The 8th European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS Congress 2022
5-–9 June 2022, Oslo, Norway

AN ACCELERATED DEFLATION-BASED PRECONDITIONER FOR PARAMETERIC SYSTEMS BASED ON SUBSPACE RECYCLING

DIONYSIOS PANAGIOTOPOULOS[1,3], WIM DESMET[1,3] AND ELKE DECKERS[2,3]

[1] Department of Mechanical Engineering
KU Leuven
Celestijnenlaan 300, 3001 Leuven, Belgium

[2] Department of Mechanical Engineering
KU Leuven
Wetenschapspark 27, 3590 Diepenbeek, Belgium

[3] DMMS lab
Flanders Make
Belgium

Key words: Preconditioner, Deflation, Krylov Subspace Recycling, AKR algorithm

Abstract. Krylov subspace recycling [1] is often deployed to accelerate the iterative solution of sequences of linear systems. Such approaches reuse a continuously updated deflation subspace to reach a converged solution within a low number of iterations. This procedure is justified for problems that describe gradually evolving phenomena, such as crack propagation, and thus involve a sequence of systems that are not simultaneously available. However considering parametric systems, these techniques might induce an unnecessary overhead cost. Specifically, by constantly updating the recycled subspace a new projection on the newly constructed subspace needs to be operated for each new system, inducing a cost that scales with $\mathcal{O}(\ell \times N^2)$ for dense systems, where N is the size of the system and ℓ is the size of the employed recycled basis. In that context, this work proposes an accelerated recycling procedure for parametric systems that is inspired by the Galerkin Model Order Reduction strategy and employs an offline – online operation splitting. In the offline part, the subspace to be recycled is constructed via an Automatic Krylov subspaces Recycling algorithm (AKR) [2] and the parametric system is projected on the subspace to yield a Reduced Order Model (ROM). Then, in the online part the construction of the deflation preconditioner only requires employing the ROM and as a result the cost of constructing the preconditioner is reduced to $\mathcal{O}(\ell^2)$. The proposed procedure is tested on a randomly parametrized linear system and is compared the non-deflated GMRES algorithm and a conventional recycling strategy presented in [11].

1 INTRODUCTION

The paper is concerned with the acceleration of the solution of parametric problems of the form

$$\mathbf{A}(\omega)\mathbf{x}(\omega) = \mathbf{b}(\omega), \quad \omega \in \Psi \tag{1}$$

where $\mathbf{A} : \Psi \to \mathbb{C}^{N\times N}$ and $\mathbf{b}, \mathbf{x} : \Psi \to \mathbb{C}^N$ and Ψ being a parameter interval. Such problems often need to be solved for multiple values of the parameter $\omega \in \Psi$, i.e. in a multi-resolution manner. Typically, a single linear system for a specific value of ω can be solved by employing an iterative solver, such as the conjugate gradient method [3], GMRES [4] and BiCG [5]. Such methods involve the expansion of some kind of Krylov subspaces and their efficiency depends on the distribution of the eigenvalues of the considered system. Exploiting this fact, it is possible to devise deflation and augmentation strategies to accelerate the iterative convergence to the solution [6, 7]. Specifically, a great advantage can be offered by deflating the system with eigenvectors or approximated eigenvectors (Ritz vectors) as demonstrated in [6] with FGMRES and [7] with GMRES-DR. The same approach can be employed for systems with multiple right hand-sides (RHS) as proposed in [8]. The idea of recycling information among different linear systems is exploited in [9] for solving sequences of varying RHSs. Finally, additional techniques such as the ones reported in [1, 10, 11] made the step to extend this approach for slowly varying systems and were based on the idea of recycling information from the a current system configuration to a subsequent one, arising due to a slow evolution of a physical phenomenon, such as a crack propagation evolution problem [1].

Moving in the same direction, the same concept has been recently exploited for the acceleration of the solution of parametric linear systems such as the ones arising for the deployment of the Boundary Element Method (BEM) within acoustic problems [12]. Most existing algorithms, such as [11] and [1], employ a sequential treatment of systems updating the recycled space $\mathscr{W} = \text{span}\{\mathbf{W}\}$ "on the fly". However, this approach can incur a considerable cost, especially when attempting to solve non-sparse systems, such as the ones arising in the BEM. Specifically, employing such recycling strategies requires the computation of a new projector matrix $\mathbf{P}$ for each new system. However, to compute such a matrix requires to project the system matrix $\mathbf{A}$ on $\mathscr{W}$ and thus is accompanied by the cost of multiplying matrix $\mathbf{A}$ with the constantly evolving basis $\mathbf{W}$. As proposed in this work, this cost can be mitigated for the case of affine parametric systems, where this product can be precomputed.

Nevertheless, to precompute this product, it is essential that a high quality recycling basis $\mathbf{W}$ is available in advance. Employing a deflation strategy with Ritz vectors, such a basis needs to be able to describe their variations, occurring due to the parameter dependence with ω, as denoted in (1). To construct such a basis, it is possible to deploy a sampling of Krylov subspaces across the parametric domain, as proposed in [2] with the Automatic Krylov subspaces Recycling (AKR) algorithm. In brief, this algorithm constructs a global solution basis by adaptively sampling Krylov subspaces and guarantees minimal sampling of the parametric domain, therefore facilitating potentially the offline stage of a respective Model Order Reduction scheme.

In that context, within this paper an accelerated deflation-based preconditioner for parametric linear systems is employed, by splitting operations to an offline and an online part and by leveraging a high quality deflation basis constructed with the AKR algorithm. In the offline part the AKR algorithm is deployed and the resulting basis $\mathbf{W}$, with $\mathscr{W} = \text{span}\{\mathbf{W}\}$, is used within a Galerkin projection scheme to yield a reduced order model (ROM). However, instead of employing this ROM to compute an approximate solution, this is only used within the construction of

a projector $\mathbf{P}$ onto $\mathscr{W}$, therefore accelerating the construction of a high quality deflation-based preconditioner.

The paper is organized as follows. In section 2 the basic recycling strategy employed in [1] and [11] is introduced. In section 3 the accelerated recycling scheme is presented along with the AKR algorithm that allows the construction of a high quality deflation basis. Then, in section 4 the proposed technique is assessed on a random parametrization example. Finally, section 5 summarizes and concludes the paper.

2 A BASIC KRYLOV SUBSPACE RECYCLING SCHEME

The solution of linear systems is typically accelerated by employing iterative solvers. In this paper the GMRES (Generalized Minimal Residual) method is employed as this is a method suitable for solving generic problems. The idea of the method is based on the fact that it minimizes the residual within each iteration solving a small least square problem. The derivation and more details for this method can be found in the seminal work of Saad [4].

2.1 A deflation based preconditioning GMRES

Aiming to solve a linear system of equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{2}$$

where $\mathbf{A} \in \mathbb{C}^{N\times N}$ and $\mathbf{b}, \mathbf{x} \in \mathbb{C}^N$, the GMRES can significantly speed up its solution by iteratively approximating it. However, in order to solve problematic cases, where a large number of iterations is required to yield an accurate solution, this method is typically combined with a suitable preconditioner. A large family of such preconditioners is based on the calculation of approximate eigenvectors and their use to deflate the solution space. In this section, a deflation framework is presented, following [13], which can be fit into different iterative solvers, e.g. within GMRES [7], GCR [1] and CG [14].

Assuming a deflation subspace $\mathscr{W} \subset \mathbb{C}^{N\times N}$ of fixed dimension ℓ, which is also chosen as the constraint subspace, and the iteratively built subspace $\mathscr{V}_m, \tilde{\mathscr{V}}_m$, where $\dim(\mathscr{V}_m) = \dim(\tilde{\mathscr{V}}_m) - 1 = m$. The goal of the deflated iterative method is to find an approximate solution

$$\hat{\mathbf{x}}_m = \mathbf{x}_0 + \mathbf{s} + \mathbf{t}_m, \quad \mathbf{s} \in \mathscr{W}, \mathbf{t}_m \in \mathscr{V}_m, \tag{3}$$

for the system (2), such that the residual

$$\mathbf{r}_m := \mathbf{b} - \mathbf{A}\hat{\mathbf{x}}_m \perp (\mathscr{W} + \tilde{\mathscr{V}}_m). \tag{4}$$

In this paper it is assumed that $\mathbf{x}_0 = 0$ and $\tilde{\mathscr{V}}_m = \mathbf{A}\mathscr{V}_m$ resulting in the following formulas in a Galerkin projection onto $\mathscr{W}$ and an oblique projection onto $\mathscr{V}_m$ and constraint subspace $\tilde{\mathscr{V}}_m$. The subspaces $\mathscr{W}, \mathscr{V}_m$ are spanned by the bases $\mathbf{W} \in \mathbb{C}^{N\times\ell}$ and $\mathbf{V}_m \in \mathbb{C}^{N\times m}$ respectively. Having $\mathbf{W}$, the corresponding orthogonal projector onto $\mathscr{W}$ can be defined as

$$\mathbf{P} = \mathbf{W}(\mathbf{W}^*\mathbf{A}\mathbf{W})^{-1}\mathbf{W}^*\mathbf{A}, \tag{5}$$

along with the sibling projector onto $\mathbf{A}\mathscr{W}$,

$$\mathbf{Q} = \mathbf{A}\mathbf{W}(\mathbf{W}^*\mathbf{A}\mathbf{W})^{-1}\mathbf{W}^*. \tag{6}$$

Algorithm 1 Deflated GMRES

1: Input: system $\mathbf{A} \in \mathbb{C}^{N\times N}, \mathbf{b} \in \mathbb{C}^N$, initial guess $\mathbf{x}_0 \in \mathbb{C}^N$, residual tolerance $r_{\text{tol}} \in \mathbb{R}$ and deflation basis $\mathbf{W}$;
2: $\mathbf{s} = \mathbf{P}\mathbf{x} = \mathbf{W}(\mathbf{W}^*\mathbf{A}\mathbf{W})^{-1}\mathbf{W}^*\mathbf{b}$
3: $\mathbf{r}_0 \leftarrow \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{s})$; $\quad \mathbf{v}_1 \leftarrow \frac{\mathbf{r}_0}{\|\mathbf{r}_0\|}$;
4: $\mathbf{V}_1 \leftarrow [\mathbf{v}_1]$; $\quad m \leftarrow 1$;
5: **while** $\|\mathbf{r}_{m-1}\| \geq r_{\text{tol}}$ **do**
6: $\quad \mathbf{v}_{m+1} \leftarrow (\mathbf{I} - \mathbf{Q})\mathbf{A}\mathbf{v}_m$; $\quad \mathbf{v}_{m+1} \leftarrow \frac{\mathbf{v}_{m+1}}{\|\mathbf{v}_{m+1}\|}$;
7: $\quad$ **for** $p \leftarrow 1 : m$ **do**
8: $\quad\quad h_{p,m} \leftarrow \mathbf{v}_p^H \mathbf{v}_{m+1}$; $\quad \mathbf{v}_{m+1} \leftarrow \mathbf{v}_{m+1} - h_{p,m}\mathbf{v}_p$;
9: $\quad$ **end for**
10: $\quad h_{m+1,m} \leftarrow \|\mathbf{v}_{m+1}\|$; $\quad \mathbf{v}_{m+1} \leftarrow \frac{\mathbf{v}_{m+1}}{\|\mathbf{v}_{m+1}\|}$; $\quad \mathbf{V}_{m+1} \leftarrow [\mathbf{V}_m \quad \mathbf{v}_{m+1}]$;
11: $\quad$ Find $\mathbf{z}_m \in \mathbb{C}_m$ such that $\min \|\|\mathbf{r}_0\|\mathbf{e}_1 - \hat{\mathbf{H}}_m\mathbf{z}_m\|$
12: $\quad \hat{\mathbf{x}}_m \leftarrow \mathbf{x}_0 + \mathbf{V}_m\mathbf{z}_m$;
13: $\quad \mathbf{r}_m \leftarrow \mathbf{r}_0 - (\mathbf{I} - \mathbf{Q})\mathbf{A}\hat{\mathbf{x}}_m$; $\quad m \leftarrow m + 1$;
14: **end while**
15: Output: Approximation basis $\mathbf{V} \in \mathbb{C}^{N\times m-1}$ and solution approximation $\hat{\mathbf{x}}_{m-1} \in \mathbb{C}^N$

The two projectors are linked by $\mathbf{Q}\mathbf{A} = \mathbf{A}\mathbf{P}$. Having defined the pair of orthogonal projectors, the deflation preconditioning strategy proceeds by splitting the solution into the part $\mathbf{s} \in \mathscr{W}$ and a part $\mathbf{t} \perp \mathscr{W}$, which can be expressed as

$$\mathbf{x} = \mathbf{P}\mathbf{x} + (\mathbf{I} - \mathbf{P})\mathbf{x} = \mathbf{s} + \mathbf{t}, \tag{7}$$

where $\mathbf{I}$ is the identity matrix. Combining equations (2) and (5), the contribution of $\mathscr{W}$ for the solution $\mathbf{x}$ can be computed by $\mathbf{s} = \mathbf{P}\mathbf{x} = \mathbf{W}(\mathbf{W}^*\mathbf{A}\mathbf{W})^{-1}\mathbf{W}^*\mathbf{b}$. After manipulating the system, the contribution $\mathbf{t}$ can be calculated by

$$(\mathbf{I} - \mathbf{Q})\mathbf{A}\mathbf{t} = (\mathbf{I} - \mathbf{Q})\mathbf{b}, \tag{8}$$

which is the deflated system from subspace $\mathscr{W}$. The deflated version of the GMRES method is given in Algorithm 1. The steps 6-13 of the algorithm correspond to a conventional GMRES procedure considering the deflated system $(\mathbf{I} - \mathbf{Q})\mathbf{A}$.

2.2 Recycling of Krylov subspaces for sequences of linear systems

Deflation strategies were initially used to accelerate the solution of a single system, as in equation (2). However, this framework was generalized in [1] and [10, 15] allowing the recycling of subspaces among different linear systems, such as the successive solution of a sequence of linear systems as in equation (1). Typically, within such techniques the recycled information is updated for each system of the given sequence. For example, in [11] it is proposed to construct the recycled subspace for the system $i+1$ by employing all Arnoldi vectors used for the solution of the system i.

The procedure for recycling subspaces for a sequence of linear systems is outlined in Algorithm 2. Regardless of the updating scheme used for the construction of the deflation basis $\mathbf{W}$ for each

Algorithm 2 Recycling of Krylov subspaces for sequences of linear systems.

1: Input: sequence of linear systems $\mathbf{A}^{(i)} \in \mathbb{C}^{N\times N}, \mathbf{b}^{(i)} \in \mathbb{C}^N, i = 1, \ldots, L_{\max}$, initial deflation basis $\mathbf{W}$.
2: **for** $i \leftarrow 1 : L_{\max}$ **do**
3: Solve $\mathbf{A}^{(i)}\mathbf{x}^{(i)} = \mathbf{b}^{(i)}$ with Deflated GMRES (Algorithm 1)
4: Update $\mathbf{W}$ as in [11] or [1]
5: **end for**
6: Output: Deflation basis $\mathbf{W}$ and solution approximations $\mathbf{x}^{(i)} \in \mathbb{C}^N, i = 1, \ldots, L_{\max}$

system, as stated in step 4, such a basis needs to be updated for each new system. Although such a basis might prove highly efficient, from the algorithmic perspective it results into a inefficient procedure, as a new projector $\mathbf{P}$ needs to be constructed for each system. The assembly of such a projector can become quite time consuming as it involves the computation of the product $\mathbf{AW}$.

3 AN ACCELERATED RECYCLING STRATEGY

As indicated in section 2, the cost of employing a deflation basis can often hinder the deployment of such a technique. In this section, an accelerated recycling strategy is proposed for parametric systems of the form (1). Exploiting the parameter dependency in (1) it is possible to cut down the cost of the deflation framework presented in section 2.1 by shifting the most computationally intensive operations of creating the projector matrix $\mathbf{P}$ onto the deflation subspace $\mathscr{W}$ offline. However, in order to enable the splitting of operations, the deflation basis $\mathbf{W} \in \mathbb{C}^{N\times\ell}$ needs to be available upfront. For that reason, the AKR algorithm [2, 16] is employed in the offline stage of the technique. Nevertheless, the produced basis is not employed as a global solution basis, but rather as a global deflation basis.

3.1 An accelerated recycling strategy for parametric systems

In this section an accelerated recycling framework is provided for the deflation of parametric systems of the form (1). Assuming a deflation subspace $\mathscr{W}$ and the family of parameter dependent subspaces $\mathscr{V}_{m(\omega)}(\omega)$, the goal is to provide an approximated solution

$$\hat{\mathbf{x}}_m(\omega) = \mathbf{x}_0(\omega) + \mathbf{s}(\omega) + \mathbf{t}_m(\omega), \quad \mathbf{s}(\omega) \in \mathscr{W}, \mathbf{t}_m(\omega) \in \mathscr{V}_{m(\omega)}(\omega), \tag{9}$$

of the system (1) such that

$$\mathbf{r}_m(\omega) := \mathbf{b}(\omega) - \mathbf{A}(\omega)\hat{\mathbf{x}}_m(\omega) \perp (\mathscr{W} + \tilde{\mathscr{V}}_{m(\omega)}(\omega)). \tag{10}$$

for each value of $\omega \in \Psi$. Accelerating the deflation strategy in Algorithm 1, the parametric description of the system (1) is taken into account. For the sake of simplicity, the parametric dependency of the system is considered affine, i.e. it can be expressed as

$$\mathbf{A}(\omega) = \sum_{i=1}^{M_{\max}} \mathbf{A}_i f_i(\omega), \quad \mathbf{b}(\omega) = \sum_{i=1}^{q_{\max}} \mathbf{b}_i g_i(\omega), \quad \omega \in \Psi, \tag{11}$$

Algorithm 3 Accelerated recycling scheme for parametric systems

1: Input: system $\mathbf{A}_i \in \mathbb{C}^{N\times N}, \mathbf{b}_i \in \mathbb{C}^N$, residual tolerance $r_{\text{tol}} \in \mathbb{R}$ and deflation basis $\mathbf{W}$;
2: Obtain $\mathbf{A}_{i,\text{red}}, \mathbf{b}_{i,\text{red}}$ through a Galerkin projection of $\mathbf{A}_i, \mathbf{b}_i$ on $\mathbf{W}$;
3: **for** all $\omega \in \Phi$ **do**
4: Assemble $\mathbf{A}_{\text{red}}(\omega)$;
5: Calculate $\mathbf{s} \in \mathscr{W}$ by $\mathbf{s} = \mathbf{W}(\mathbf{A}_{\text{red}}(\omega))^{-1}\mathbf{W}^*\mathbf{b}$;
6: Continue with lines 3-13 of Algorithm 1 employing $\mathbf{A}_{\text{red}}(\omega)$ in operations containing $\mathbf{P}, \mathbf{Q}$;
7: **end for**
8: Output: Approximated solution $\hat{\mathbf{x}}(\omega)$

where $\mathbf{A}_i \in \mathbb{C}^{N\times N}$, $\mathbf{b}_i \in \mathbb{C}^N$ and $f_i, g_i : \Psi \to \mathbb{C}$. Additionally, the basis $\mathbf{W}$ is assumed to be available upfront and as such it does not need to be updated for the different values of ω.

Constructing $\mathbf{W}$ upfront, it is possible to follow a Model Order Reduction (MOR) scheme to split operations into an offline and an online part. The offline part contains the most time consuming operations (related with N), where a Reduced Order Model (ROM) is constructed. Then, in the online part most of the operations demonstrate an algorithmic complexity that scales favourably with the size of the ROM $\ell := \text{rank}(\mathbf{W})$. The method proposed in this paper, instead of attempting to provide an approximate solution $\hat{\mathbf{x}}(\omega) \in \mathscr{W}$, it employs the basis $\mathbf{W}$ in a deflation framework.

Specifically, in the offline part of the proposed method a Galerkin projection, i.e. $\mathscr{W} \equiv \tilde{\mathscr{W}}$, is deployed to the parametric system (1). An approximation $\mathbf{s}$ of the true solution $\mathbf{x}$ is expressed as a linear combination of the basis vectors of $\mathbf{W}$

$$\mathbf{x}(\omega) \approx \mathbf{s}(\omega) = \mathbf{x}_0(\omega) + \mathbf{W}\mathbf{y}(\omega), \tag{12}$$

where $\mathbf{y} : \Psi \to \mathbb{C}^\ell$ includes the participation coefficients of each basis vector of $\mathbf{W}$ and $\mathbf{x}_0(\omega)$ an initial guess for the solution. Since, often in MOR the initial guess is elected as $\mathbf{x}_0(\omega) = 0$, in the following expressions $\mathbf{x}_0(\omega)$ is omitted. Substituting (12) in (1) and multiplying from the left with the conjugate transpose of the deflation basis $\mathbf{W}^H$, yields the reduced system

$$\mathbf{W}^H\mathbf{A}(\omega)\mathbf{W}\mathbf{y}(\omega) = \mathbf{W}^H\mathbf{b}(\omega), \tag{13}$$

which can be assembled efficiently, i.e. demonstrating an algorithmic efficiency that scales with ℓ^2. Given that $\mathbf{A}(\omega), \mathbf{b}(\omega)$ have an affine structure as in (11), inserting the affine expressions in (13) yields

$$\sum_{i=0}^{M_{\max}} f_i(\omega)\mathbf{A}_{i,\text{red}}\mathbf{y}(\omega) = \sum_{q=0}^{q_{\max}} g_q(\omega)\mathbf{b}_{q,\text{red}}, \tag{14}$$

where $\mathbf{A}_{i,\text{red}} \in \mathbb{C}^{\ell\times\ell}$ and $\mathbf{b}_{q,\text{red}} \in \mathbb{C}^\ell$ are the coefficients of the reduced affine system. Incorporating this strategy within the deployment of Algorithm 1 in case of parametric systems can significantly speed up the procedure of solving a sequence of $L = |\Phi|$ linear systems assuming that $L \gg M_{\max}$.

In Algorithm 3 the proposed accelerated procedure for the sequential solution of the linear systems for all $\omega \in \Phi$ is outlined. A ROM is built in line 2, which is subsequently employed within

all operations related to the orthogonal projectors $\mathbf{P}$ and $\mathbf{Q}$. Having efficiently constructed $\mathbf{P}$ and $\mathbf{Q}$, the remaining lines of Algorithm 1 are executed without any changes for all $\omega \in \Phi$.

3.2 An Automatic Krylov Recycling algorithm for the construction of a global deflation basis

The above introduced strategy is based on the assumption that a high quality deflation basis is available beforehand. To construct such a basis the Automatic Krylov subspaces Recycling (AKR) [2] and its generalization for multi-parameter systems [16] can be employed. These algorithms construct high quality global solution bases that can be used in the offline stages of the corresponding MOR and pMOR strategies. Since the resulting bases sample Krylov subspaces from systems arising from different parameter configurations, they are rich in spectral content, thus capturing the variations of the spectrum across the entire parameter range.

The AKR algorithm constructs an appropriate ordered set of parameter values Ω of which the subspaces $\mathscr{K}_m(\omega), \omega \in \Omega$ are recycled and a sufficiently high dimension of the subspaces to be recycled, the construction of a global basis is possible by combining the Krylov subspaces as

$$\mathscr{K}^{tot} = \mathscr{K}^{\omega_1}_{m(\omega_1)} \cup \mathscr{K}^{\omega_2}_{m(\omega_2)} \cup \cdots \cup \mathscr{K}^{\omega_S}_{m(\omega_S)}, \tag{15}$$

where $S := |\Omega|$ and $m(\omega_j)$ with $j = 1, \ldots, S$ the dimension of the respective Krylov subspaces for each $\omega_j \in \Omega$. The spacing between two consecutive values of Ω as well as the respective subspace dimension $m(\omega_j)$ are adaptively selected by ensuring a predefined residual threshold for each $\omega_m := (\omega_j + \omega_{j+1})/2)$, where $\omega_j, \omega_{j+1} \in \Omega$. Having constructed the basis $\mathbf{W}$, where $\mathscr{K}^{tot} \equiv \mathscr{W} = \text{span}\{\mathbf{W}\}$, it is straightforward to obtain the reduced order system of equation (14), by sequentially projecting all $\mathbf{A}_i$ onto $\mathscr{W}$.

4 NUMERICAL ASSESSMENT

In this section the performance of the proposed technique is investigated in comparison with the non-deflated GMRES technique and a conventional recycling technique (TRKS) given in [11]. To emphasize on the general applicability of the technique, it is deployed to solve a randomly parametrized multiresolution problem, not stemming from any specific application.

4.1 Construction of the dense system

The first example constitutes a random dense system of the form (1) with a given single parameter affine dependency and predefined eigenvalue distribution. The parametrized matrix can be expressed as

$$\mathbf{A}(\omega) = \mathbf{A}_0 + \omega \mathbf{A}_1, \tag{16}$$

where $\mathbf{A} : \Psi \to \mathbb{C}^{N \times N}$, with $\Psi := [0, 10]$ and $N := 5000$. Since the right-hand side vector $\mathbf{b} : \Psi \to \mathbb{C}^N$ does not significantly influence the convergence of iterative techniques, without loss of generality it can be selected as constant, namely $\mathbf{b}(\omega) := [1 \quad 1 \ldots 1]^T$. The affine coefficients $\mathbf{A}_0$ and $\mathbf{A}_1$ are constructed as

$$\mathbf{A}_\text{i} := \text{rand}(5000) + \mathscr{j} \cdot \text{rand}(5000), \tag{17}$$

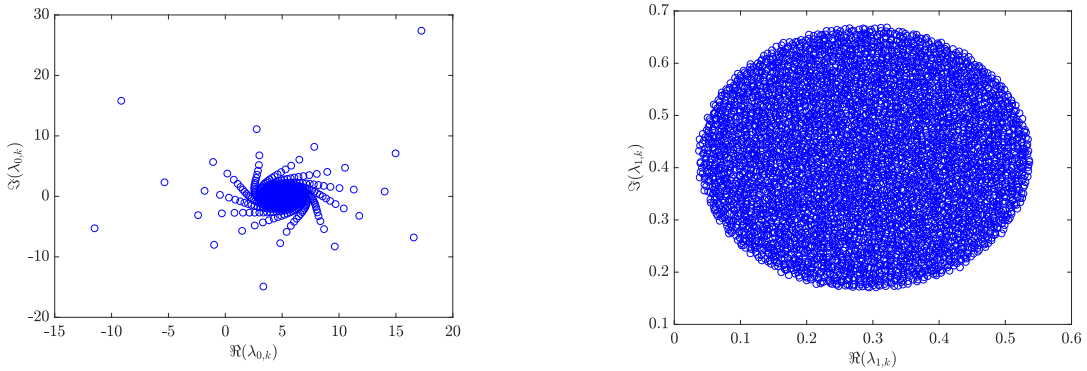
where $\mathbf{A}_i \in \mathbb{C}^{5000\times5000}$, with $i := 0, 1$, j is the imaginary unit and rand(N) constitutes the Matlab® function that generates an $N\times N$ array of uniformly distributed values. As documented in section 2 and elaborated in [17], iterative solution techniques are more efficient for systems with eigenvalues clustered far from 0. Thus, in order to obtain such a system, the eigenvalues of the matrix are manipulated accordingly. In detail, assuming $\mathbf{A}_0$ is diagonalizable, after deploying the eigendecomposition

$$\mathbf{\Lambda}_i = \mathbf{\Theta}_i^{-1}\mathbf{A}_i\mathbf{\Theta}_i \quad i := 0, 1, \tag{18}$$

the diagonal matrix $\mathbf{\Lambda}_i = \text{diag}(\lambda_{i,1}, \ldots, \lambda_{i,N})$ containing the eigenvalues $\lambda_{i,\kappa}$ of each matrix $\mathbf{A}_i$, as well as the corresponding eigenvector matrix $\mathbf{\Theta}_i$ are obtained. In order to construct a system with acceptable convergence properties the eigenvalues of $\mathbf{A}_0$ are replaced by

$$\lambda_{0,\kappa} = 5 + \frac{30e^{j20\kappa}}{\sqrt{\kappa}}, \quad \kappa := 1, \cdots, N, \tag{19}$$

which creates an eigenspectrum which corresponds to matrices stemming from the second kind Fredholm integral operator, commonly encountered in discretizations of the Helmholtz equation [17]. Such an eigenspectrum is characterized by a cluster of eigenvalues around 5 and the corresponding system converges with a rate that depends on the decay of the sequence to zero. Additionally, the eigenvalues of $\mathbf{A}_1$ are shifted by the highest eigenvalue magnitude as $\mathbf{\Lambda}_{1,\kappa} := \mathbf{\Lambda}_1 + \mathbf{I}\max(\|\lambda_{1,\kappa}\|), \kappa := 1, \ldots, N$, where $\mathbf{I}$ is the identity matrix, to guarantee that even at higher values of ω, any systems resulting from expression (16) will yield an acceptable convergence behaviour. The eigenvalue distribution of $\mathbf{A}_0$ and $\mathbf{A}_1$ are given in Figure 1.



(a) Eigenvalue distribution $\mathbf{A}_0$

(b) Eigenvalue distribution $\mathbf{A}_1$

Figure 1: Eigenspectra of coefficient matrices

4.2 Construction of the deflation basis

Having a system in the form (1) with a dependency of the system matrix of (16), it is possible to employ the AKR algorithm [2] to construct a global solution basis. For the setup of the AKR algorithm it is selected $r_{\text{tol}} := 0.1$. Executing the AKR algorithm yields a global basis $\mathbf{W}$ recycling 669 vectors collected from 20 systems sampled across the whole parameter domain,

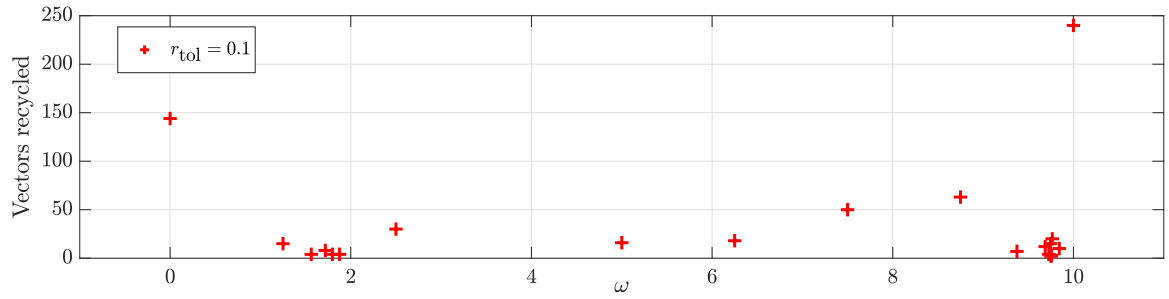


Figure 2: Vectors recycled for a randomly parametrized dense system for a basis constructed with AKR algorithm

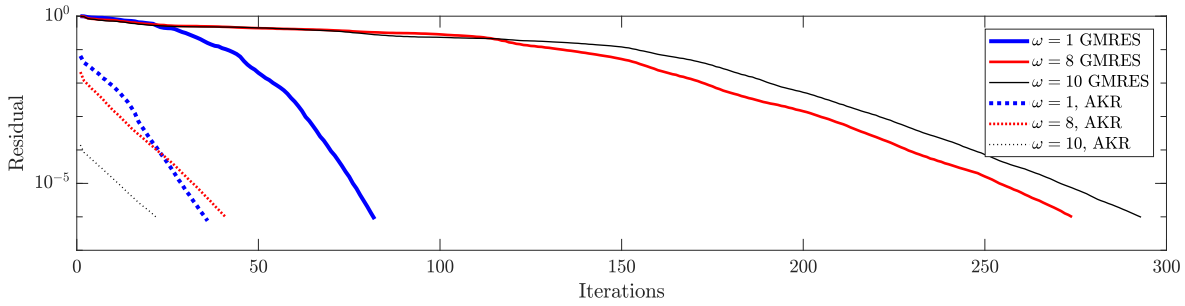


Figure 3: Convergence of system with deflation bases for different values of a randomly parametrized dense system

thus $|\Omega| = 20$. The sampling pattern to construct the basis is given in Figure 2. As illustrated, the sampling procedure employed with AKR is adaptive with certain regions of the parameter interval being more intensively sampled, while for others a coarse sampling is sufficient.

The effect of employing $\mathbf{W}$ as a deflation basis can be seen in Figure 3, where the convergence for three different values of ω is depicted. The convergence is accelerated in two ways, as not only the convergence rate is higher, but also a good initial guess is produced by computing the approximate solution on $\mathbf{W}$. This initial guess demonstrates the effect of the AKR algorithm, when used in a MOR scheme, as the approximate solution satisfies the predefined r_{tol}. This secondary effect was exploited to provide a global solution basis in [2, 16].

4.3 Deployment of the accelerated recycling strategy

Having created a high quality deflation basis $\mathbf{W}$, it is possible to employ the accelerated recycling scheme outlined in Algorithm 3 to solve the system described in (16) for a dense grid of parameter values $\Phi := [0 : 0.05 : 10]$ resulting into the solution of a sequence of $|\Phi| = 201$ linear systems. The main advantage of employing the proposed technique comes from sampling the subspaces in advance. In that way, during the online phase of the method only a limited number of iterations is required to achieve convergence of the solution. In addition, since the sampling is adaptive, essentially sampling more vectors where necessary, the number of iterations required in the online stage remains relatively constant for all $\omega \in \Psi$. Comparing to the conventional

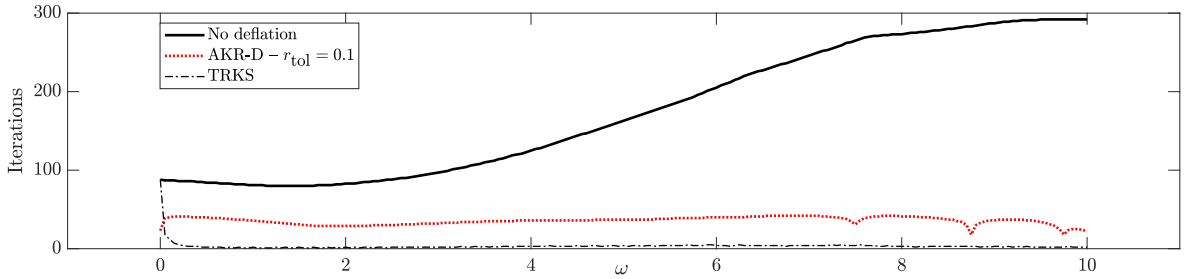


Figure 4: Iterations with accelerated recycling scheme and TRKS (Algorithm 2) for a randomly parametrized dense system

recycling technique (TRKS) given in Algorithm 2, it is noticed that gradually enriching the TRKS basis results in a very low number of iterations (Figure 4).

The total cost required for the parameter sweep is demonstrated in Table 1, where the total number of matrix-vector multiplications is used as a metric. As it is seen in the table, the proposed technique employs a large number of matrix vector multiplications in the offline stage to project the matrices $\mathbf{A}_0, \mathbf{A}_1$ onto $\mathbf{W}$. Comparing the total costs involved in the proposed method (offline+online) with the total cost of running an individual GMRES procedure for each system sequentially, the benefit of the proposed technique is apparent, reducing the total cost of the sweep to 24% of the initial cost. Finally, comparing to the TRKS conventional recycling scheme, it is noticed that although in terms of the iteration gain TRKS is superior to the proposed scheme, in terms of the total numerical cost induced the situation is reversed.

Solver	Matvecs Online	Matvecs Offline	Total	Ratio
GMRES (Non-Deflated)	35220	–	35220	1
TRKS	95231	–	95231	2.7
AKR – $r_{\text{tol}} = 0.1$	7176	1338	8514	0.24

Table 1: Cumulative matrix-vector products with accelerated recycling scheme, TRKS and GMRES for a randomly parametrized dense system

5 CONCLUSIONS

In this paper an accelerated recycling scheme is proposed. The presented technique implements a split of operations to an offline and an online stage aiming at accelerating the construction of a deflation based preconditioner. In order to achieve such a separation of operations, the AKR algorithm is deployed to construct a high quality deflation basis in the offline stage. The technique is tested on a randomly parametrized dense system and succeeds in speeding up the total cost of a parameter sweep by a factor of 4. Finally, comparing to a conventional recycling scheme, the proposed strategy is significantly faster as it shifts a large part of operations to an offline stage.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the Research Fund KU Leuven for its support.

REFERENCES

[1] 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 (5) (2006) 1651–1674.

[2] D. Panagiotopoulos, W. Desmet, E. Deckers, An automatic krylov subspaces recycling technique for the construction of a global solution basis of non-affine parametric linear systems, Computer Methods in Applied Mechanics and Engineering 373 (2021) 113510.

[3] M. R. Hestenes, E. Stiefel, et al., Methods of conjugate gradients for solving linear systems, Journal of research of the National Bureau of Standards 49 (6) (1952) 409–436.

[4] Y. Saad, M. H. Schultz, Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on scientific and statistical computing 7 (3) (1986) 856–869.

[5] D. P. O'Leary, The block conjugate gradient algorithm and related methods, Linear algebra and its applications 29 (1980) 293–322.

[6] A. Chapman, Y. Saad, Deflated and augmented Krylov subspace techniques, Numerical linear algebra with applications 4 (1) (1997) 43–66.

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

[8] J. Erhel, F. Guyomarc'h, An augmented conjugate gradient method for solving consecutive symmetric positive definite linear systems, SIAM Journal on Matrix Analysis and Applications 21 (4) (2000) 1279–1299.

[9] P. Benner, L. Feng, Recycling Krylov Subspaces for Solving Linear Systems with Successively Changing Right-hand Sides Arising in Model Reduction, Springer Netherlands, Dordrecht, 2011, pp. 125–140.

[10] F. Risler, C. Rey, Iterative accelerating algorithms with Krylov subspaces for the solution to large-scale nonlinear problems, Numerical algorithms 23 (1) (2000) 1.

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

[12] S. Keuchel, J. Biermann, O. von Estorff, A combination of the fast multipole boundary element method and Krylov subspace recycling solvers, Engineering Analysis with Boundary Elements 65 (2016) 136–146.

[13] K. M. Soodhalter, E. de Sturler, M. E. Kilmer, A survey of subspace recycling iterative methods, GAMM-Mitteilungen 43 (4) (2020) e202000016.

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

[15] C. Rey, F. Risler, A rayleigh–ritz preconditioner for the iterative solution to large scale nonlinear problems, Numerical Algorithms 17 (3) (1998) 279–311.

[16] D. Panagiotopoulos, W. Desmet, E. Deckers, Parametric model order reduction for acoustic bem systems through a multi-parameter krylov subspaces recycling strategy, International Journal for Numerical Methods in Engineering (2022).

[17] X. Antoine, M. Darbas, An introduction to operator preconditioning for the fast iterative integral equation solution of time-harmonic scattering problems, Multiscale Science and Engineering (2021) 1–35.