGS or MGS. After noting that $\kappa(Q) = 1 + O(\epsilon)$, either (4.3) or (4.4) implies (4.5).

Remark. *The numerical stability of GS algorithms can be also measured by the condition number of Q . In infinite precision arithmetic, we have the exact orthonormal matrix Q and hence it always holds that $\kappa(Q) = 1$. Therefore, with finite precision arithmetic, $\kappa(Q)$ presents the quality of orthogonality, i.e. the numerical stability.*

In the context of the flexible Arnoldi process, W can be given as AZ_m . In the variants of the GMRES method, the orthogonality of the matrix formed by Krylov vectors plays an important role in minimizing the residual norm. Since the minimization problem is usually formulated using orthonormality constraints, failing to maintain orthonormality can lead to non-optimal solutions or even convergence failure. In the randomized FGMRES, Propositions 1, 2 and 3 will be no longer applicable if the loss of Θ orthogonality exceeds a certain threshold. In such cases, we must solve a least squares problem to determine the coefficients \mathbf{c} , which are given by $V_{m+1}^H \Theta^H \Theta V_{m+1}$.

4.2. Computational costs

One of the main advantages of RGS is the reduction of computational costs in orthonormalization. It is well known that the complexity of CGS is $2nm^2$ flops, and MGS has the same cost [27]. On the other hand, when we suppose $t \ll n$, we can disregard costs of the random sketching and the computations with sketched vectors, e.g. \mathbf{p} and \mathbf{s} in Algorithm 1. More precisely, let Θ be chosen to give the most efficient in sketching. Then, for instance, if the sketching is performed by SRHT, the random sketching in Lines 7 and 10, Algorithm 1 requires at most $4n \log(t+1)$ flops in total for each iteration [12]. Let us consider the computational cost of RGS corresponding to Lines 7–11 in Algorithm 1 for j -th iteration as following in Table 1.

	Corresponding line	Complexity (flops)
Sketch a vector of \mathbb{C}^n	Lines 7 and 10	$4n \log(t+1)$
Compute $H_m(1:j, j)$	Line 8	$O(tj^2)$ or $2tj$
Compute q	Line 9	$2nj$
Compute $H_m(j+1, j)$, \mathbf{v}_{j+1} and \mathbf{s}_{j+1}	Line 11	$n + t^2 + t$

Table 1: Cost analysis in Θ orthogonalization for j -th iteration.

For sufficiently large j , the complexity of Lines 7 and 10 is dominated by Line 9, and since $j \leq m \leq t \ll n$. Depending on the way of computing $H_m(j+1, j)$, Line 8 requires different cost. For example, if we use Householder QR or Givens QR as the least squares solver, it costs $2tj^2$ flops or $3tj^2$ flops, respectively. Otherwise, using (2.1) leads us to obtain the complexity $2tj$ flops for Line 8. With $j \leq m \leq t \ll n$, the cost of Lines 8 and 11 is dominated by that of Line 9 too. Consequently, we have

$$\text{total cost of RGS} \approx \sum_{j=1}^m 2nj = nm^2 \text{ flops.}$$

Therefore, RGS algorithm has just half cost of CGS/MGS algorithms.

Regardless of the choice of the Gram-Schmidt process to construct Krylov basis vectors, FGMRES methods have the same computational costs except for the orthogonalization process. In other words, the difference in costs between our proposed algorithms and other FGMRES based on CGS/MGS occurs only in the Arnoldi process where RGS has half cost of CGS/MGS. On the other hand, in FGMRES, other costs of computations such as solving least squares problems for \mathbf{y}^* , approximating the solutions and updating residuals, are dominated by the computational cost of Arnoldi iteration excluding computations of applying operators and preconditioners. For example, they require at most $O(nm)$ flops whereas the Gram-Schmidt process needs $O(nm^2)$ flops. In the same way, FGMRES-DR has similar complexity. As it is commonly known that the complexity of finding eigendecomposition (via SVD) of a $m \times m$ matrix is m^3 flops, the total cost of FGMRES-DR is either nm^2 flops in the randomized version or $2nm^2$ flops in the CGS/MGS variants for $m \ll n$. It holds in the singular vectors based deflation as well. More precisely, we refer to [6] for the computational cost of a generic cycle of FGMRES and FGMRES-DR and [10] for that of GMRES-MDR. Therefore, the randomized FGMRES methods has half less complexity than other FGMRES methods based on CGS/MGS. However, if we employ the RGS2 algorithm, we need to perform l_2 QR decomposition on the Krylov basis matrix, resulting in an increase in the computational cost. Hence the total cost becomes $3nm^2$ flops. Nevertheless, it is less than the cost of other reorthogonalized methods such as CGS2 and MGS2 [17] that is $4nm^2$ flops.

5. Numerical experiments

In this section, we present numerical experiments of our proposed algorithms to exhibit numerical performance in stability as well as convergence. We have implemented the proposed algorithms in MATLAB and used the MATLAB's local functions, e.g. to solve least squares problem by `\` and eigenvalue problems by `eig()`, to define preconditioners, etc. Furthermore, the rescaled Rademacher distribution has

been employed in sketching, i.e. the sketching matrix Θ will be defined by the Rademacher distribution. All MATLAB codes for the following numerical examples are available at Jang's GitHub (<https://github.com/Yongseok7717/RandomizedGMRES>).

First of all, we compare the stability of RGS for the construction of an orthogonal basis to validate our methodology in regarding with the well-known CGS/MGS algorithms without reorthogonalization. Next, we solve linear systems arising in compressible turbulent flows with using the randomized FGMRES methods. We shall consider their numerical performances regarding convergence such as relative residual norms with respect to the number of Arnoldi iterations.

5.1. Example 1: QR decomposition of an ill-conditioned matrix

Let us define a synthetic function such that

$$f_\mu(x) = \frac{\sin(10(\mu + x))}{\cos(100(\mu - x)) + 1.1},$$

where $x \in [0, 1]$ and $\mu \in [0, 1]$. Using this function, we consider the following matrix W defined by

$$W_{ij} = f_{\mu_j}(x_i) \quad \text{for } x_i = i\delta_x \text{ and } \mu_j = j\delta_\mu,$$

where $\delta_x = 1/n$ and $\delta_\mu = 1/m$ with $n = 10^5$ and $m = 250$. Hence W is a matrix of n by m and this is ill-conditioned, e.g. the condition number of $W = O(10^{14})$.

We decompose the submatrices of W by QR factorization with respect to Gram-Schmidt process, e.g. CGS, MGS and RGS algorithms. To be specific, we perform QR factorization of W_i which is a $n \times i$ matrix extracted from the first i columns of W , resulting in $W_i = W(:, 1:i) = Q_i R_i$ with respect to GS process. We then compute condition numbers of Q_i varying with $i = 1, \dots, 250$ and approximation errors of the QR factorization with the quantity $\|W_i - Q_i R_i\| / \|W_i\|$. Note that the condition numbers close to 1 imply that the the algorithms are numerically stable.

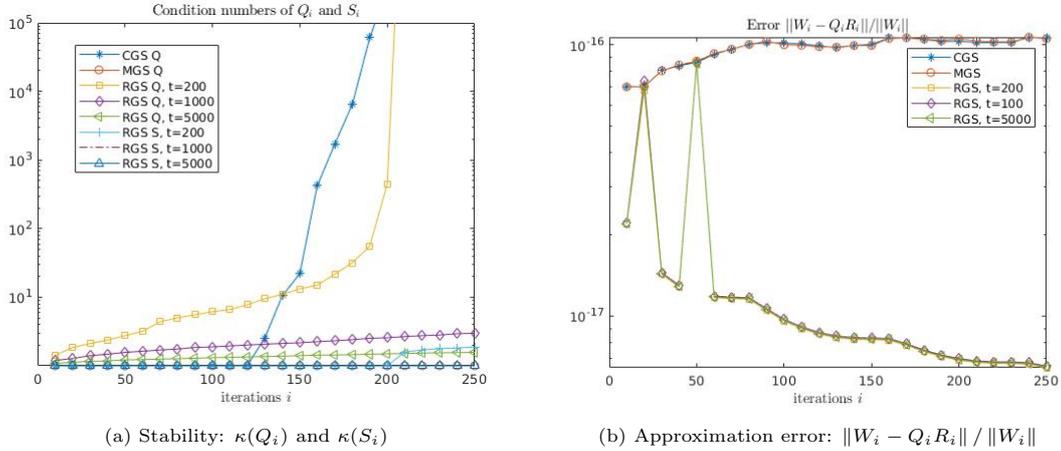


Figure 1: QR decomposition of W_i with respect to GS process varying with the number of columns.

Figure 1 illustrates numerical results of QR decomposition of W with CGS, MGS and RGS choices. In the experiment, there are three different sizes of subspace dimension, $t=200, 1000$, and 5000 , respectively, for sketching. While the condition numbers $\kappa(Q_i)$ obtained by the CGS process increase dramatically for iterations $i \geq 100$, other orthogonal matrices Q_i given by the procedures of MGS or RGS with $t \geq 1000$ are numerically stable. As we noted before, according to [12], ϵ -subspace embedding property in Θ leads to $\kappa(Q_i) = 1 + O(\epsilon)$. Moreover, theoretically to satisfy the condition of the sketching matrix (defined with Rademacher distribution) equipped with the (ϵ, δ, d) oblivious l_2 -subspace embedding such that $t \geq 7.87\epsilon^{-2}(6.9d + \log(1/\delta))$, large t should be chosen for better stability. Hence, as shown in Figure 1a, for not

sufficiently large t , e.g. $t = 200$, the subspace embedding property does not hold so that there exists a large loss of Θ orthogonality. For $t \geq 1000$, $\kappa(Q_i)$ by RGS is quite similar to the trend observed with MGS. On the other hand, the approximation errors are small enough, regardless of the GS process and t . Because the approximation errors only depend on low degree polynomials in m and n (e.g. see (4.1)), whereas the loss of orthogonality relies on the condition number (e.g. see (4.2), (4.3) and (4.4)).

Remark. Here, we employ the Rademacher distribution for the sketching matrix. Although there are other possible random matrices, e.g. based on Gaussian distribution, SRHT, P-SRHT, etc., significant differences in stability or accuracy of approximation do not appear [12]. For further numerical results with W , we refer to [12] that exhibits also a multi-precision variant of RGS.

From the numerical result of Example 1, we were convinced that the RGS algorithm provides well-conditioned orthonormal matrices with high quality of accuracy in approximations. Thus, it allows us to construct proper Krylov subspace basis vectors to solve (ill-conditioned) linear systems.

5.2. Example 2: NACA12

In this numerical experiment, we solve non-symmetric linear systems arising in turbulent CFD simulations of airfoils, namely *NACA airfoils*. These well-known airfoils were developed by the *National Advisory Committee for Aeronautics* (NACA, USA) and *NACA12* is of our particular interest for the test case. For more details, we refer to [28]. Also, please see [29] for PDEs of the turbulence model, *SA-negative*.

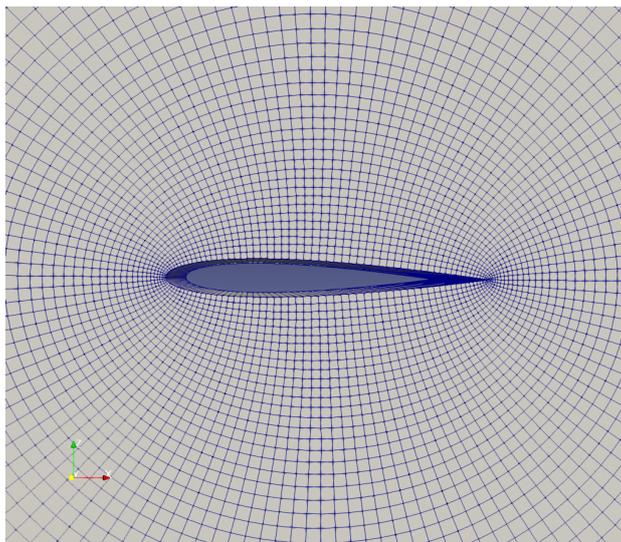


Figure 2: NACA12 mesh.

Consider a steady state CFD problem on the open bounded domain $\Omega \in \mathbb{R}^d$ for $d = 2$. When we denote the vector of conservative variables by $\boldsymbol{\omega}$, one can formulate the differential form of the fundamental equations of fluid dynamics such as continuity, momentum and energy equations, using *Reynolds-averaged Navier-Stokes* (RANS) modelling, as follows:

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{g}, \quad (5.1)$$

where \mathbf{F} is the convective and diffusive fluxes and \mathbf{g} is a source term. For convenience, we can introduce the residual of the space operator \mathcal{R} over Ω such that

$$\frac{\partial \boldsymbol{\omega}}{\partial t} = -\mathcal{R}(\boldsymbol{\omega}),$$

then the steady state solutions can be computed by quasi-Newton’s iterations implemented with the pseudo-transient continuation technique such that

$$\left(\frac{|\Omega|}{n_{\text{CFL}}\Delta t} I + \frac{\partial \mathcal{R}}{\partial \boldsymbol{\omega}}(\boldsymbol{\omega}^j) \right) \Delta \boldsymbol{\omega} = -\mathcal{R}(\boldsymbol{\omega}^j), \quad (5.2)$$

where Δt is the time step, n_{CFL} is the CFL number and $\Delta \boldsymbol{\omega} = \boldsymbol{\omega}^{j+1} - \boldsymbol{\omega}^j$ for j -th Newton iteration. The CFL number is defined by

$$n_{\text{CFL}} = \min \left(\frac{n_{\text{min}}}{\xi^\beta}, n_{\text{max}} \right), \quad \xi = \max \left(\frac{\|\mathcal{R}(\boldsymbol{\omega}^j)\|_{L_2}}{\|\mathcal{R}(\boldsymbol{\omega}^0)\|_{L_2}}, \frac{\|\mathcal{R}(\boldsymbol{\omega}^j)\|_{L_\infty}}{\|\mathcal{R}(\boldsymbol{\omega}^0)\|_{L_\infty}} \right),$$

where n_{min} , n_{max} and β are user-defined parameters. For more details of the governing equations, we refer to [30]. In the following Examples 2 and 3, we consider a fixed single iteration of (5.2). Moreover, using the finite volume schemes, we shall solve the discrete problems $A\boldsymbol{x} = \boldsymbol{b}$ where A and \boldsymbol{x} are the first and second term of the left-hand side of the discretized (5.2), and \boldsymbol{b} is the right-hand side of the discretized (5.2), respectively.

In Example 2, we solve the NACA12 test case of inviscid fluid with Mach number 0.5 and flap angle 0. Using Euler equations and the first order finite volume scheme, the discretized PDE of the SA-negative turbulence model on NACA12 with respect to the structured mesh in Figure 2 results in a non-symmetric linear system $A\boldsymbol{x} = \boldsymbol{b}$ built by the CFD software `e1sA` (ONERA-Safran property) based on finite volume schemes [31]. The matrix $A \in \mathbb{R}^{n \times n}$ for $n = 81,920$, with $nnz(A) = 5,291,223$, is real, non-positive definite, a blockwise structured sparse matrix with a symmetric pattern and not severely ill-conditioned, e.g. $\kappa(A) = O(10^6)$. To solve the linear system problem, we apply *incomplete LU* (ILU0) preconditioner to GMRES methods. We then define the sketching matrix Θ by Rademacher distribution with $t = 1,000$. Hence, we can solve the linear system with the proposed randomized GMRES methods: Algorithm 2 with Propositions 1 and 2, Algorithm 3 with Proposition 3.

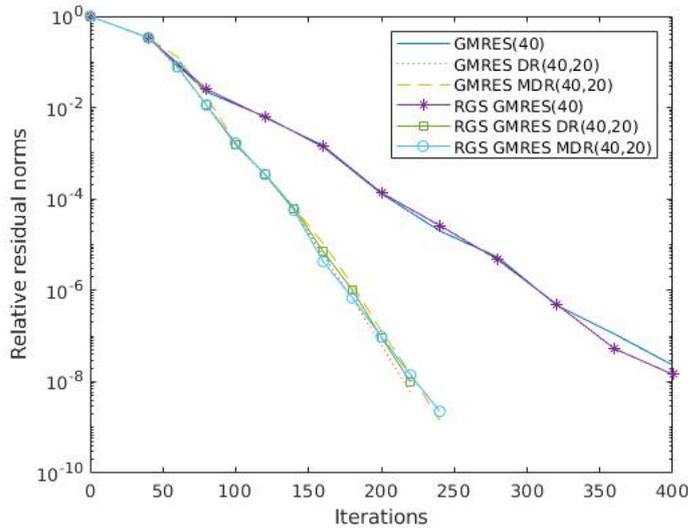


Figure 3: Comparison between the classical GMRES and the randomized GMRES with/without deflated restarts.

Figure 3 shows the relative residual norm $\|\boldsymbol{r}_m\| / \|\boldsymbol{b}\|$ with respect to iteration numbers of each Arnoldi process in GMRES. This numerical result illustrates that the randomized GMRES has the same convergence as the classical GMRES which employing either CGS or MGS. And we can observe that for a choice of 20 vectors to be deflated, both deflation schemes enhance the performance, for example almost twice better than the restart GMRES without deflation.

Remark. As stated before, RGS is beneficial in not only computational costs but also robust stability. However, in Example 2, the matrix A is not so ill-conditioned hence it is not clear to see the advantage of randomized variants in stability as well as convergence, e.g. see Figure 3. Indeed, if a set of basis vectors for the Krylov subspace is well-defined and stable, there is no impact on the convergence analysis of GMRES. Thus, regardless of types of GS, if the basis matrix has a good condition number, e.g. $\kappa(V_m) \approx 1$, the convergence rate will be the same.

For both variants of deflation subspaces, increasing the number of deflated vectors k slightly improves the convergence in Figure 4. Despite no general recommendation for the choice of k , if the number of deflation vectors closes to the restart parameter m , the deflated restarting strategies mostly show the best performance (e.g. see also [6]).

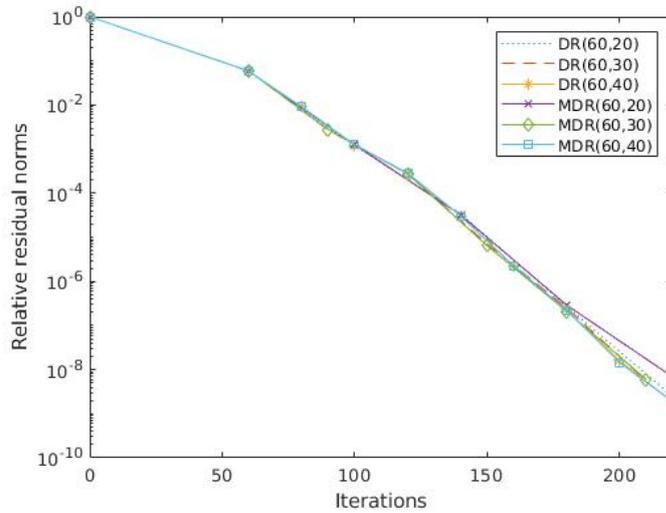


Figure 4: Convergence of RGS-GMRES-DR and MDR with respect to the number of deflated vectors.

Obviously, increasing m improves the convergence of GMRES as it is commonly known that a large restart parameter yields better information in the GMRES residual polynomial, e.g. please see [32]. However, too large restarts may deteriorate the performance and we may also encounter practical issues such as memory requirements so that it is necessary to introduce effective parameter values, for instance, the adaptive strategy [32]. Meanwhile, finding the optimal k is still ambiguous with the goal of minimizing the number of iterations until satisfying the convergence criteria. The key of deflation behind is how much we can disregard eigenvectors/singular vectors associated with eigenvalues/singular values close to the origin. As a result, the deflated Krylov subspaces will have better spectral properties to enhance the convergence of the Krylov subspace methods. Nevertheless, the theoretical analysis of choosing k will be required.

When we solved the example of NACA12, we could not observe significant differences between CGS and RGS in terms of convergence because of the moderate condition number of A . The deflation strategies certainly enhanced the convergent rates. In the next example, we consider similar turbulent models but resulting in ill-conditioned linear systems.

5.3. Example 3: LS89

In this example, we shall solve a steady subsonic flow of a compressible turbulent model (SA-negative). As displayed in Figure 5, we consider LS89 [33, 34] test case with the following unstructured mesh. The LS89 test case is one of the challenging turbine test cases which is frequently used to benchmark CFD tools.

After discretizing the model problem (5.2), e.g. using a second order finite volume scheme on RANS equations, we solve the linear problem of the ill-conditioned system. The corresponding matrix A has

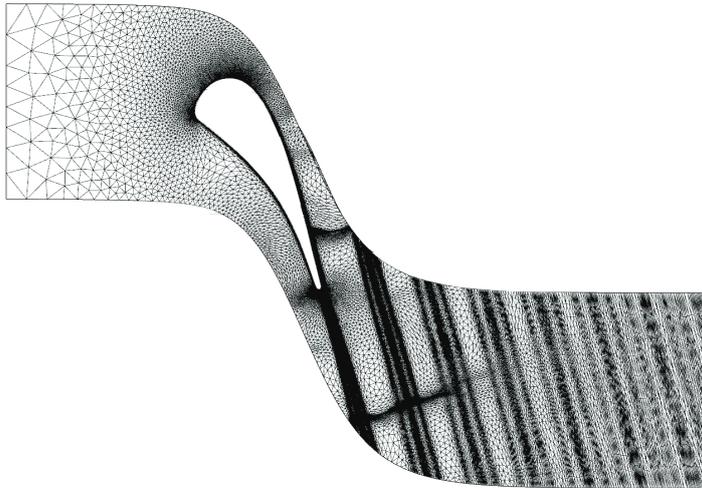


Figure 5: LS89 unstructured mesh.

extremely large condition number such as $O(10^{14})$ with $n = 115,368$ and $nnz(A) = 6,015,042$. Due to the difficulty of solving the linear system, the choice of solvers is of importance. In our test case, we employ inner-outer GMRES with *restricted additive Schwarz* (RAS) preconditioning for inner iterations. More precisely, we want to use the RAS preconditioned GMRES as the outer preconditioner. We then solve the linear system with our proposed GMRES methods varying with the Gram-Schmidt process, deflation strategy, etc. As before, we define the sketching matrix Θ of the Rademacher distribution with $t = 1,000$.

We compare the randomized GMRES methods with the classical variants, e.g. using CGS for orthogonal basis vectors. We set the fixed inner iteration $m_{\text{inner}} = 15$, outer restart $m_{\text{outer}} = 40$ and the number of deflated vectors $k = 15$ if deflated restarts are applied. In RAS preconditioners for inner GMRES, we consider 6 overlapping and 12 subdivisions for the coarse level. Before solving the linear system, we consider the stability of GS algorithms in the flexible Arnoldi process.

Figure 6 illustrates condition numbers of Krylov basis matrices and the approximate errors in flexible Arnoldi iterations, e.g. $\|AZ_i - V_{i+1}H_i\| / \|AZ_i\|$, with respect to GS process. The condition numbers and the approximate errors represent the stability of orthogonal basis matrices and the quality of Arnoldi identity, respectively. As in the previous example, RGS leads to significantly better stability than CGS, e.g. see Figure 6a. Furthermore, while $\kappa(V_i) \approx 10$ for $i \geq 70$ in MGS, the condition numbers of the orthogonal basis matrices by RGS stay at $1 + O(\epsilon)$. We can also observe that RGS results more accurate Arnoldi relation satisfying (4.1) than other GSs in Figure 6b. Due to this instability of Krylov basis matrices from CGS/MGS, FGMRES could degrade numerically.

Next, we solve the linear system of the test case LS89 with respect to GS process and deflation strategy. Figure 7 displays relative residual norms with respect to (outer) iterations. After each cycle of outer FGMRES, we compute a loss of orthogonality of basis vectors. For example, we evaluate the loss of orthogonality by

$$\|I_{m+1} - V_{m+1}^H V_{m+1}\| \quad \text{and} \quad \|I_{m+1} - S_{m+1}^H S_{m+1}\|$$

given by the CGS and RGS algorithms, respectively. Moreover, if deflated restarts are exploited, we additionally determine the loss of orthogonality on the deflated space V_{k+1} or S_{k+1} (*resp.* V_k or S_k) in FGMRES-DR (*resp.* FGMRES-MDR).

While we dealt with the moderately conditioned matrix in the NACA12 example, we need to solve the ill-conditioned system so that poor performance might occur in CGS variants. To be specific, even though the CGS-FGMRES and RGS-FGMRES show similar convergence rates, Figure 7 illustrates that the (harmonic Ritz based) deflation involved with CGS deteriorates the performance. Indeed, to introduce the compact form of \mathbf{c} , e.g. (2.9) and (2.10), sufficient orthogonality of basis vectors of Krylov subspaces is required

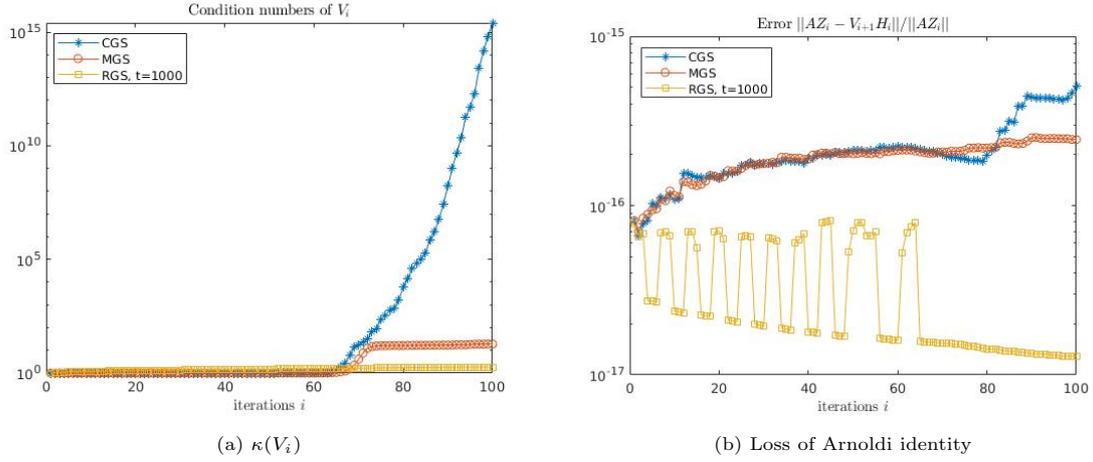


Figure 6: Stability of Arnoldi iterations with respect to GS process varying with the number of columns.

as noted in [7] hence it failed in this test case with the CGS algorithm. As shown in Table 2, the loss of orthogonality in the CGS algorithm is always bigger than that in the RGS algorithm. Hence, we can observe that while the convergence of the deflation of harmonic eigenvectors with CGS algorithm stagnates, RGS-FGMRES-DR improves the performance. In cases of SVD based deflation, we have similar convergence rates between CGS and RGS process but the convergence with the CGS algorithm slightly deteriorated after 4th cycle of restarts. Therefore, RGS-FGMRES-MDR presented the best numerical performance.

Remark. *The orthogonality of the Krylov basis vectors plays an important role in the variants of GMRES. Principally, the GMRES methods find approximate solutions to minimize residual norms along with orthogonal directions. Also, the simplified form of \mathbf{c} , e.g. in Propositions 1- 3, is derived by the orthogonality. More precisely, the compact form of \mathbf{c} in GMRES relies on the orthogonality between the first vector and other vectors, e.g. $\mathbf{v}_1 \perp \mathbf{v}_j, \forall j = 2, \dots, m + 1$. In the same way, the compact form for GMRES-MDR requires the orthogonality between $(k + 1)$ -th vector and other vectors. However, in case of GMRES-DR, we need more orthogonality such that*

$$\mathbf{v}_i \perp \mathbf{v}_j, \quad \forall i = 1, \dots, k + 1, j = 1, \dots, m + 1 \quad \text{and} \quad i \neq j.$$

Therefore, in practice, the numerical performance of GMRES-DR with the compact form is much more sensitive to the loss of orthogonality than those of GMRES and GMRES-MDR.

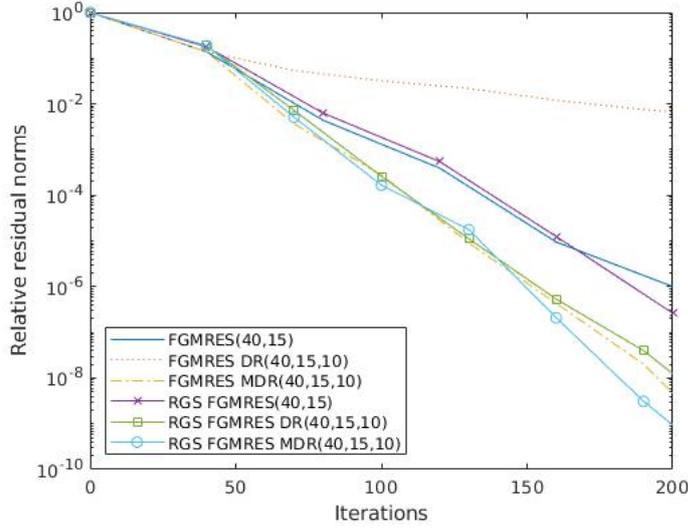


Figure 7: Convergence rates in solving LS89: FGMRES($m_{\text{outer}}, m_{\text{inner}}, k$) with the RAS preconditioning.

# cycle	Loss of orthogonality in CGS				
	No deflation	DR		MDR	
	V_{m+1}	V_{k+1}	V_{m+1}	V_k	V_{m+1}
1st cycle	4.3e-13	-	4.3e-13	-	4.3e-13
2nd cycle	2.3e-13	3.3e-13	1.1e-12	1.3e-13	3.6e-12
3rd cycle	2.8e-13	1.0e-12	2.0e-12	2.3e-13	2.4e-12
4th cycle	4.0e-13	1.6e-12	4.3e-12	5.0e-13	7.4e-12
5th cycle	1.8e-13	3.8e-12	2.7e-11	7.9e-13	1.0e-11
6th cycle	-	1.9e-11	3.7e-11	1.1e-12	8.8e-12
7th cycle	-	3.4e-11	6.6e-11	1.5e-12	1.1e-11

# cycle	Loss of orthogonality in RGS				
	No deflation	DR		MDR	
	S_{m+1}	S_{k+1}	S_{m+1}	S_k	S_{m+1}
1st cycle	5.8e-15	-	5.3e-15	-	5.1e-15
2nd cycle	5.0e-15	2.2e-15	1.0e-14	2.2e-15	6.7e-13
3rd cycle	5.6e-15	3.5e-15	1.0e-14	2.0e-14	3.7e-13
4th cycle	5.6e-15	3.5e-15	1.1e-14	4.1e-14	4.1e-13
5th cycle	5.7e-15	4.3e-15	1.1e-14	4.9e-14	8.1e-13
6th cycle	-	4.0e-15	1.1e-14	6.9e-14	6.5e-13
7th cycle	-	4.0e-15	1.1e-14	7.2e-14	3.0e-13

Table 2: Loss of orthogonality for each cycle.

6. Conclusion

We have presented new FGMRES methods based on a randomized Gram-Schmidt algorithm that has a beneficial effect on computational costs. Compared to other existing Gram-Schmidt processes, it requires half the computational complexity and has comparable stability. We have also introduced two ways of deflated restarting for faster convergence. By removing the spectrum clustered near the origin or small singular values, the randomized GMRES becomes more robust and efficient. Numerical experiments have shown that the randomized FGMRES with deflated restarts had the best performance due to the stable set of basis vectors. Furthermore, we have introduced reorthogonalized RGS (RGS2) to impose l_2 orthogonality and avoid loss of orthogonality. While the numerical examples of RGS2 are not given in this paper, we plan to present more details on its benefits in future studies.

Acknowledgement

This work is supported by DGAC (*Direction Générale de l'Aviation Civile*) in the frame of the SONICE project.

References

- [1] Y. Saad, M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM Journal on scientific and statistical computing* 7 (1986) 856–869.
- [2] C. C. Paige, M. A. Saunders, Solution of sparse indefinite systems of linear equations, *SIAM journal on numerical analysis* 12 (1975) 617–629.
- [3] N. M. Nachtigal, S. C. Reddy, L. N. Trefethen, How fast are nonsymmetric matrix iterations?, *SIAM Journal on Matrix Analysis and Applications* 13 (1992) 778–795.
- [4] M. Embree, How descriptive are gmres convergence bounds?, *arXiv preprint arXiv:2209.01231* (2022).
- [5] R. B. Morgan, GMRES with deflated restarting, *SIAM Journal on Scientific Computing* 24 (2002) 20–37.
- [6] L. Giraud, S. Gratton, X. Pinel, X. Vasseur, Flexible GMRES with deflated restarting, *SIAM Journal on Scientific Computing* 32 (2010) 1858–1878.
- [7] S. Röllin, W. Fichtner, Improving the accuracy of GMRES with deflated restarting, *SIAM Journal on Scientific Computing* 30 (2008) 232–245.
- [8] V. Simoncini, On the convergence of restarted Krylov subspace methods, *SIAM Journal on Matrix Analysis and Applications* 22 (2000) 430–452.
- [9] M. L. Parks, E. De Sturler, G. Mackey, D. D. Johnson, S. Maiti, Recycling Krylov subspaces for sequences of linear systems, *SIAM Journal on Scientific Computing* 28 (2006) 1651–1674.
- [10] H. A. Daas, L. Grigori, P. Hénon, P. Ricoux, Recycling Krylov subspaces and truncating deflation subspaces for solving sequence of linear systems, *ACM Transactions on Mathematical Software (TOMS)* 47 (2021) 1–30.
- [11] L. M. Carvalho, S. Gratton, R. Lago, X. Vasseur, A flexible generalized conjugate residual method with inner orthogonalization and deflated restarting, *SIAM Journal on Matrix Analysis and Applications* 32 (2011) 1212–1235.
- [12] O. Balabanov, L. Grigori, Randomized Gram–Schmidt process with application to GMRES, *SIAM Journal on Scientific Computing* 44 (2022) A1450–A1474.
- [13] O. Balabanov, L. Grigori, Randomized block Gram–Schmidt process for solution of linear systems and eigenvalue problems, *arXiv preprint arXiv:2111.14641* (2021).
- [14] Y. Nakatsukasa, J. A. Tropp, Fast & accurate randomized algorithms for linear systems and eigenvalue problems, *arXiv preprint arXiv:2111.00113* (2021).
- [15] Å. Björck, *Numerical methods for least squares problems*, SIAM, 1996.
- [16] N. J. Higham, *Accuracy and stability of numerical algorithms*, SIAM, 2002.
- [17] L. Giraud, J. Langou, M. Rozložník, The loss of orthogonality in the Gram–Schmidt orthogonalization process, *Computers & Mathematics with Applications* 50 (2005) 1069–1075.
- [18] D. Achlioptas, Database-friendly random projections: Johnson–Lindenstrauss with binary coins, *Journal of computer and System Sciences* 66 (2003) 671–687.
- [19] N. Halko, P.-G. Martinsson, J. A. Tropp, Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions, *SIAM review* 53 (2011) 217–288.
- [20] D. P. Woodruff, Sketching as a tool for numerical linear algebra, *Foundations and Trends in Theoretical Computer Science* 10 (2014) 1–157.
- [21] O. Balabanov, A. Nouy, Randomized linear algebra for model reduction. Part I: Galerkin methods and error estimation, *Advances in Computational Mathematics* 45 (2019) 2969–3019.
- [22] Y. Saad, A flexible inner-outer preconditioned GMRES algorithm, *SIAM Journal on Scientific Computing* 14 (1993) 461–469.
- [23] C. C. Paige, B. N. Parlett, H. A. Van der Vorst, Approximate solutions and eigenvalue bounds from Krylov subspaces, *Numerical linear algebra with applications* 2 (1995) 115–133.
- [24] N. N. Abdelmalek, Round off error analysis for Gram–Schmidt method and solution of linear least squares problems, *BIT Numerical Mathematics* 11 (1971) 345–367.
- [25] L. Giraud, J. Langou, M. Rozložník, J. v. d. Eshof, Rounding error analysis of the classical Gram–Schmidt orthogonalization process, *Numerische Mathematik* 101 (2005) 87–100.
- [26] Å. Björck, Solving linear least squares problems by Gram–Schmidt orthogonalization, *BIT Numerical Mathematics* 7 (1967) 1–21.
- [27] C. F. Van Loan, G. Golub, *Matrix computations* (Johns Hopkins studies in mathematical sciences), *Matrix Computations* (1996).
- [28] E. N. Jacobs, K. E. Ward, R. M. Pinkerton, The characteristics of 78 related airfoil section from tests in the variable-density wind tunnel, *Technical Report*, NASA, 1933.
- [29] S. R. Allmaras, F. T. Johnson, Modifications and clarifications for the implementation of the Spalart–Allmaras turbulence model, in: *Seventh international conference on computational fluid dynamics (ICCFD7)*, volume 1902, Big Island, HI, 2012, pp. 1–11.
- [30] A. Crivellini, F. Bassi, An implicit matrix-free discontinuous Galerkin solver for viscous and turbulent aerodynamic simulations, *Computers & fluids* 50 (2011) 81–93.

- [31] L. Cambier, S. Heib, S. Plot, The ONERA elsA CFD software : input from research and feedback from industry, *Mech. Ind.* (2013) .
- [32] A. H. Baker, E. R. Jessup, T. V. Kolev, A simple strategy for varying the restart parameter in GMRES(m), *Journal of computational and applied mathematics* 230 (2009) 751–761.
- [33] T. Arts, M. L. De Rouvroit, Aero-thermal performance of a two dimensional highly loaded transonic turbine nozzle guide vane: A test case for inviscid and viscous flow computations, in: *Turbo Expo: Power for Land, Sea, and Air*, volume 79047, American Society of Mechanical Engineers, 1990, p. V001T01A106.
- [34] T. Arts, M. L. De Rouvroit, A. Rutherford, Aero-thermal investigation of a highly loaded transonic linear turbine guide vane cascade, Technical Report, von Karman Institute for Fluid Dynamics, 1990.