An Automatic Krylov subspaces Recycling technique for the construction of a global solution basis of non-affine parametric linear systems

Dionysios Panagiotopoulos[a,b,*], Wim Desmet[a,b], Elke Deckers[a,b]

[a]*KU Leuven, Department of Mechanical Engineering, Belgium*
[b]*DMMS lab, Flanders Make*

Abstract

Recycling of Krylov subspaces is often used to obtain an augmentation subspace in the context of iterative algorithms for the solution of sequences of linear systems. However, it still remains difficult to quantify the effect of subspaces recycling and thus to determine the dimension of the subspaces to be recycled targeting a specific accuracy. In that context, this work proposes the Automatic Krylov subspaces Recycling algorithm (AKR) that automates the selection of Krylov subspaces to be recycled and generates a basis that can provide sufficiently accurate approximations of the solution for a parametric system on a predefined interval Ψ. The constructed basis is employed as a Galerkin projection basis for a model order reduction (MOR) scheme in the context of non-affine parametric systems. In the offline phase of the MOR scheme, AKR constructs a projection subspace $\mathcal{W}$ by sampling Krylov subspaces at an iteratively built set of parameter values Ω. Keeping a balance between the solution accuracy and the memory required, the algorithm, apart from guaranteeing a predefined residual level r_{tol}, also permits the predetermination of a threshold regarding the maximum memory employed. Nevertheless, following the unpreconditioned Krylov methods effectiveness criteria, the proposed technique proves to be efficient for systems with relatively clustered eigenvalues such as the ones encountered in

[*]Corresponding author
Email address: dionysios.panagiotopoulos@kuleuven.be (Dionysios Panagiotopoulos)

the conventional Boundary Element Method. The performance of the proposed AKR algorithm is assessed in comparison with an alternative version of the reduced basis method, which is based on the same assumptions as the AKR and is specifically designed to provide a good benchmark. These techniques are deployed for a randomly generated complex system and an acoustic BEM system. The advantage of employing AKR is demonstrated as fewer system assemblies are required for the construction of the projection basis.

Keywords: Krylov subspaces recycling, Automatic procedure, Memory efficient, Model Order Reduction, BEM

1. Introduction

Solving mechanical problems modelled with partial differential equations (PDE) often results in linear parameter dependent systems, where the solution is required in a multiresolution context, i.e. for multiple values of a single dimension parameter $\omega \in \Psi$, where $\Psi := [\omega_{\min}, \omega_{\max}]$ is an interval in a single dimensional parameter space. Such systems can be written as

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

where $\mathbf{A} : \Psi \to \mathbb{C}^{N \times N}$ and $\mathbf{b}, \mathbf{x} : \Psi \to \mathbb{C}^N$. With increasing complexity of the mechanical problems, the corresponding linear systems and solution intervals increase in size, rendering the system solution in such a multiresolution context quite computationally demanding. Dealing with systems that involve a non-affine parameter dependency, i.e. $\mathbf{A}$ and $\mathbf{b}$ are not of the form $\mathbf{A}(\omega) = \sum_{k=1}^{K} \Theta_k(\omega)\mathbf{A}_k$ with $\mathbf{A}_k \in \mathbb{C}^{N \times N}$ and $\Theta_k : \Psi \to \mathbb{C}$, elevates also the assembly costs of the multiresolution analysis, as the assembly of the system for each $\omega \in \Psi$ is required. Such systems arise for instance in an acoustic Boundary Element Method (BEM) [1] analysis where the system matrices are non-affinely dependent from frequency. In that context, this work proposes an algorithm to efficiently construct a subspace $\mathcal{W}$, on which approximate solutions $\hat{\mathbf{x}}(\omega)$ for

$\omega \in \Psi$ can be found, given a target required accuracy r_{tol} and avoiding numerous system assemblies and full solutions.

Accelerating the solution of sequences of linear systems has been the objective of multiple works. Approaching the problem from the perspective of individual linear systems for each $\omega \in \Psi$, iterative methods offer significant acceleration not only for the solution of linear systems [2, 3] but also for the approximation of their eigenvalues and eigenvectors. Among the iterative solution methods, Krylov subspaces play a significant role as several algorithms, such as the conjugate gradient method [4] and GMRES [5], are based on the iterative expansion of some kind of Krylov subspaces. As their efficiency depends on the distribution of the eigenvalues of the considered system, improvements of these methods have been proposed including deflation and augmentation strategies as described in [6, 7, 8, 9], demonstrating the great advantage offered by the reuse of Krylov subspaces. Most of the techniques related to the reuse of Krylov subspaces concern the solution of an identical linear system for a sequence of varying right-hand sides, see e.g. [10]. However, certain techniques have been proposed including recycling of vectors as well among slowly varying systems [11, 12, 13, 14]. Specifically, Risler and Rey in [11] propose to augment the projection space by all the Krylov subspaces of the previous iterations while Gosselet et al. in [12] make a step further proposing a selective reuse of Krylov subspaces. Finally, Parks et al. in [14] generalize the GMRES-DR algorithm by Morgan [7] that recycles approximate invariant subspaces, by allowing the recycling of any subspace and thus proposing the GCRO-DR algorithm for sequences of linear systems.

Dealing with sequences of linear systems renders the connection of Krylov solvers with model order reduction (MOR) techniques attainable and straightforward. Originating from the model order reduction perspective, Ryckelynck derived the *a-priori* hyperreduction method [15, 16], which combines an incremental proper orthogonal decomposition (POD) [17] with the addition of few Krylov subspaces for time dependent non-linear problems. Another similar approach was proposed by Kerfriden et al. [18], where the augmented conjugate gradient method (AugCG) [8] is deployed to obtain an approximate solution, which is then also appended to the reduction basis. Krylov subspaces in MOR are also usually employed for linear time invariant systems [19] or the linearized counterparts of non-linear systems to match the subspace of the moments of a Padé approximation of the transfer function of a system [20].

Other common model order reduction approaches are also based on the concept of finding an appropriate basis for the reduction of (multi-)parameter dependent systems. Although modal truncation methods [21] constitute popular techniques of reducing the order of the model, the calculation of the eigenvectors in problems with non-affine parameter dependency might not be a simple task. POD methods [17, 22] are frequently employed in the context of complicated non-linear problems to construct some empirical eigenvectors. Having a similar objective to the POD, the reduced basis method [23, 24] constitutes one of the most efficient approaches to construct a representative basis for affinely parametrized partial differential equations as it involves a greedy sampling scheme. It iteratively enriches the reduced basis with $\mathbf{x}(\omega_m)$, where ω_m is the parameter value with the highest residual, and thus it succeeds in guaranteeing a predefined residual level for all $\omega \in \Psi$. Nevertheless as it relies upon full system solutions and a residual based error estimator, the cost of the offline procedure for building the reduced basis can be considerable. Specifically, it strongly depends on the cost of solving the full system for each newly appended $\mathbf{x}(\omega_m)$ and the cost of assembling the system and computing the residual for $\omega \in \Phi$ within each iteration, where $\Phi \subset \Psi$ consists of values Ψ on a dense grid. Thus, in the case of non-affine parametric systems with complex responses, the use of the reduced basis algorithm is hindered as a high number of iterations, involving an increased number of system assemblies and full solutions might be required. Recovering affinity before creating the reduced basis as discussed in [25] might cause memory excess especially in case of non-sparse systems. These issues are illustrated in [26] for systems arising in BEM for acoustic problems.

In that context, this work proposes a new algorithm to assemble a reduced basis $\mathbf{W}$ for systems of type (1), where $\mathcal{W} = \text{span}\{\mathbf{W}\}$, employing the concept of Krylov subspaces recycling that aims to reduce the system assemblies and full solutions required in the offline phase of a respective MOR technique. It constitutes a reduced basis method that utilizes as input, instead of the full solutions, the Arnoldi vectors assembled for the calculation of approximate solutions through a full orthogonalization method (FOM) scheme [27, 28, 29] at an ordered set of parameter values $\Omega \subset \Psi$. The dimension $s(\omega)$ of the respective Krylov subspace $\mathcal{K}^{\omega}_{s(\omega)}$ employed for each $\omega \in \Omega$ as well as Ω itself are determined iteratively through an automated procedure. In detail, in the construction stage of $\mathbf{W}$ the residual

$$\mathbf{r}_i(\omega_m) := \mathbf{b}(\omega_m) - \mathbf{A}(\omega_m)\hat{\mathbf{x}}_i(\omega_m) \perp \mathbf{W}, \tag{2}$$

is built for each iteration i, where

$$\hat{\mathbf{x}}_i(\omega_m) \in \mathbf{x}_0(\omega) + \mathcal{W}, \tag{3}$$

where $\hat{\mathbf{x}}_i(\omega_m)$ is the i^{th} approximation of the true solution $\mathbf{x}(\omega_m)$ at a parameter value $\omega_m \in \Psi \backslash \Omega$ and $\mathbf{x}_0 : \Psi \rightarrow \mathbb{C}^N$ is an initial guess for the approximated solution. By deploying a stagnation criterion, it is decided whether to enrich $\mathcal{W}$ by increasing the dimension $s(\omega_a), s(\omega_b)$ of $\mathcal{K}^{\omega_a}_{s(\omega_a)}, \mathcal{K}^{\omega_b}_{s(\omega_b)}$, where $\omega_a, \omega_b \in \Omega$ the elements of Ω closest to ω_m, or in case of stagnation, to enrich $\mathcal{W}$ with $\mathcal{K}^{\omega_m}_{s(\omega_m)}$, thus $\Omega \leftarrow \Omega \cup \{\omega_m\}$, increasing $s(\omega_m)$ until $\|\mathbf{r}_i(\omega_m)\| \leq r_{\text{tol}}$ is satisfied, where r_{tol} is a preselected residual tolerance and $\|\mathbf{r}_i(\omega_m)\|$ is the Euclidean norm of the respective residual.

As subspaces recycling is more efficient for closely related systems [14], the method is based on an residual based error indicator that enforces r_{tol} only at $\omega_m := 0.5(\omega_a + \omega_b)$, where ω_a, ω_b are two consecutive elements of Ω. Although the error indicator builds and checks only $\|\mathbf{r}_i(\omega_m)\|$ and consequently r_{tol} can be mathematically guaranteed only for ω_m, given the nature of subspace recycling, in practice, the method succeeds in achieving $\|\mathbf{r}(\omega)\| \leq r_{\text{tol}} \quad \forall \quad \omega \in \Psi$. Starting from that, the basis can be employed as an augmentation space in the augmented conjugate gradient algorithm [12] having quantified the effect of recycling and guaranteeing that an approximate solution is close to the true solution. However, in this work, the basis is deployed in a model order reduction context for the Galerkin projection of non-affine parametric algebraic systems that result from a discretized PDE. To enable an offline projection of the respective algebraic system, first the given non-affine parameter dependency can be lifted with a polynomial approximation of the system matrix coefficients without inflicting memory excess as described in [30]. Nevertheless, to alleviate any concerns about the memory required either by the projected polynomial approximation of the system or the basis $\mathbf{W}$ itself, a memory constrained version of the algorithm is also proposed. This version accepts as input a positive integer $\ell \in \mathbb{N}$, which determines the maximum dimension of the subspace to be recycled. Given this setting, the algorithm yields the maximum interval $\Psi_\ell \subseteq \Psi$ such that $\|\mathbf{r}(\omega)\| \leq r_{\text{tol}} \quad \forall \quad \omega \in \Psi_\ell$. This leads to an adaptive windowing technique that outputs different bases $\mathbf{W}_1, \ldots, \mathbf{W}_Q$ always respecting the predefined maximum dimension, that are efficient for dedicated parameter subintervals $\Psi_{\ell_1} \cup \cdots \cup \Psi_{\ell_Q} = \Psi$, with Q being the total number of subintervals.

The automatic recycling method presented here does not include any system preconditioning technique. This implies that it works more efficiently for systems with relatively clustered eigenvalues that promote fast convergence with iterative solution algorithms [31]. Non-affine parametric systems with such an eigenvalue distribution are usually encountered in case of the conventional BEM for the Helmholtz equation [32, 33] and thus the use of iterative solution techniques is facilitated as shown in [34, 35, 36]. Consequently, the method proposed in this work can be deployed as an automation step of the model reduction technique for BEM proposed in [30]. Specifically, in contrast to the proposed AKR algorithm, in [30] the elements of Ω and the dimension of the respective Krylov subspaces $s(\omega)$ are defined as user inputs resulting in a basis that might not be effective for the full parameter range Ψ.

The paper is organized as follows. In section 2 the full orthogonalization method including an extension with an augmentation strategy is introduced. In section 3, based on the augmented FOM, the Automatic Krylov subspaces Recycling (AKR) technique is presented and then the alternative memory efficient version is demonstrated. Section 4 enables the combination of AKR with a model reduction framework for BEM, introducing also a windowing technique based on the memory constrained algorithm. In addition, an especially designed for benchmarking purposes version of the reduced basis algorithm is presented that employs the same assumptions as AKR. In section 5 the proposed techniques are assessed on selected numerical examples. First the AKR is deployed on a random system and then the combined AKR with MOR is employed for an academic BEM model case. Comparing both examples to the alternative reduced basis approach, the advantage of employing AKR is evident necessitating fewer system assemblies and in general inducing lower computational cost. Finally, section 6 summarizes and concludes the paper.

2. Full Orthogonalization Method

The deployment of Krylov subspaces is widely used for the iterative solution of linear systems and several methods are based on Krylov subspaces constructed through the Arnoldi [37] or Lanczos [38] algorithm. An iterative method suitable for dealing with generally unsymmetric systems is termed as the full orthogonalization method (FOM), which in case of Hermitian positive definite systems matrices coincides with the well-known conjugate gradient method [39, Chapter 10]. The name of the method stems from the fact that it demands that the residual be orthogonal to the projection basis in each iteration. This section presents the basic principles of the method and an augmentation strategy that will be employed in section 3 for the assembly of a global reduction basis. Detailed information about the characteristics of the method can be found in the seminal works of Axelsson [27] and Saad [28].

2.1. The original method

The method considers a linear system of equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{4}$$

Algorithm 1 Full orthogonalization method

```
Input: system $\mathbf{A} \in \mathbb{C}^{N\times N}, \mathbf{b} \in \mathbb{C}^N$, initial guess $\mathbf{x}_0 \in \mathbb{C}^N$ and residual tolerance $r_{\text{tol}} \in \mathbb{R}$;
$\mathbf{r}_0 \leftarrow \mathbf{b} - \mathbf{A}\mathbf{x}_0$;    $\mathbf{v}_1 \leftarrow \frac{\mathbf{r}_0}{\|\mathbf{r}_0\|}$;
$\mathbf{V}_1 \leftarrow [\mathbf{v}_1]$;    $m \leftarrow 1$;
while $\|\mathbf{r}_{m-1}\| \geq r_{\text{tol}}$ do
    $\mathbf{v}_{m+1} \leftarrow \mathbf{A}\mathbf{v}_m$;    $\mathbf{v}_{m+1} \leftarrow \frac{\mathbf{v}_{m+1}}{\|\mathbf{v}_{m+1}\|}$;
    for $p \leftarrow 1 : m$ do
        $h_{p,m} \leftarrow \mathbf{v}_p^H \mathbf{v}_{m+1}$;    $\mathbf{v}_{m+1} \leftarrow \mathbf{v}_{m+1} - h_{p,m}\mathbf{v}_p$;
    end for
    $h_{m+1,m} \leftarrow \|\mathbf{v}_{m+1}\|$;    $\mathbf{v}_{m+1} \leftarrow \frac{\mathbf{v}_{m+1}}{\|\mathbf{v}_{m+1}\|}$;    $\mathbf{V}_{m+1} \leftarrow [\mathbf{V}_m \quad \mathbf{v}_{m+1}]$;
    $\hat{\mathbf{x}}_m \leftarrow \mathbf{V}_m(\mathbf{H}_m)^{-1}\mathbf{V}_m^H\mathbf{r}_0$;
    $\mathbf{r}_m \leftarrow \mathbf{r}_0 - \mathbf{A}\hat{\mathbf{x}}_m$;    $m \leftarrow m + 1$;
end while
Output: Approximation basis $\mathbf{V} \in \mathbb{C}^{N\times m-1}$ and solution approximation $\hat{\mathbf{x}}_{m-1} \in \mathbb{C}^N$
```

where $\mathbf{A} \in \mathbb{C}^{N\times N}$ and $\mathbf{b}, \mathbf{x} \in \mathbb{C}^N$. As outlined in Algorithm 1, the FOM computes, within each iteration m, approximate solutions

$$\hat{\mathbf{x}}_m \in \mathbf{x}_0 + \mathcal{K}_m, \tag{5}$$

where $\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0) := \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots, \mathbf{A}^{m-2}\mathbf{r}_0, \mathbf{A}^{m-1}\mathbf{r}_0\}$ is the Krylov subspace produced by the initial residual $\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$ and $\mathbf{x}_0$ is an initial guess. The dimension of the subspace is increased in each iteration until $\|\mathbf{r}_m\| \leq r_{\text{tol}}$, where

$$\mathbf{r}_m := \mathbf{b} - \mathbf{A}\hat{\mathbf{x}}_m \perp \mathbf{v}_j, \qquad j := 1, \ldots, m. \tag{6}$$

is the residual after each iteration. The $\mathbf{r}_m$ remains orthogonal with respect to the Hermitian inner product to the linearly independent Arnoldi vectors $\mathbf{v}_j$ composing the basis $\mathbf{V}_m := [\mathbf{v}_1 \quad \mathbf{v}_2 \ldots \mathbf{v}_m]$, where $\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0) = \text{span}\{\mathbf{V}\}$, and r_{tol} is a predefined target residual. The basis $\mathbf{V}$ is commonly produced by an Arnoldi algorithm [37] in combination with a modified Gram-Schmidt orthogonalization procedure [40] given in Algorithm 1 lines 5-9. In Algorithm 1 h_{ij} are the entries of the Hessenberg matrix $\mathbf{H}_m$, which represents the projection of $\mathbf{A}$ on $\mathcal{K}_m$, and $\mathbf{V}^H$ is the conjugate transpose of $\mathbf{V}$.

The cost of repeatedly calculating the residual is not considered significant. In detail, since the projection of the system is already computed in the Hessenberg matrix $\mathbf{H}_m$ the procedure only consists of solving the reduced system and the computing the residual $\mathbf{r}_m$ corresponding to lines 10-11 in Algorithm 1. The cost of calculating the residual scales with $\mathcal{O}(N^2)$, while the solution of the reduced system with $\mathcal{O}(m^3)$. Adding multiple vectors per iteration can alleviate the cost of the procedure by limiting the number of iterations and consequently the number of reduced solutions and residuals computed.

2.2. An augmentation strategy

Augmentation strategies to enrich the generated Krylov subspaces can be proven useful and speed up significantly the convergence of the iterative procedure approximating the solution of (4) [6]. Reusing subspaces can also provide a great advantage in the context of the consecutive solution of related systems [8, 13]. Thus, aiming for a multiresolution analysis of parameter dependent systems as given in (1), it is beneficial to define an augmentation strategy to recycle information obtained from the sequential deployment of the FOM presented in Algorithm 1.

The augmentation strategy employed in this work constitutes a slightly modified version of the augmented Arnoldi – modified Gram-Schmidt algorithm introduced in [41]. Namely, the newly generated Arnoldi vectors $\mathbf{v}_j$ by Algorithm 1 are orthogonalized with respect to a given orthonormal augmentation basis $\mathbf{W}$, with $\mathcal{W} = \operatorname{span}\{\mathbf{W}\}$, to yield $\mathbf{v}_j^{\perp\mathbf{W}}$. This comes in contrast to the procedure followed in [41], where the column vectors $\mathbf{w}_j$ of $\mathbf{W}$ are orthogonalized with respect to $\mathbf{V}$. The required orthogonalization is conducted with a modified Gram-Schmidt procedure. The Arnoldi basis of a related system is often

Algorithm 2 Augmented full orthogonalization method

```
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 augmentation basis $\mathbf{W} \in \mathbb{C}^{N \times \ell}$;
$\mathbf{r}_0 \leftarrow \mathbf{b} - \mathbf{A}\mathbf{x}_0$;
$\hat{\mathbf{x}}_1 \leftarrow \mathbf{W}(\mathbf{W}^H\mathbf{A}\mathbf{W})^{-1}\mathbf{W}^H\mathbf{r}_0$;   $\mathbf{r}_1 \leftarrow \mathbf{r}_0 - \mathbf{A}\hat{\mathbf{x}}_1$;   $\mathbf{v}_1 \leftarrow \frac{\mathbf{r}_1}{\|\mathbf{r}_1\|}$;
$\mathbf{V}_1 \leftarrow [\mathbf{v}_1]$;   $m \leftarrow 1$;   $\ell \leftarrow \text{rank}(\mathbf{W})$
while $\|\mathbf{r}_m\| \geq r_{\text{tol}}$ do
    $\mathbf{v}_{m+1} \leftarrow \mathbf{A}\mathbf{v}_m$;   $\mathbf{v}_{m+1} \leftarrow \frac{\mathbf{v}_{m+1}}{\|\mathbf{v}_{m+1}\|}$;
    for $p \leftarrow 1 : m$ do
        $h_{p,m} \leftarrow \mathbf{v}_p^H\mathbf{v}_{m+1}$;   $\mathbf{v}_{m+1} \leftarrow \mathbf{v}_{m+1} - h_{p,m}\mathbf{v}_l$;
    end for
    $\mathbf{v}_{m+1}^{\perp\mathbf{W}} \leftarrow \mathbf{v}_{m+1}$;   $\mathbf{V}_{m+1} \leftarrow [\mathbf{V}_m \quad \mathbf{v}_{m+1}]$
    for $i \leftarrow 1 : \ell$ do
        $\beta_i \leftarrow \mathbf{w}_i^H\mathbf{v}_{m+1}$;   $\mathbf{v}_{m+1}^{\perp\mathbf{W}} \leftarrow \mathbf{v}_{m+1}^{\perp\mathbf{W}} - \beta_i\mathbf{w}_i$;
    end for
    $\mathbf{v}_{m+1}^{\perp\mathbf{W}} \leftarrow \frac{\mathbf{v}_{m+1}^{\perp\mathbf{W}}}{\|\mathbf{v}_{m+1}^{\perp\mathbf{W}}\|}$;   $\mathbf{W} \leftarrow [\mathbf{W} \quad \mathbf{v}_{m+1}^{\perp\mathbf{W}}]$;
    $\hat{\mathbf{x}}_{m+1} \leftarrow \mathbf{W}(\mathbf{W}^H\mathbf{A}\mathbf{W})^{-1}\mathbf{W}^H\mathbf{r}_0$;   $\mathbf{r}_{m+1} \leftarrow \mathbf{r}_0 - \mathbf{A}\hat{\mathbf{x}}_{m+1}$;
    $\ell \leftarrow \ell + 1$;   $m \leftarrow m + 1$
end while
Output: Approximation basis $\mathbf{W} \in \mathbb{C}^{N \times \ell - 1}$, Arnoldi basis $\mathbf{V} \in \mathbb{C}^{N \times m-1}$ and solution approximation $\hat{\mathbf{x}}_{m-1} \in \mathbb{C}^{N}$
```

employed as an augmentation basis in Algorithm 2. The procedure for electing an efficient augmentation basis in the context of parametric systems is demonstrated in section 3. The augmented FOM outlined in Algorithm 2, proceeds to the iterative assessment of the produced residual $\mathbf{r}_m$ and halts when the predefined residual tolerance r_{tol} is reached.

The motivation for reversing the bases in Algorithm 2 as compared to the one presented in [41] lies on the global intended use of the output basis $\mathbf{W}$, the construction of which will be described further in chapter 3. In detail, in a multiresolution context i.e. $\mathbf{A}^i\mathbf{x}^i = \mathbf{b}^i$, where $i := 1, \ldots, N_{\text{tot}}$ and N_{tot} the total number of systems to be resolved, the basis $\mathbf{W}$ needs to be able to simultaneously provide quality approximations $\hat{\mathbf{x}}^i$ for all $i := 1, \ldots, N_{\text{tot}}$ to achieve a global character. This cannot be guaranteed with the algorithm presented in [41]. On the contrary, sequential deployment of Algorithm 2 for all $i := 1, \ldots, N_{\text{tot}}$ will result in an augmentation basis for the system with $i := N_{\text{tot}}$ that will also constitute a high quality projection basis for systems with $i := 1, \ldots, N_{\text{tot}} - 1$. The above scheme is quite similar to the total reuse of Krylov subspaces (TRKS) reported by Gosselet et al. in [12], which recycles all the Krylov subspaces for all N_{tot} systems.

The presented augmentation strategy scores lower in terms of efficiency with respect to other documented augmentation strategies such as the AugCG, presented in [8] regarding the required dimension of Krylov subspaces to be assembled. This comes naturally as a consequence of the fact that the Krylov sequence $\mathcal{K}_m$ does not exploit a projector which would inherently take into account the information contained in the subspace $\mathcal{W}$. Nevertheless, it is preferred as following a similar procedure to AugCG would incur in different subspaces for each system i, disrupting the global character of the basis. Hence, it proves more beneficial for a multiresolution context where the basis is assembled only in advance and not for each system individually.

3. Automatic Krylov recycling technique

In this section the proposed automatic techniques for the recycling of Krylov subspaces are introduced. In the first part the AKR algorithm is demonstrated and the basic tools and limitations are presented. In the second part a memory efficient alternative is introduced, where the parameter interval for which the produced projection basis is able to sufficiently approximate the solution, is yielded adaptively leveraging a memory constraint.

3.1. Basic approach

Consider a parametric linear system of the form (1), which is restated here for reasons of completeness

$$\mathbf{A}(\omega)\mathbf{x}(\omega) = \mathbf{b}(\omega), \tag{7}$$

where $\omega \in \Psi$ is a single dimension parameter, $\mathbf{A} : \Psi \to \mathbb{C}^{N\times N}, \mathbf{b} : \Psi \to \mathbb{C}^N$ and $\Psi := [\omega_{\min}, \omega_{\max}] \subset \mathbb{R}$ a parameter interval. Demanding the solution $\mathbf{x}(\omega)$ for each $\omega \in \Phi \subset \Psi$, where Φ is a discrete set with cardinality $|\Phi| = N_{\text{tot}}$ results in dealing with a sequence of linear systems for which a basis $\mathbf{W} = \text{span}\{\mathcal{W}\}$ can be constructed using the procedure outlined in Algorithm 2. Namely, Algorithm 2 can be sequentially deployed for all $\omega \in \Psi$ resulting in the basis $\mathbf{W} \in \mathbb{C}^{N\times \ell}$ that leads to approximate solutions $\hat{\mathbf{x}}(\omega)$ from which the residual

$$\mathbf{r}(\omega) = \mathbf{r}_0(\omega) - \mathbf{A}(\omega)\hat{\mathbf{x}}(\omega) \perp \mathbf{W} \tag{8}$$

satisfies $\|\mathbf{r}(\omega)\| \le r_{\text{tol}} \quad \forall \omega \in \Phi$, where r_{tol} is the residual tolerance and

$$\hat{\mathbf{x}}(\omega) \in \mathbf{x}_0(\omega) + \mathcal{W}, \tag{9}$$

with $\mathbf{x}_0 : \Psi \to \mathbb{C}^N$ an initial guess of the solution and $\mathbf{r}_0(\omega)$ the corresponding residual. Nevertheless, sequentially deploying Algorithm 2 will require the assembly and construction of Krylov subspaces $\mathcal{K}^{\omega}_{s(\omega)}$ of dimension $s(\omega)$ of the system (7) for all $\omega \in \Phi$ which in case of non-affine parametric systems can be a time consuming task. Therefore, in this section the algorithm AKR is presented that allows the construction of the basis $\mathbf{W}$ by deploying Algorithm 2, and thus leveraging only the Krylov subspaces $\mathcal{K}^{\omega}_{s(\omega)}$, for $\omega \in \Omega$, a discrete ordered set, where $\Omega \subset \Phi$ and $|\Omega| \ll |\Phi|$.

The AKR algorithm automates the construction of Ω as well as the selection of the dimension $s(\omega)$ of the subspaces $\mathcal{K}^{\omega}_{s(\omega)}, \omega \in \Omega$. It involves an adaptive procedure, which is based on the assumption that given a basis $\mathbf{W}$,

$$r(\omega) := \|\mathbf{b}(\omega) - \mathbf{A}(\omega)\mathbf{W}(\mathbf{W}^H\mathbf{A}(\omega)\mathbf{W})^{-1}\mathbf{W}^H\mathbf{b}(\omega)\|, \tag{10}$$

$r : \Psi \to \mathbb{R}^+$ is $C^0(\Psi)$, implying that Krylov subspaces recycling is more efficient for neighbouring systems in the parameter interval Ψ. Leveraging this assumption, the algorithm commences by deploying Algorithm 1 having as inputs the systems at $\omega_a := \omega_{\text{min}}$ and $\omega_b := \omega_{\text{max}}$, the extremes of the interval Ψ, $r_{\text{tol}_1} := r_{\text{tol}}$ and $r_{\text{tol}_2} := \alpha r_{\text{tol}}$, where $\alpha \leq 1$, resulting in the augmentation bases

$$\begin{aligned} \mathbf{V}_{\omega_a, r_{\text{tol}_1}} =& [\mathbf{v}_{\omega_a,1} \dots \mathbf{v}_{\omega_a,s(\omega_a)}], \\ \mathbf{V}_{\omega_b, r_{\text{tol}_1}} =& [\mathbf{v}_{\omega_b,1} \dots \mathbf{v}_{\omega_b,s(\omega_b)}] \end{aligned} \tag{11}$$

and enrichment bases

$$\begin{aligned} \mathbf{V}_{\omega_a, r_{\text{tol}_2}} =& [\mathbf{V}_{\omega_a, r_{\text{tol}_1}} \quad \mathbf{v}_{\omega_a,s(\omega_a)+1} \dots \mathbf{v}_{\omega_a,p(\omega_a)}], \\ \mathbf{V}_{\omega_b, r_{\text{tol}_2}} =& [\mathbf{V}_{\omega_b, r_{\text{tol}_1}} \quad \mathbf{v}_{\omega_b,s(\omega_b)+1} \dots \mathbf{v}_{\omega_b,p(\omega_b)}]. \end{aligned} \tag{12}$$

Then, it continues with the creation of an initial orthonormal augmentation basis $\mathbf{W}$ by employing a QR decomposition on $[\mathbf{V}_{\omega_a, r_{\text{tol}_1}} \quad \mathbf{V}_{\omega_b, r_{\text{tol}_1}}]$. Having created $\mathbf{W}$, the algorithm follows a bisection strategy that iteratively checks the residual $\mathbf{r}_i(\omega_m)$ for each iteration i, where $\omega_m := 0.5(\omega_a + \omega_b)$. In case $\|\mathbf{r}_i(\omega_m)\| > r_{\text{tol}}$, the augmentation basis is further enriched iteratively by the vectors $\mathbf{v}^{\perp \mathbf{W}}_{\omega_a,s(\omega_a)+i}$ and $\mathbf{v}^{\perp \mathbf{W}}_{\omega_b,s(\omega_b)+i}$, which are $\mathbf{v}_{\omega_a,s(\omega_a)+i}$ and $\mathbf{v}_{\omega_b,s(\omega_b)+i}$ orthogonalized to $\mathbf{W}$ with a modified Gram-Schmidt procedure. The enrichment with additional Arnoldi vectors produced at ω_a and ω_b is stopped in case either stagnation occurs or $s(\omega_a) + i = p(\omega_a)$ and $s(\omega_b) + i = p(\omega_b)$, i.e. all the additional Arnoldi vectors stored in $\mathbf{V}_{\omega_a, r_{\text{tol}_2}}$ and $\mathbf{V}_{\omega_b, r_{\text{tol}_2}}$ are employed. The stagnation criterion is defined by

$$\frac{|\|\mathbf{r}_i(\omega_m)\| - \|\mathbf{r}_{i+S}(\omega_m)\||}{\|\mathbf{r}_{i+S}(\omega_m)\|} < \epsilon_{\text{stag}} \tag{13}$$

where $S \in \mathbb{N}$ and $\epsilon_{\text{stag}} \in (0, 1)$. For this case, Algorithm 2 is deployed to achieve $\|\mathbf{r}(\omega_m)\| \leq r_{\text{tol}}$, having as input the system at ω_m, r_{tol_1} and r_{tol_2} and outputs $\mathbf{W}_{\omega_m,\text{tol}_1}$, $\mathbf{W}_{\omega_m,\text{tol}_2}$, $\mathbf{V}_{\omega_m,\text{tol}_1}$, $\mathbf{V}_{\omega_m,\text{tol}_2}$. Subsequently the sampling information is updated by $\Omega \leftarrow \Omega \cup \{\omega_m\}$, the new augmentation basis becomes $\mathbf{W} \leftarrow \mathbf{W}_{\omega_m,\text{tol}_1}$ and the additional enrichment basis holds as $\mathbf{V}_{\omega_m,\text{tol}_2}$. The same bisection procedure continues by defining a new $\omega_m := 0.5(\omega_a + \omega_b)$ for all systems at $\omega_a, \omega_b \in \Omega$, any two consecutive elements of Ω, until $\|\mathbf{r}_i(\omega_m)\| \leq r_{\text{tol}}$ without any occurrence of stagnation. Finally, having ensured the residual criterion for all ω_m without employing any of the Krylov subspaces $\mathcal{K}^{\omega_m}_{s(\omega_m)}$, a basis $\mathbf{W}$ is assembled that, in practice, fulfils the objective of being able to produce approximate solutions $\hat{\mathbf{x}}(\omega)$ such that $\|\mathbf{r}(\omega)\| \leq r_{\text{tol}} \quad \forall \quad \omega \in \Psi \supset \Phi$ assembling (7) only at a limited number of $\omega \in \Psi$. The AKR algorithm is outlined in Algorithm 3.

In the above, the user defined parameters S, α and ϵ_{stag} appear to play a role in the creation of the basis. Selecting a lower α leads to obtaining more Arnoldi vectors for the enrichment procedure, namely $\mathbf{W}_{\omega_m, r_{\text{tol}_2}}$, however requiring larger memory for the storage of these vectors. Assembling these additional vectors does not significantly increase the computational cost, as it amounts only to a few additional iterations in Algorithm 1 or 2. Regarding the stagnation parameters S and ϵ_{stag}, higher S or lower ϵ_{stag} leads to a more difficult to satisfy stagnation criterion and thus to the addition of a larger number of enrichment vectors to $\mathbf{W}$ before stagnation occurs. These two parameters affect mostly the construction of Ω and $s(\omega)$, $\omega \in \Omega$ but not the quality of the output basis $\mathbf{W}$.

The automatic procedure of Algorithm 3 resembles the reduced basis method in the way that it enriches the global basis by information extracted from the system that, based on the assumption made, yields the highest residual within a given interval, i.e. at ω_m in $[\omega_a, \omega_b]$. However, the enrichment is conducted either by including additional Arnoldi vectors for $\omega_a, \omega_b \in \Omega$ or by sampling new points $\Omega \leftarrow \Omega \cup \{\omega_m\}$, starting by default and without loss of generality by the upper end of Ω. Instead of using the full solutions $\mathbf{x}(\omega)$ of the system to construct the projection subspace $\mathcal{W}$, it employs the components of the full solutions i.e. the Arnoldi vectors $\mathbf{V}_\omega$, thus facilitating convergence of the solution for $\omega \in \Psi \backslash \Omega$. This implies that the Arnoldi vectors $\mathbf{V}_\omega$ for $\omega \in \Psi \backslash \Omega$ are approximated on an iteratively built subspace based on $\mathbf{V}_\omega$ for $\omega \in \Omega$. Nevertheless, although the basis consists of information on the level of the Arnoldi vectors, the criterion utilized to control the quality of the basis does not concern the similarity of the

Algorithm 3 Automatic Krylov recycling (AKR)

```
Input: $\Psi \leftarrow [\omega_{\min}, \omega_{\max}]$, stagnation parameters $\epsilon_{\text{stag}}, \alpha \in \mathbb{R}$ and $S \in \mathbb{N}$, residual tolerance $r_{\text{tol}} \in \mathbb{R}$ and an initial guess of the solution $\mathbf{x}_0(\omega) : \Psi \to \mathbb{C}^N$;
$\omega_a \leftarrow \omega_{\min}$; $\omega_b \leftarrow \omega_{\max}$; $\Omega \leftarrow \{\omega_a\} \cup \{\omega_b\}$;
Deploy Algorithm 1 for $\omega_a$ and $\omega_b$ with settings $r_{\text{tol}_1} \leftarrow r_{\text{tol}}$ and $r_{\text{tol}_2} \leftarrow \alpha r_{\text{tol}}$ to construct $\mathbf{V}_{\omega_a, r_{\text{tol}_1}} \leftarrow [\mathbf{v}_{\omega_a,1} \ldots \mathbf{v}_{\omega_a,s(\omega_a)}], \mathbf{V}_{\omega_b, r_{\text{tol}_1}} \leftarrow [\omega_{\mathbf{a}},_{\omega_b,1} \ldots \mathbf{v}_{\omega_b,s(\omega_b)}]$ and $\mathbf{V}_{\omega_a, r_{\text{tol}_2}} \leftarrow [\mathbf{V}_{\omega_a, r_{\text{tol}_1}} \quad \mathbf{v}_{\omega_a,s(\omega_a)+1} \ldots \mathbf{v}_{\omega_a,p(\omega_a)}]$, $\mathbf{V}_{\omega_b, r_{\text{tol}_2}} \leftarrow [\mathbf{V}_{\omega_b, r_{\text{tol}_1}} \quad \mathbf{v}_{\omega_b,s(\omega_b)+1} \ldots \mathbf{v}_{\omega_b,p(\omega_b)}]$ respectively;
Construct the initial orthogonal augmentation basis $\mathbf{W}$ by QR decomposition on $[\mathbf{V}_{\omega_a, r_{\text{tol}_1}} \quad \mathbf{V}_{\omega_b, r_{\text{tol}_1}}]$;
**for** all $\omega_m \leftarrow 0.5(\omega_a + \omega_b)$, where $\omega_a, \omega_b$ consecutive values of $\Omega$ **do**
    $i \leftarrow 1$;    $\mathbf{r}_0(\omega_m) \leftarrow \mathbf{b}(\omega_m) - \mathbf{A}(\omega_m)\mathbf{x}_0(\omega_m)$;
    $\hat{\mathbf{x}}_i(\omega_m) \leftarrow \mathbf{W}(\mathbf{W}^H \mathbf{A}(\omega_m)\mathbf{W})^{-1}\mathbf{W}^H \mathbf{r}_0(\omega_m)$;
      $\mathbf{r}_i(\omega_m) \leftarrow \mathbf{r}_0(\omega_m) - \mathbf{A}(\omega_m)\hat{\mathbf{x}}_i(\omega_m)$;
    **while** $\|\mathbf{r}_i(\omega_m)\| > r_{\text{tol}}$ **do**
        **if** *Stagnation* $\neq 1$ **then**
            $i \leftarrow i + 1$
            Orthogonalize $\mathbf{v}_{\omega_a,s(\omega_a)+1}, \mathbf{v}_{\omega_b,s(\omega_b)+1}$ to $\mathbf{W}$ to obtain
          $\mathbf{v}^{\perp \mathbf{W}}_{\omega_a,s(\omega_a)+1}, \mathbf{v}^{\perp \mathbf{W}}_{\omega_b,s(\omega_b)+1}$
            Enrich augmentation basis: $\mathbf{W} \leftarrow [\mathbf{W} \quad \mathbf{v}^{\perp \mathbf{W}}_{\omega_a,s(\omega_a)+1} \quad \mathbf{v}^{\perp \mathbf{W}}_{\omega_b,s(\omega_b)+1}]$
            $\hat{\mathbf{x}}_i(\omega_m) \leftarrow \mathbf{W}(\mathbf{W}^H \mathbf{A}(\omega_m)\mathbf{W})^{-1}\mathbf{W}^H \mathbf{r}_0(\omega_m)$;
          $\mathbf{r}_i(\omega_m) \leftarrow \mathbf{r}_0(\omega_m) - \mathbf{A}(\omega_m)\hat{\mathbf{x}}_i(\omega_m)$;
            $s(\omega_a) \leftarrow s(\omega_a) + 1$;    $s(\omega_b) \leftarrow s(\omega_b) + 1$;
        **else**
            Run Algorithm 2 with $\omega \leftarrow \omega_m, r_{\text{tol}_1}, r_{\text{tol}_2}$ to obtain
          $\mathbf{W}_{\omega_m,\text{tol}_1}, \mathbf{W}_{\omega_m,\text{tol}_2}, \mathbf{V}_{\omega_m,\text{tol}_1}, \mathbf{V}_{\omega_m,\text{tol}_2}$;
            $\mathbf{W} \leftarrow \mathbf{W}_{\omega_m,\text{tol}_1}$;    $\Omega \leftarrow \Omega \cup \{\omega_m\}$;
        **end if**
    **end while**
**end for**
Output: global basis $\mathbf{W} \in \mathbb{C}^{N \times \ell}$
```

true Arnoldi vectors $\mathbf{V}_{\omega_m}$ for $\omega_m \in \Psi \backslash \Omega$ to the potentially approximated ones, but rather the distance of the true solution $\mathbf{x}(\omega_m)$ to the approximated $\hat{\mathbf{x}}(\omega_m)$.

Since the objective of the algorithm is to define a projection basis $\mathbf{W}$ that is able to provide an approximation of the solution $\hat{\mathbf{x}}(\omega), \omega \in \Psi$ of sufficient quality dictated by r_{tol}, it is not useful to employ techniques that potentially accelerate the convergence of the solution approximations of single systems. As implied by (9), leveraging a good initial guess can lead to faster convergence of the approximated solution, however this would undermine the global character of the basis as $\mathbf{x}_0(\omega)$ takes a different value for each $\omega \in \Psi$. Therefore, since guaranteeing the quality of $\mathbf{x}_0(\omega)$ for all $\omega \in \Psi$ is not straightforward, it is important that the basis is built with the most conservative approach of not having any good initial guess available, thus selecting $\mathbf{x}_0(\omega) := 0$ for all $\omega \in \Psi$. However, Algorithm 3 is expressed in the general form including an initial guess $\mathbf{x}_0(\omega)$ to accommodate cases such as parametrizations in boundary conditions as reported in [42].

Attempting to approximate $\mathbf{x}(\omega)$ for all $\omega \in \Psi$ leaves the method vulnerable to memory excess while the augmentation basis $\mathbf{W}$ becomes larger. Although restarting techniques could arise as a potential option to alleviate any memory concerns, losing certain information after each restart ruins the global character of the basis and thus is not desirable. Alleviating potential memory concerns, a memory constrained alternative of the proposed AKR algorithm is introduced in section 3.2. Finally, as AKR is based on a FOM procedure, the efficiency becomes greater for systems that demonstrate good clustering of their eigenvalues as a lower number of Arnoldi vectors is required. Increasing the efficiency of the presented technique also for systems that do not fulfil the above requirement could occur by the combination with a suitable parameter dependent preconditioner, which nonetheless is not examined in this work.

3.2. Memory constrained automatic Krylov recycling

As the required memory can prove to be the bottleneck for the Automatic Krylov Recycling algorithm, it is useful to define an alternative that takes into consideration a memory constraint and sets a maximum dimension of the global projection basis. The constraint is applied by the means of an integer $\ell \in \mathbb{N}$ that indicates the maximum vectors allowed in a projection basis $\mathbf{W} = \mathrm{span}\{\mathcal{W}\}$. Thus, the objective of the memory constrained algorithm is the construction of $\mathbf{W}$ to ensure the residual

$$\mathbf{r}(\omega) = \mathbf{b}(\omega) - \mathbf{A}(\omega)\hat{\mathbf{x}}(\omega) \perp \mathbf{W}, \tag{14}$$

$$\hat{\mathbf{x}}(\omega) \in \mathbf{x}_0(\omega) + \mathcal{W}, \tag{15}$$

satisfies $\|\mathbf{r}(\omega)\| \leq r_{\mathrm{tol}} \quad \forall \omega \in \Psi_\ell \subseteq \Psi$, where Ψ_ℓ is determined by the algorithm such that

$$\mathrm{rank}(\mathbf{W}) \leq \ell. \tag{16}$$

Deploying this constraint, the memory requirement ℓ is linked to the parameter interval Ψ_ℓ for which $\mathbf{W}$ provides sufficiently accurate approximations of the solutions $\mathbf{x}(\omega)$. The algorithm is outlined in Algorithm 4

Algorithm 4 initiates with the same steps as Algorithm 3. The augmentation basis is iteratively enriched until either condition (16) is violated or $\|\mathbf{r}(\omega_m)\| \leq r_{\mathrm{tol}}$, where ω_m is defined by any two consecutive elements $\omega_a, \omega_b \in \Omega$ by $\omega_m := 0.5(\omega_a + \omega_b)$. In the former case, Ω is truncated by its lower end $\min(\Omega)$, i.e. $\Omega \leftarrow \Omega \backslash \{\min(\Omega)\}$, and the basis $\mathbf{W}$ is reassembled by orthogonalizing the Arnoldi vectors $\mathbf{V}_{\omega, r_{\mathrm{tol}_1}} = \mathrm{span}\{\mathcal{K}^{\omega}_{s(\omega)}\}$ of the remaining $\omega \in \Omega$. Then, after executing Algorithm 2 for the new $\omega_{\min} := \min(\Omega)$ for settings r_{tol_1} and r_{tol_2} to ensure $\|\mathbf{r}(\omega_{\min})\| \leq r_{\mathrm{tol}}$, the enrichment proceeds for the newly determined Ω and $\Psi_\ell := [\omega_{\min}, \omega_{\max}]$. Proceeding to the truncation of the interval Ψ from its lower end is justified as by default defining ω_m starts for the upper end of Ω.

In case condition (16) is fulfilled for the preselected parameter range Ψ, either the algorithm is ceased having ensured that the basis is valid for the whole Ψ, or the basis can be further enriched with additional Arnoldi vectors and extend Ψ in a forward manner. In detail, the algorithm proceeds by checking

$\omega_m := \omega_{\min} - \Delta\omega$, where $\Delta\omega := \min(\Omega \backslash \{\omega_{\min}\}) - \omega_{\min}$ being the distance of the two smallest elements of Ω.

Algorithm 4 Memory constrained automatic Krylov recycling

```
Input: $\Psi \leftarrow [\omega_{\min}, \omega_{\max}]$, stagnation parameters $\epsilon_{\text{stag}}, \alpha \in \mathbb{R}$ and $S \in \mathbb{N}$, residual tolerance $r_{\text{tol}} \in \mathbb{R}$, an initial guess of the solution $\mathbf{x}_0(\omega) : \Psi \rightarrow \mathbb{C}^N$ and memory parameter $\ell \in \mathbb{N}$ ;
$\omega_a \leftarrow \omega_{\min}$; $\omega_b \leftarrow \omega_{\max}$; $\Omega \leftarrow \{\omega_a\} \cup \{\omega_b\}$;
Deploy Algorithm 1 for $\omega_a$ and $\omega_b$ with settings $r_{\text{tol}_1} \leftarrow r_{\text{tol}}$ and $r_{\text{tol}_2} \leftarrow \alpha r_{\text{tol}}$ to construct $\mathbf{V}_{\omega_a, r_{\text{tol}_1}} \leftarrow [\mathbf{v}_{\omega_a,1} \dots \mathbf{v}_{\omega_a,s(\omega_a)}]$, $\mathbf{V}_{\omega_b, r_{\text{tol}_1}} \leftarrow [\omega_{\mathbf{a}},_{\omega_b,1} \dots \mathbf{v}_{\omega_b,s(\omega_b)}]$ and $\mathbf{V}_{\omega_a, r_{\text{tol}_2}} \leftarrow [\mathbf{V}_{\omega_a, r_{\text{tol}_1}} \quad \mathbf{v}_{\omega_a,s(\omega_a)+1} \dots \mathbf{v}_{\omega_a,p(\omega_a)}]$, $\mathbf{V}_{\omega_b, r_{\text{tol}_2}} \leftarrow [\mathbf{V}_{\omega_b, r_{\text{tol}_1}} \quad \mathbf{v}_{\omega_b,s(\omega_b)+1} \dots \mathbf{v}_{\omega_b,p(\omega_b)}]$ respectively;
Construct the initial orthogonal augmentation basis $\mathbf{W}$ by QR decomposition on $[\mathbf{V}_{\omega_a, r_{\text{tol}_1}} \quad \mathbf{V}_{\omega_b, r_{\text{tol}_1}}]$;
for all $\omega_m \leftarrow 0.5(\omega_a + \omega_b)$, where $\omega_a, \omega_b$ consecutive values of $\Omega$ do
    $i \leftarrow 1$;    $\mathbf{r}_0(\omega_m) \leftarrow \mathbf{b}(\omega_m) - \mathbf{A}(\omega_m)\mathbf{x}_0(\omega_m)$;
    $\hat{\mathbf{x}}_i(\omega_m) \leftarrow \mathbf{W}(\mathbf{W}^H \mathbf{A}(\omega_m)\mathbf{W})^{-1}\mathbf{W}^H \mathbf{r}_0(\omega_m)$;
        $\mathbf{r}_i(\omega_m) \leftarrow \mathbf{r}_0(\omega_m) - \mathbf{A}(\omega_m)\hat{\mathbf{x}}_i(\omega_m)$;
    while $\|\mathbf{r}_i(\omega_m)\| > r_{\text{tol}}$ do
        if *Stagnation* $\neq 1$ then
            $i \leftarrow i + 1$
            Orthogonalize $\mathbf{v}_{\omega_a,s(\omega_a)+1}, \mathbf{v}_{\omega_b,s(\omega_b)+1}$ to $\mathbf{W}$ to obtain $\mathbf{v}^{\perp \mathbf{W}}_{\omega_a,s(\omega_a)+1}, \mathbf{v}^{\perp \mathbf{W}}_{\omega_b,s(\omega_b)+1}$
            Enrich augmentation basis: $\mathbf{W} \leftarrow [\mathbf{W} \quad \mathbf{v}^{\perp \mathbf{W}}_{\omega_a,s(\omega_a)+1} \quad \mathbf{v}^{\perp \mathbf{W}}_{\omega_b,s(\omega_b)+1}]$
            $\hat{\mathbf{x}}_i(\omega_m) \leftarrow \mathbf{W}(\mathbf{W}^H \mathbf{A}(\omega_m)\mathbf{W})^{-1}\mathbf{W}^H \mathbf{r}_0(\omega_m)$;
        $\mathbf{r}_i(\omega_m) \leftarrow \mathbf{r}_0(\omega_m) - \mathbf{A}(\omega_m)\hat{\mathbf{x}}_i(\omega_m)$;
            $s(\omega_a) \leftarrow s(\omega_a) + 1$;    $s(\omega_b) \leftarrow s(\omega_b) + 1$;
        else
            Run Algorithm 2 with $\omega \leftarrow \omega_m, r_{\text{tol}_1}, r_{\text{tol}_2}$ to obtain $\mathbf{W}_{\omega_m,\text{tol}_1}, \mathbf{W}_{\omega_m,\text{tol}_2}, \mathbf{V}_{\omega_m,\text{tol}_1}, \mathbf{V}_{\omega_m,\text{tol}_2}$;
            $\mathbf{W} \leftarrow \mathbf{W}_{\omega_m,\text{tol}_1}$;    $\Omega \leftarrow \Omega \cup \{\omega_m\}$;
        end if
```

```
19:     end while
20:     if rank($\mathbf{W}$) $> \ell$ then
21:         $\Omega \leftarrow \Omega \backslash \{\omega_{\min}\}$;   $\omega_{\min} \leftarrow \min(\Omega)$;
22:         Reassemble $\mathbf{W}$ with $\mathcal{W} \leftarrow \mathcal{K}^{\omega_1}_{s(\omega_1)} \cup \cdots \cup \mathcal{K}^{\omega_{\max}}_{s(\omega_{\max})}$, with
            $\omega_1, \ldots, \omega_{\max} \in \Omega$;
23:         Run Algorithm 2 with $\omega \leftarrow \omega_{\min}, r_{\mathrm{tol}_1}, r_{\mathrm{tol}_2}$ to obtain
            $\mathbf{W}_{\omega_m,\mathrm{tol}_1}, \mathbf{W}_{\omega_m,\mathrm{tol}_2}, \mathbf{V}_{\omega_m,\mathrm{tol}_1}, \mathbf{V}_{\omega_m,\mathrm{tol}_2}$;
24:         $\Psi_\ell \leftarrow [\omega_{\min}, \omega_{\max}]$
25:     else
26:         $\Delta\omega \leftarrow \min(\Omega \backslash \{\omega_{\min}\}) - \omega_{\min}$
27:         Repeat 6- 19 setting $\omega_m \leftarrow \omega_{\min} - \Delta\omega$ until rank($\mathbf{W}$) $\geq \ell$
28:         $\omega_{\min} \leftarrow \omega_{\min} - \Delta\omega$;   $\Psi_\ell \leftarrow [\min(\Omega), \max(\Omega)]$;
29:     end if
30: end for
31: Output: global basis $\mathbf{W} \in \mathbb{C}^{N \times \ell}$ and validity range $\Psi_\ell$
```

As seen in Step 22 of the algorithm, in case rank($\mathbf{W}$) $> \ell$, $\mathbf{W}$ is reassembled eliminating the components associated with $\omega_{\min}$ which is also removed from Ω. This procedure implies that the respective Arnoldi vectors for $\omega \in \Omega$ need to be stored along with the order of vectors $s(\omega), \omega \in \Omega$ to enable their later reutilization in the reassembly step. At this step, a QR decomposition is employed to create an orthonormal basis by these Arnoldi vectors. As this reinitialization only occurs for a moderate amount of times, the accompanying cost is not considered significant.

4. Combination of the AKR with model order reduction

Guaranteeing r_{tol} for $\omega \in \Psi$ of (7) can be either leveraged in a computational scheme or in combination with a model order reduction scheme. In the first case, the approximate solution obtained by a projection on the reduction basis $\mathbf{W}$ can be employed as the initial guess for an iterative solution scheme, while in the latter the sequence of linear systems is reduced to a reduced-order parametric system, gaining significantly in the computational cost required both for the assembly and solution of each individual system. In this section the combination of the automatic Krylov recycling technique with a model order reduction scheme for non-affine parametric systems is elaborated. First, a Galerkin projection is demonstrated in combination with an appropriate polynomial approximation of the system. Subsequently, the proposed model order reduction technique is reviewed in the context of Boundary Elements.

4.1. Galerkin Projection of non-affine parametric linear systems

Deploying the Galerkin projection to a linear system is based on the approximation of the true solution $\mathbf{x}$ as a linear combination of the basis vectors of $\mathbf{W}$ to yield an approximated solution

$$\mathbf{x}(\omega) \approx \hat{\mathbf{x}}(\omega) = \mathbf{x}_0(\omega) + \mathbf{W}\mathbf{y}(\omega), \tag{17}$$

where $\mathbf{y}(\omega)$ are the degrees of freedom of the reduced system 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 (17) in (7) and multiplying from the left with the conjugate transpose of the projection matrix $\mathbf{W}^H$, yields the reduced system

$$\mathbf{W}^H\mathbf{A}(\omega)\mathbf{W}\mathbf{y}(\omega) = \mathbf{W}^H\mathbf{b}(\omega), \tag{18}$$

which can be solved more efficiently than the full system. This definition of the reduced system in (18) and approximate solution $\hat{\mathbf{x}}$ is similar to the one introduced in (8) and (9) employed as the objectives of AKR in section 3. Thus, AKR can be leveraged in the offline stage of a MOR technique to obtain the reduction basis $\mathbf{W}$.

Nevertheless, having $\mathbf{W}$ does not lead to a straightforward projection procedure, since as demonstrated also in [30] the sequence of system assembly and projection is algorithmically quite important. This holds especially regarding non-affine parametric systems, where $\mathbf{A}$ and $\mathbf{b}$ are not of the form

$\mathbf{A}(\omega) = \sum_{k=1}^{K} \Theta_k(\omega)\mathbf{A}_k$ with $\mathbf{A}_k \in \mathbb{C}^{N\times N}$ and $\Theta_k : \Psi \to \mathbb{C}$, implying that each system needs first to be assembled and then projected, thus requiring a full system assembly for each $\omega \in \Psi$. Instead, in case the projection basis $\mathbf{W}$ is already constructed, it is possible to reverse this sequence by first leveraging a polynomial approximation of the system so that it is expressed in an affine way. Given that $\mathbf{A}(\omega), \mathbf{b}(\omega)$ are analytic functions of ω in Ψ, the system can be approximated by polynomial expansions of orders $M_{\max}$ and $q_{\max}$, rendering the parametric dependency affine as

$$\Big(\sum_{i=0}^{M_{\max}} c_i(\omega)\mathbf{A}_i\Big)\mathbf{x}(\omega) = \sum_{q=0}^{q_{\max}} g_q(\omega)\mathbf{b}_q, \tag{19}$$

where $\mathbf{A}_i \in \mathbb{C}^{N\times N}, \mathbf{b}_q \in \mathbb{C}^N$ are the parameter-independent system coefficients and c_i, g_q the scalar polynomial functions of ω. The type of the selected polynomial approximations depends on the analytic functions that are approximated and the respective orders $M_{\max}$ and $q_{\max}$ can be determined *a-priori* in many cases [30, 43]. Reversing the order of assembling and projecting is conducted by replacing (19) in (18) and shifting the summation operation to obtain a Galerkin projected matrix polynomial as

$$\begin{aligned} \sum_{i=0}^{M_{\max}} c_i(\omega)\mathbf{W}^H\mathbf{A}_i\mathbf{W}\mathbf{y}(\omega) &= \sum_{q=0}^{q_{\max}} g_q(\omega)\mathbf{W}^H\mathbf{b}_q \Leftrightarrow \\ \sum_{i=0}^{M_{\max}} c_i(\omega)\mathbf{A}_{i,red}\mathbf{y}(\omega) &= \sum_{q=0}^{q_{\max}} g_q(\omega)\mathbf{b}_{q,red}, \end{aligned} \tag{20}$$

where $\mathbf{A}_{i,red} \in \mathbb{C}^{\ell\times\ell}$ and $\mathbf{b}_{q,red} \in \mathbb{C}^{\ell}$ are the coefficients of the reduced polynomial system and $\ell := \text{rank}(\mathbf{W})$. In such a manner, only assembly and projection of the reduced system takes place, avoiding the assembly of the full system for each $\omega \in \Psi$. Finally, in case $\mathbf{W}$ is not available beforehand, the polynomial expansion of the system might inflict memory excess as all $\mathbf{A}_i$ need to be stored simultaneously.

4.2. Memory constrained MOR for a broad parameter domain

Transforming the system of (7) in a matrix polynomial as in equation (19) requires storing all the matrix coefficients of the system. Although the storage

Algorithm 5 Memory constrained MOR for non-affine parametric systems with analytic matrix entries

```
Input $\Psi_{\text{tot}} \leftarrow [\omega_{\min}, \omega_{\max}]$, memory parameter $\ell \in \mathbb{N}$, residual tolerance $r_{\text{tol}} \in \mathbb{R}$, stagnation parameters $\alpha, \epsilon_{\text{stag}} \in \mathbb{R}$ and $S \in \mathbb{N}$;
$\Psi \leftarrow \Psi_{\text{tot}}$;
Run Algorithm 4 with input $\Psi, r_{\text{tol}}, \alpha, S, \epsilon_{\text{stag}}$ to obtain $\Psi_{\ell_1}, \mathbf{W}_1$;
$i \leftarrow 1$;
while $\Psi_{\ell_i} \subset \Psi$ do
    $\Psi \leftarrow \Psi \backslash \Psi_{\ell_i}$;    $i \leftarrow i + 1$;
    Run Algorithm 4 with input $\Psi, r_{\text{tol}}, \alpha, S, \epsilon_{\text{stag}}$ to obtain $\Psi_{\ell_i}, \mathbf{W}_i$;
end while
$Q \leftarrow i$;
Output $\mathbf{W}_1 \in \mathbb{C}^{N \times \ell_1}, \ldots, \mathbf{W}_Q \in \mathbb{C}^{N \times \ell_Q}$ and $\Psi_{\ell_1}, \ldots, \Psi_{\ell_Q}$ where $\Psi_{\text{tot}} = \Psi_{\ell_1} \cup \cdots \cup \Psi_{\ell_Q}$
```

requirements are considerably relaxed by exploiting a projection of the coefficient matrices $\mathbf{A}_i$, they might still prove to be the bottleneck regarding the proposed AKR-MOR technique. In this case the memory constrained alternative of AKR in combination with a windowing technique can prove beneficial to alleviate and control the memory required for the simultaneous storage the projected matrices.

The proposed technique accepts as input the target interval Ψ_{tot}, for which the model reduction scheme needs to provide high quality approximations, and a maximum dimension of the reduction basis ℓ. After executing Algorithm 4, a truncated parameter interval Ψ_ℓ is yielded for which the memory constrained basis ensures a predefined residual tolerance r_{tol}. The same procedure is sequentially deployed for the remaining interval $\Psi \leftarrow \Psi \backslash \Psi_\ell$ until the whole interval Ψ_{tot} is covered, namely until $\Psi_{\text{tot}} = \Psi_{\ell_1} \cup \cdots \cup \Psi_{\ell_Q}$, where Q is the number of subintervals. This procedure is outlined in Algorithm 5.

4.3. The AKR for model reduction of acoustic BEM problems

The above technique can be utilized in the context of the conventional acoustic Boundary Element Method. The linear system that is yielded as a result of a Boundary Element approximation takes the form of

$$\mathbf{H}(k)\mathbf{p}(k) = \mathbf{G}(k)\mathbf{u}(k), \quad k \in \Psi, \tag{21}$$

where $\mathbf{p}, \mathbf{u}$ are the pressure and normal velocity respectively, k the wavenumber related to frequency f with the speed of sound in the medium c as $k = \frac{2\pi f}{c}$ and $\Psi := [k_{\min}, k_{\max}] \subset \mathbb{R}$ a user input interval. $\mathbf{G}, \mathbf{H} : \Psi \to \mathbb{C}^{N \times N}$ are the non-affine parametric BEM system matrices arising from the discretization of the single and double layer potentials [33, 44]. After application of the known boundary conditions, equation (21) is transformed to a linear system

$$\mathbf{A}(k)\mathbf{x}(k) = \mathbf{b}(k), \tag{22}$$

where $\mathbf{A} : \Psi \to \mathbb{C}^{N \times N}, \mathbf{b} : \Psi \to \mathbb{C}^{N}$, which is similar to equation (7). Being a non-affine parametric system, the deployment of the proposed AKR-MOR technique is suitable. Additionally, as illustrated in [30] and [43] the BEM system matrices can be efficiently approximated by a matrix polynomial rendering the frequency dependency affine. This allows reversing the procedures of assembling and projection of the system on a representative basis enabling efficient frequency sweep analyses.

Furthermore, the deployment of the developed AKR-MOR technique is facilitated by the BEM system matrices that demonstrate well-clustered eigenvalues as shown in [32]. Thus, a potential deployment of the full orthogonalization method or GMRES converges within a moderate number of iterations even without preconditioning [35]. Thus, employing the AKR the dimension of the resulting final global projection basis will remain within reasonable extents.

Finally, concerning the deployment of the method for interior undamped BEM problems that normally include resonances, it is more appropriate to utilise a residual measure normalized against a quantity, which is also affected by the existence of resonances. Using the common normalized residual might inflict a local excess of the residual tolerance, which normally would not appear employing the true relative error, which is normalized with the true or approximated solution. For that reason, it is decided to proceed with a residual quantity normalized against the approximated solution instead of the respective right hand-side as

$$\mathbf{r}_{norm}(k) := \frac{\mathbf{r}(k)}{\|\hat{\mathbf{x}}(k)\|}, \tag{23}$$

with $\hat{\mathbf{x}}(k)$ being the approximated solution and $\mathbf{r}(k)$ the respective residual.

4.4. An alternative reduced basis method for the assessment of the AKR-MOR

As elaborated in section 3, AKR can be seen as an alternative to the reduced basis method. The basic difference lies on the input taken for the construction of the projection basis, as the latter utilizes the full solutions at an ordered set of parameter values Ω, whereas the former employs vectors that lead to sufficient approximations of the full solution i.e. the Arnoldi vectors at Ω. This implies on the one hand that the yielded basis will span a larger set of directions but on the other hand, employing the same Ω, it will be more efficient for a broader interval.

The second difference can be located in the error estimator that the two techniques employ. Although they both leverage a residual based error estimator, the reduced basis method necessitates the calculation of the residual for a fine grid of parameter values $\Phi \supset \Omega$, with $|\Phi| \gg |\Omega|$ to pinpoint the parameter value for which the worst approximated solution is yielded, while the AKR assumes that the highest error lies at ω_m and thus enforces the desired residual accuracy r_{tol} at ω_m, where $\omega_m := 0.5(\omega_a + \omega_b)$ with ω_a and ω_b being two consecutive elements of Ω. Computing the residual on Φ can be relatively simple for parametric systems with affine dependency, but in case of non-affine parametric dependencies, full system assemblies are required for all the values of Φ, inducing a high cost for each iteration of the reduced basis method.

Algorithm 6 Alternative reduced basis method

```
Input $\Psi \leftarrow [\omega_{\min}, \omega_{\max}]$ and $r_{\text{tol}} \in \mathbb{R}$
$\omega_a \leftarrow \omega_{\min}$ and $\omega_b \leftarrow \omega_{\max}$;
Construct full solutions at $\omega_a, \omega_b$, $\mathbf{x}(\omega_a)$ and $\mathbf{x}(\omega_b)$;
Construct an orthogonal basis $\mathbf{W}$ by a QR decomposition on $[\mathbf{x}(\omega_a) \quad \mathbf{x}(\omega_b)]$;
$\Omega_{\text{RBM}} \leftarrow \{\omega_a\} \cup \{\omega_b\}$;
for all $\omega_m \leftarrow 0.5(\omega_a + \omega_b)$, where $\omega_a, \omega_b$ consecutive values of $\Omega_{\text{RBM}}$ do
    $\hat{\mathbf{x}}(\omega_m) \leftarrow \mathbf{W}(\mathbf{W}^H \mathbf{A}(\omega_m)\mathbf{W})^{-1}\mathbf{W}^H \mathbf{b}(\omega_m)$;
        $\mathbf{r}(\omega_m) \leftarrow \mathbf{b}(\omega_m) - \mathbf{A}(\omega_m)\hat{\mathbf{x}}(\omega_m)$;
    if $\|\mathbf{r}(\omega_m)\| > r_{\text{tol}}$ then
        $\mathbf{x}(\omega_m) \leftarrow \mathbf{A}^{-1}(\omega_m)\mathbf{b}(\omega_m)$
        Orthogonalize $\mathbf{x}(\omega_m)$ to $\mathbf{W}$ to obtain $\mathbf{x}^{\perp \mathbf{W}}(\omega_m)$
        $\mathbf{W} \leftarrow [\mathbf{W} \quad \mathbf{x}^{\perp \mathbf{W}}(\omega_m)]$;    $\Omega_{\text{RBM}} \leftarrow \Omega_{\text{RBM}} \cup \{\omega_m\}$;
    end if
end for
Output global basis $\mathbf{W} \in \mathbb{C}^{N \times \ell}$
```

The assumption made in the AKR is valid due to the broadband convergence behaviour offered by the recycling of the components of the full solutions. Making the same assumption in the context of the reduced basis approach, an alternative Reduced Basis Method (RBM) can be devised that constructs a reduced basis $\mathbf{W} = \text{span}\{\mathcal{W}\}$ to ensure that approximate solutions $\hat{\mathbf{x}}(\omega) \in \mathcal{W}$ of the system satisfy $\|\mathbf{r}(\omega)\| \leq r_{\text{tol}} \quad \forall \quad \omega \in \Phi \subset \Psi$, where Φ is a discrete set with $|\Phi| = N_{\text{tot}}$. Attempting to keep the number of systems assembled to a minimum, the basis $\mathbf{W}$ is constructed by checking the residual condition only at $\omega_m := 0.5(\omega_a + \omega_b)$, where $\omega_a, \omega_b \in \Omega_{\text{RBM}} \subset \Psi$ and Ω_{RBM} is a discrete ordered set with $|\Omega_{\text{RBM}}| \ll |\Phi|$. In case this condition is not satisfied, it appends the full solution of the system $\mathbf{x}(\omega_m)$ to the reduction subspace, i.e. $\mathcal{W} \leftarrow \mathcal{W} \cup \{\mathbf{x}(\omega_m)\}$. In that way, each system is constructed only once and only if $\|\mathbf{r}(\omega_m)\| \leq r_{\text{tol}}$, $\mathbf{x}(\omega_m)$ is evaluated. However, given the relatively narrowband influence of appending $\mathbf{x}(\omega_m)$ to the reduction subspace $\mathcal{W}$, $\|\mathbf{r}(\omega)\| \leq r_{\text{tol}}$ is satisfied only in the direct vicinity of $\omega \in \Omega_{\text{RBM}}$. Thus, deploying both the alternative RBM and the AKR, it is expected that $|\Omega_{\text{AKR}}| < |\Omega_{\text{RBM}}|$. This implies that in comparison with the AKR, the alternative RBM requires not only the full assembly but also the full solution for a larger number of systems, consequently inducing a higher computational cost in the offline phase of a corresponding MOR technique. The procedure followed by the alternative RBM is presented in Algorithm 6.

5. Numerical Assessment

In this section the performance of the proposed AKR algorithm and the respective MOR techniques are investigated in terms of the quality of the yielded bases and the accompanying cost induced. The assessment is conducted by employing the full solutions as well as the alternative RBM for benchmarking purposes. The latter is utilised to bring AKR in the right context and to assess its performance with respect to a comparable competitive technique. The comparison is performed by evaluating the performance of both techniques on two numerical multiresolution parametric examples. All computations are performed on a single machine of 32GB RAM and 2.9GHz processing power.

5.1. AKR on a random system

In order to demonstrate the general applicability of the method, the first example constitutes a random system with a given single parameter non-affine dependency and predefined eigenvalue distribution. It is treated solely with the AKR technique, as the assumed parametric dependency is not analytic for each matrix entry. Since the model reduction is considered only after the assembly of the system, AKR is compared with other conventional system solution techniques as well as with an approximation yielded by a projection on an alternative RBM produced basis. The efficiency of the technique is assessed in terms of the resulting residual as well as the wall-clock time required for each method.

5.1.1. Construction and solution of the system

A random densely populated square system comprising of 2500 equations is constructed as

$$\mathbf{A}_{\text{rand}} := \text{rand}(2500) + j \cdot \text{rand}(2500), \tag{24}$$

where $\mathbf{A}_{\text{rand}} \in \mathbb{C}^{2500\times 2500}$, j is the imaginary unit and $\text{rand}(N)$ constitutes the Matlab® function that generates an $N \times N$ array of uniformly distributed values. As documented in section 3, AKR is more efficient for systems with good clustering of their eigenvalues. Thus, in order to obtain such a system, the eigenvalues of the matrix are manipulated accordingly. In detail, assuming $\mathbf{A}_{\text{rand}}$ is diagonalizable, after deploying the eigendecomposition

$$\mathbf{\Lambda} = \mathbf{\Theta}^{-1}\mathbf{A}_{\text{rand}}\mathbf{\Theta} \tag{25}$$

the diagonal matrix $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_N)$ containing the eigenvalues of the matrix λ_i, as well as the corresponding eigenvector matrix $\mathbf{\Theta}$ are obtained. The eigenvalues of the matrix are shifted by the highest eigenvalue magnitude as $\mathbf{\Lambda}_{\text{wc}} := \mathbf{\Lambda} + \mathbf{I}\max(\|\lambda_i\|), i := 1, \dots, N$, where $\mathbf{I}$ is the identity matrix, to guarantee that a reconstructed matrix $\mathbf{A}_{\text{wc}}$ is well-conditioned. To induce a non-affine parametric dependency to the system both eigenvectors $\mathbf{\Theta}$ and eigenvalues $\mathbf{\Lambda}_{\text{wc}}$ are perturbed with an analytic function of a parameter ω as

$$\begin{aligned}\mathbf{\Lambda}_{\text{pert}}(\omega) :=& \mathbf{\Lambda}_{\text{wc}} + \mathbf{Y}(\omega),\\ \mathbf{\Theta}_{\text{pert}}(\omega) :=& \mathbf{\Theta} + \mathbf{\Phi}(\omega).\end{aligned} \tag{26}$$

where $\mathbf{Y}(\omega) : \mathbb{R} \to \mathbb{C}^{N\times N}$ is a diagonal matrix and $\mathbf{\Phi}(\omega) : \mathbb{R} \to \mathbb{C}^{N\times N}$. The diagonal entry $\lambda_{i,\text{pert}}(\omega)$ of $\mathbf{\Lambda}_{\text{pert}}(\omega)$ depends on the diagonal entry y_i of $\mathbf{Y}$ by the analytic function

$$\lambda_{i,\text{pert}}(\omega) := \lambda_{i,\text{wc}} + e^{j y_i \omega}, \tag{27}$$

where $\lambda_{i,\text{wc}}$ is the diagonal entry of $\mathbf{\Lambda}_{\text{wc}}$ and ω a single dimension parameter in $\mathbb{R}$. The perturbed eigenvector $\boldsymbol{\theta}_{i,\text{pert}}$ is the i^{th} column of $\mathbf{\Theta}_{\text{pert}}$ and is defined as

$$\boldsymbol{\theta}_{i,\text{pert}}(\omega) := \boldsymbol{\theta}_i + e^{j\boldsymbol{\phi}_i \omega} \tag{28}$$

where $\boldsymbol{\theta}_i$ is the respective unperturbed eigenvector, $\boldsymbol{\phi}_i$ is the i^{th} column of $\boldsymbol{\Phi}$ and

$$e^{j\boldsymbol{\phi}_i \omega} := \begin{bmatrix} e^{j\phi_{1,i}\omega} \\ \vdots \\ e^{j\phi_{N,i}\omega} \end{bmatrix}. \tag{29}$$

In order to create different clusters of eigenvalues a monotonically increasing function $\xi(\omega) : \mathbb{R} \to \mathbb{R}$ is employed and $\lfloor \xi(\omega) \rfloor$ number of eigenvalues are scaled as

$$\lambda_{i,\text{recon}}(\omega) := \begin{cases} 10^3 \cdot \lambda_{i,\text{pert}}(\omega) & i \leq \xi(\omega) \\ \lambda_{i,\text{pert}}(\omega) & i > \xi(\omega), \end{cases} \tag{30}$$

where $\lambda_{i,\text{recon}}(\omega)$ is the i^{th} entry of the diagonal matrix $\boldsymbol{\Lambda}_{\text{recon}}(\omega)$. Finally, the system matrix is reconstructed as

$$\mathbf{A}_{\text{recon}}(\omega) = \boldsymbol{\Theta}_{\text{pert}}(\omega)\boldsymbol{\Lambda}_{\text{recon}}(\omega)\boldsymbol{\Theta}_{\text{pert}}^{-1}(\omega). \tag{31}$$

The system is completed by electing a random parameter dependent right hand-side as

$$\mathbf{b}_{\text{rand}}(\omega) := \begin{bmatrix} cos(b_1\omega) \\ \vdots \\ cos(b_N\omega) \end{bmatrix}, \tag{32}$$

where b_i is the i^{th} component of a vector $\mathbf{b}$ of uniformly distributed random values. The final parametric system is

$$\mathbf{A}_{\text{recon}}(\omega)\mathbf{x}(\omega) = \mathbf{b}_{\text{rand}}(\omega), \quad \omega \in \Psi \tag{33}$$

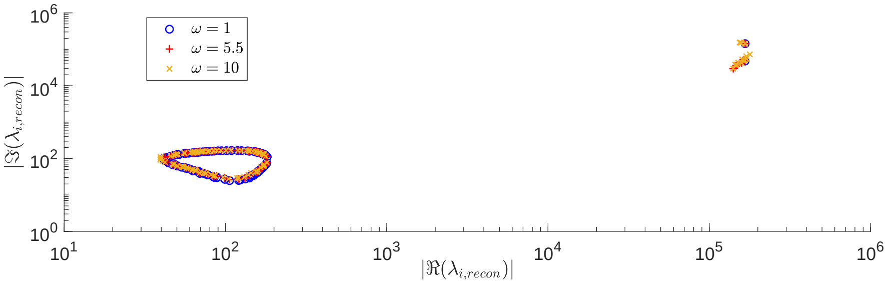


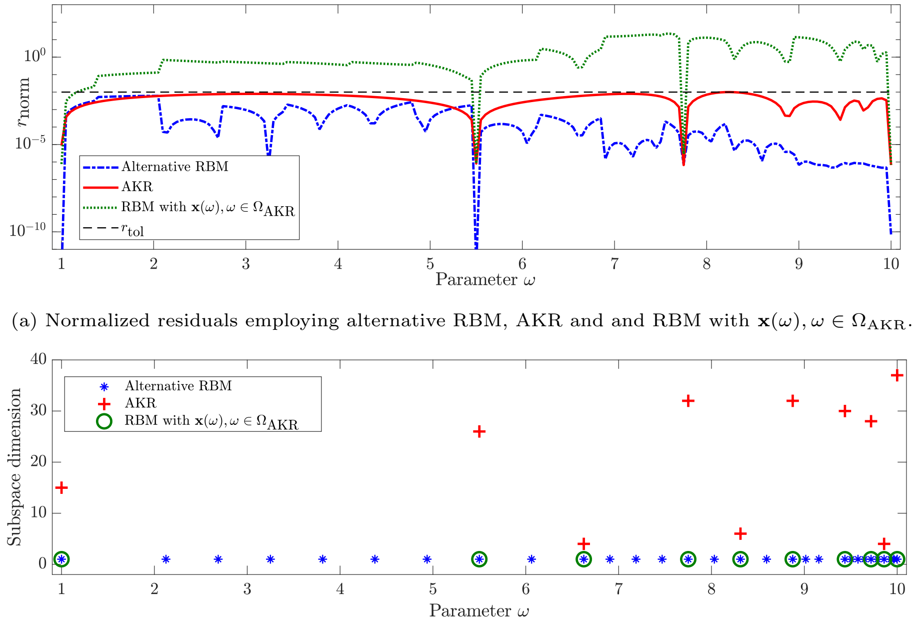
Figure 1: Eigenvalue distribution for random complex matrix of size 2500

where $\Psi := [1, 10]$. In Figure 1 the real and imaginary part of the eigenvalues $\lambda_{i,\mathrm{recon}}, i := 1, \ldots, 100$ are depicted for three values of $\omega \in \Psi$. One main cluster of eigenvalues appears to be dominant, while increasing the value of ω induces a more complicated spectrum with more eigenvalue clusters.

The parametric system of (33) is resolved by constructing reduction bases $\mathbf{W}_{\mathrm{AKR}}$ and $\mathbf{W}_{\mathrm{RBM}}$, leveraging AKR and the alternative RBM respectively with a residual tolerance $r_{\mathrm{tol}} := 10^{-2}$. The system is subsequently projected on the constructed bases and solved to obtain the approximate solutions $\hat{\mathbf{x}}_{\mathrm{AKR}}(\omega)$ and $\hat{\mathbf{x}}_{\mathrm{RBM}}(\omega)$ for all $\omega \in \Phi$. The set Φ is defined by discretizing Ψ with $\Delta\omega = 0.05$, resulting in $|\Phi| = 181$.

In Figure 2a the normalized residuals corresponding to $\hat{\mathbf{x}}_{\mathrm{AKR}}(\omega)$ and $\hat{\mathbf{x}}_{\mathrm{RBM}}(\omega)$ are shown. Alongside, an additional residual curve is plotted to indicate the quality of approximation, utilizing the full solutions at $\omega \in \Omega_{\mathrm{AKR}}$, which is the AKR sampled parameter values, in an RBM approach. In Figure 2b the sampled parameter values Ω_{AKR}, Ω_{AKR} are shown combined with the subspace dimension that is contributing to the reduction basis. While in the case of the AKR the respective subspace dimension indicates the number of orthogonalized Arnoldi vectors employed at each $\omega \in \Omega_{\mathrm{AKR}}$, the one-dimensional subspace involved in the two remaining procedures at each $\omega \in \Omega_{\mathrm{RBM}}$ and $\omega \in \Omega_{\mathrm{AKR}}$ implies that only the full solutions are inserted in the respective RBM approaches.

Inspecting Figure 2, AKR outperforms the two RBM approaches with respect to the level of residual achieved for the corresponding Ω employed. On



(a) Normalized residuals employing alternative RBM, AKR and and RBM with $\mathbf{x}(\omega), \omega \in \Omega_{\text{AKR}}$.

(b) Configurations for alternative RBM, AKR and RBM with $\mathbf{x}(\omega), \omega \in \Omega_{\text{AKR}}$.

Figure 2: Solution of random complex system of size 2500

the one hand, concerning the construction of the reduction basis, the alternative RBM approach necessitates full solutions at an Ω_{RBM}, with $|\Omega_{\text{RBM}}| \gg |\Omega_{\text{AKR}}|$ to yield a residual that lies below the predefined r_{tol}. In detail, the alternative RBM employs 57 full assemblies and 29 full solutions, while the AKR necessitates only 19 full assemblies and 10 partial solutions. Hence, not only does it result into assembling and solving more full systems than AKR, but it also offers an approximation that largely over-achieves the preselected residual level r_{tol}. On the other hand, constructing the reduction basis with an RBM procedure employing only the full solutions $\mathbf{x}(\omega), \omega \in \Omega_{\text{AKR}}$ provides approximations of considerably lower quality than the AKR. On the contrary, the AKR algorithm yields a residual curve that follows closely r_{tol}. Nevertheless, it cannot be overlooked that including subspaces of higher dimension at each $\omega \in \Omega_{\text{AKR}}$ as proposed by AKR, leads to a reduced basis which is larger than the one created in both RBM configurations. As presented in section 3.2, the efficient splitting of Ψ can be the remedy against a potentially high dimensionality of the reduced basis.

Due to the increasing complexity of the system at higher parameter values corresponding to additional eigenvalue clusters, both the alternative RBM and AKR generate a finer sampling grid Ω for $\omega > 5.5$. However, although both algorithms initiate their bisection procedure by default from the upper end of Ω, only in the RBM case the subspaces sampled at $\Omega_{\text{RBM}}, \omega < 5.5$ are increasingly influencing the accuracy of $\hat{\mathbf{x}}_{\text{RBM}}(\omega)$ for $\omega > 5.5$. This leads to the conclusion that splitting Ψ to several subintervals can be more efficient in case of the AKR, where iteratively adding the Arnoldi vectors at $\omega_m \in \Omega_{\text{AKR}}$ affects the accuracy of $\hat{\mathbf{x}}(\omega)$ only in the direct vicinity of ω_m.

5.1.2. Computational cost assessment

Apart from the efficiency of the proposed method with respect to the quality of the resulting approximations, its computational performance is examined. In Table 1 the computational costs of solving the system with the different methods are given in terms of the wall-clock time. Overall, the system assembly requires the highest amount of time and thus it follows that the cost for the construction of the projection bases $\mathbf{W}_{\text{AKR}}$ and $\mathbf{W}_{\text{RBM}}$ is lower for the AKR than leveraging the alternative RBM approach, as a result of $|\Omega_{\text{RBM}}| \gg |\Omega_{\text{AKR}}|$. This implies that AKR is also more computationally advantageous comparing to the conventional RBM. Although in that case a fewer number of full solutions might be required, as the construction of Ω_{RBM} is optimized, residuals for all $\omega \in \Phi$ are computed in each iteration, leading to multiple system assemblies per iteration.

Additionally, given the already constructed bases, a reduction of the cost is also apparent regarding the solution time per system comparing both to a direct solution scheme and the iterative solution strategy. Nevertheless, given the Φ under consideration and the fixed system assembly cost, which is identical for all solution methods, the overhead offline cost needed for the RBM and the AKR methods is not compensated. Thus, the sequential solution for each

Operation	Iterative solution (GMRES)	Direct solution	Alternative RBM solution	AKR solution
Total assembly time	13m	13m	13m	13m
Construction of projection basis	-	-	6m	4m
Solution per system	0.25s	0.65s	0.04s	0.12s
Total wall-clock time	13m 45s	15m	19m 7s	17m 22s

Table 1: Computational costs of different operations included in the solution of the complex random system of 2500 unknowns.

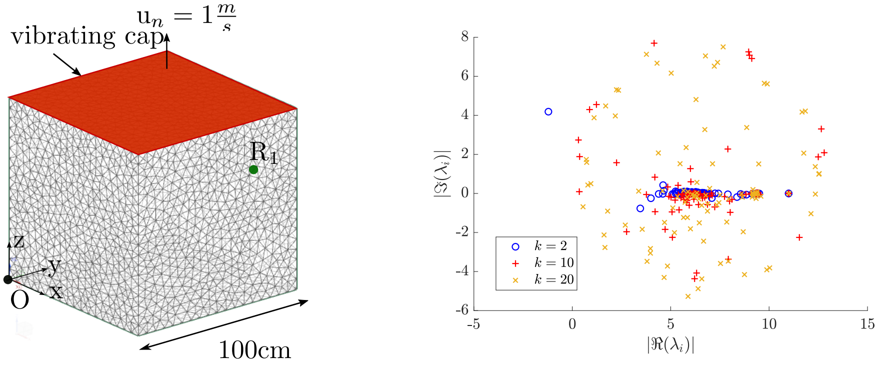
$\omega \in \Phi$ with an iterative method (GMRES) is the most economical strategy. Selecting a finer resolution grid Φ could lead in a straightforward manner to a clear computational advantage of the AKR. However, the gain can be more pronounced speeding up not only the solution time but also the system assembly time as proposed with the combination of AKR with MOR and demonstrated in the following example.

5.2. The AKR for the model reduction of an acoustic BEM system

In the second example AKR and the alternative RBM approaches are leveraged in the offline phase of a MOR technique, thus speeding up both the assembly and the solution of the corresponding systems. The parametric problem under consideration arises by requiring a frequency sweep analysis for a BEM academic test case. AKR is benchmarked against the alternative RBM and the full solution. Finally, this example is also solved with the proposed memory constrained AKR-MOR technique introducing a constraint regarding the maximum dimension of the projection basis. The assessment of the technique is conducted both in terms of the resulting residuals and the total wall clock time required.

5.2.1. Construction and solution of system

In this section AKR and the alternative RBM are leveraged to automatically produce the projection bases $\mathbf{W}_{\text{AKR}}$ and $\mathbf{W}_{\text{RBM}}$, utilized in a Galerkin model reduction scheme to reduce the cost of a multiresolution BEM simulation of an academic example. The model considered is the interior Helmholtz problem of a cube with a vibrating cap as in Figure 3a. Positioning the unit cube at the start of the axes $O := [0, 0, 0]$ and towards their positive direction, the receiver is located at $R_1 := [0.75, 0.75, 0.75]$. The cube consists of 2606 Degrees of Freedom (DOF), which is a discretization that, considering the requirement of 6 elements per wavelength [45], yields an accurate response for the wavenumber range of $k_{\text{range}} := [0, 22]m^{-1}$. The assembly and solution of BEM matrices is conducted in a Matlab environment by the OpenBEM code [46]. The spectrum of the resulting BEM system matrix for $k := 2, 10, 20$ is given in Figure 3b and consists of one big cluster of eigenvalues and an increasing number of additional clusters for an increasing value of k.



(a) Cube model

(b) Eigenvalue distribution

Figure 3: Cube model and spectral properties of system matrices

The frequency sweep is conducted for a $\Psi := [50, 1000]Hz$ and taking a $\Delta f := 1$Hz, it results in a grid of values $\Phi \subset \Psi$ with $|\Phi| = 951$, thus requiring the solution of 951 systems. The desired sweep is performed by employing both AKR and the alternative RBM for the offline phase of the MOR strategy and by the conventional BEM approach. In the offline phase of both MOR

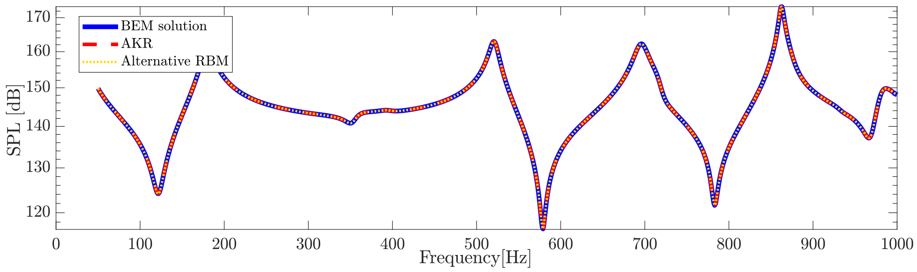
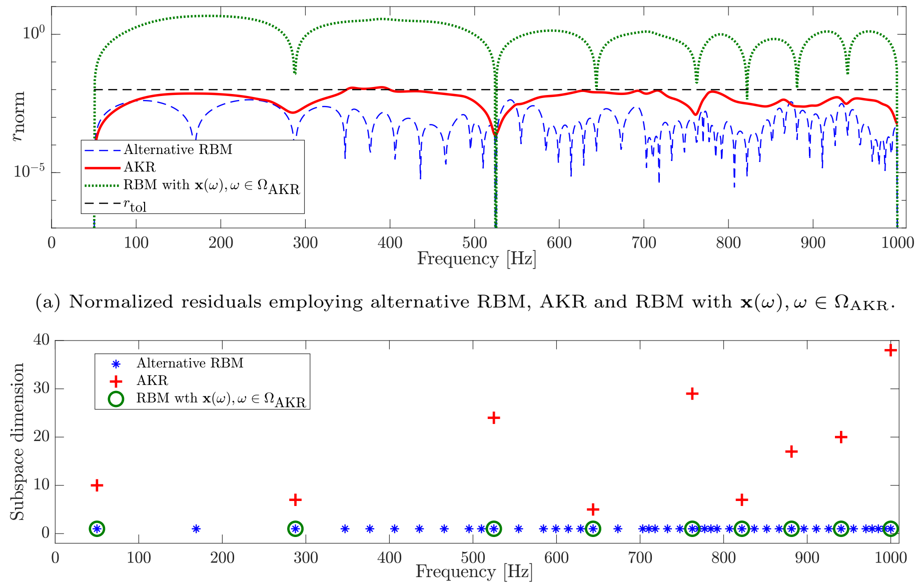


Figure 4: Response at R_1; Cube with vibrating cap

techniques the normalized residual for resonant problems (section 4.3) is used as error indicator and the residual tolerance is set at $r_{\text{tol}} := 10^{-2}$. Comparing the resulting response on R_1 for the AKR-MOR to the response acquired by a conventional BEM approach in Figure 4, no differences are discernible proving that the considered r_{tol} is sufficient.

In Figure 5 the residuals produced by the approximated solutions are shown together with their respective sampling patterns used for the construction of Ω_{AKR} and Ω_{RBM}. Comparing the AKR with the alternative RBM, although the approximated solutions $\hat{\mathbf{x}}_{\text{AKR}}$ and $\hat{\mathbf{x}}_{\text{RBM}}$ yield residuals of approximately the same quality, $|\Omega_{\text{RBM}}| \gg |\Omega_{\text{AKR}}|$ holds, denoting a computational advantage of AKR. Specifically, the alternative RBM requires 44 full solutions to assemble a basis $\mathbf{W}_{\text{RBM}}$, which can provide approximate solutions that satisfy r_{tol}, while the AKR requires only 9 partial solves to construct $\mathbf{W}_{\text{AKR}}$. Furthermore, inspecting the number of full assemblies, the alternative RBM requires 87 full assemblies, while the AKR requires only 17. Nevertheless, comparing the dimensions of the subspaces employed at Ω_{AKR} and Ω_{RBM}, AKR produces a basis spanning a subspace of 157 dimensions while the alternative RBM basis spans a subspace of only 44 dimensions, which is identical to the number of the full solutions required.

The advantage of using the Arnoldi vectors instead of the full solutions is evident in the comparison of the AKR with an RBM approach taking only the $\mathbf{x}(\omega), \omega \in \Omega_{\text{AKR}}$. Using the full solutions guarantees a low residual only in the



(a) Normalized residuals employing alternative RBM, AKR and RBM with $\mathbf{x}(\omega), \omega \in \Omega_{\text{AKR}}$.

(b) Configurations for alternative RBM, AKR and RBM with $\mathbf{x}(\omega), \omega \in \Omega_{\text{AKR}}$.

Figure 5: Residuals and sampling patterns fro the reduction of a BEM model of a cube with a vibrating cap

neighbourhood of the parameter $\omega \in \Omega_{\text{AKR}}$, while using the associated Arnoldi vectors leads to a satisfactory residual for all $\omega \in \Psi$. Of course, the difference in the size of the reduction basis between AKR and both RBM approaches cannot be disregarded. Nevertheless, this can be remedied by imposing a memory constraint as proposed in section 4.2 linking the maximum size of the basis with a frequency interval as is demonstrated in section 5.2.3.

5.2.2. *Computational cost assessment*

Similarly to the random system, the computational cost is also examined for the BEM system under consideration. As elaborated in section 4.3, the model reduction approach reduces the assembly time by first projecting a polynomial matrix approximating the system. On the contrary, the conventional BEM approach necessitates sequential assemblies for each individual $\omega \in \Phi$. The advantage of leveraging a MOR technique in combination with a kernel polynomial expansion has been demonstrated in [30], leading to significant acceleration in both the assembly and the solution of the system. Since AKR and RBM can be utilised as a technique to construct the projection basis $\mathbf{W}_{\text{AKR}}$ and $\mathbf{W}_{\text{RBM}}$ respectively, the focus here will be given to the cost related to the offline phase of the respective MOR techniques.

Operation	Alternative RBM solution	AKR solution	BEM
Construction of projection basis	12h 36m	2h 33m	-
Construction of Matrix Polynomial	7h 13m	7h 13m	-
Total Assembly time	2s	15s	137h 22m
Total Solution Time	1s	4s	19m
Total Time	19h 49m	9h 46m	137h 41m

Table 2: Computational costs of different operations included in the solution of the BEM system of the interior Cube problem.

In Table 2 the wall-clock times required by both the AKR and the alternative RBM are given in comparison with the respective cost of the conventional BEM. Due to the use of Matlab as the development environment, the cost of the system assembly ranks as the highest cost, associating approximately 9 minutes for a single BEM matrix. Simulating larger problems could lead to a more pronounced cost for solving the system than assembling due to the corresponding algorithmic complexities scaling with $\mathcal{O}(N^2)$ for assembling and $\mathcal{O}(N^3)$ for solving. Comparing the cost of the two methods employed for the construction of the projection bases $\mathbf{W}_{\text{AKR}}$ and $\mathbf{W}_{\text{RBM}}$, AKR necessitates only 20% of the time required in the alternative RBM. In effect, as the cost of solving the system is almost negligible to the one of the assembly phase, the advantage of the AKR is linked with the smaller number of systems that are assembled in the offline stage. Investigating larger systems, the computational advantage would also incorporate the gain of avoiding to fully solve the system at $\omega \in \Omega_{\text{AKR}}$.

5.2.3. Memory constrained AKR for the MOR of an acoustic BEM system

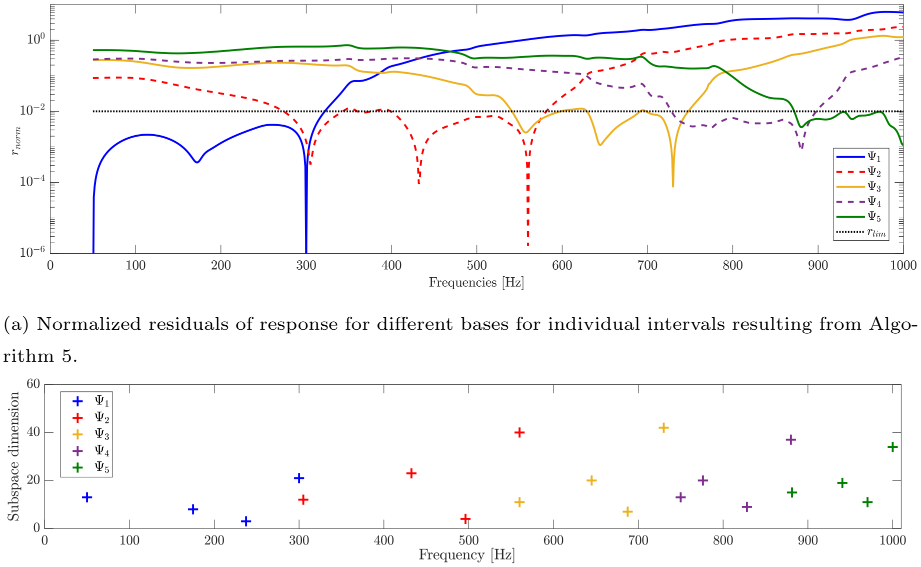
Constructing the reduction basis with the AKR leads to reduced order models that combined with a kernel polynomial expansion might inflict a memory excess. Although in this case, the corresponding reduced order model (ROM) is composed of 157 DOFs, which can be easily handled, for larger models this can prove critical and thus using the memory constrained AKR can be beneficial. In this section, the same internal cube (2606 DOFs) Helmholtz problem is reduced with the memory constrained AKR to approximate the solution of the same interval $\Psi_{\text{tot}} := \Psi$ employing only 80 DOFs. The number of DOFs in this case is selected as approximately 50% of the reduced DOFs resulted by the basic AKR approach, i.e. $\ell := 80$, to demonstrate the capability of the algorithm to impose a size constraint. However, in case of larger models the number of reduced DOFs can be effectively linked to the system memory.

Subinterval	Ψ_{ℓ_1}	Ψ_{ℓ_2}	Ψ_{ℓ_3}	Ψ_{ℓ_4}	Ψ_{ℓ_5}
Frequency [Hz]	$50 - 300$	$300 - 560$	$560 - 730$	$730 - 880$	$880 - 1000$
Basis size	45	79	80	79	79

Table 3: Frequency subintervals yielded by Algorithm 5

Deploying Algorithm 5 results in a separation of the initial parameter interval Ψ_{tot} into 5 smaller subintervals with $\Psi_{\text{tot}} = \Psi_{\ell_1} \cup \cdots \cup \Psi_{\ell_5}$ and to the corresponding projection bases $\mathbf{W}_1, \ldots, \mathbf{W}_5$. The frequency intervals $\Psi_{\ell_i}, i := 1, \ldots, 5$ and the dimension of the basis $\mathbf{W}_i$ corresponding to each Ψ_{ℓ_i} is given in Table 3. Furthermore, in Figure 6a and Figure 6b the normalized residuals yielded by the deployment of the reduction bases $\mathbf{W}_i$ to the whole Ψ_{tot} and the corresponding sampling within each Ψ_{ℓ_i} are demonstrated respectively.

As expected, the residuals originating from the individual approximations employing $\mathbf{W}_i$ satisfy the predefined residual tolerance of $r_{\text{tol}} := 10^{-2}$ within the respective Ψ_{ℓ_i}, while they fail to satisfy r_{tol} outside the interval. Judging from the sampling occurring within the subintervals, the subspace of the



(a) Normalized residuals of response for different bases for individual intervals resulting from Algorithm 5.

(b) Sampling within the windows of memory constrained AKR configuration.

Figure 6: Cube with vibrating cap

highest dimension is required at the upper end of the subinterval. This can be explained as, on the one hand, the procedure starts ensuring convergence from the higher part of the interval, while on the other hand, physically, it signifies an increased influence of the subspaces assembled at higher frequencies to the solution of the lower frequency systems as reported previously in [30]. Based on the residual curves as well as on the respective validity intervals, it follows that higher frequencies with more complex resonant behaviours, as expected, necessitate recycling of subspaces of higher dimension to achieve the predefined r_{tol}. This implies that setting a memory constraint results in narrower subintervals for higher frequencies and broader for the lower frequency bands.

6. Conclusion

In this work a novel method for automatically recycling Krylov subspaces across a predefined parameter interval for parametric linear systems is introduced. The method proposes selective sampling of Krylov subspaces on an automatically produced set of parameter values to construct a global projection basis, such that the approximate solution yielded by projection on the created basis satisfies a predefined residual tolerance for the interval considered. The proposed method is employed in the context of model order reduction of non-affine systems in combination with a matrix polynomial expansion to render the parametric dependency affine and alleviate the cost of parametric sweeps. It aims at minimizing the number of systems that need to be assembled and solved in the offline phase of the MOR strategy. Due to the polynomial approximation of the system, even the reduced model can be prone to memory excess motivating the development of a memory constrained model reduction technique. The technique proposed accepts a maximum size for the produced basis leading to a truncated parametric region of validity. Sequential deployment of the memory constrained algorithm leads to a windowing technique to cover the full parameter space required.

Since the AKR algorithms are based on a Krylov based iterative solution procedure, their applicability is limited to systems with rather clustered eigenvalues that facilitate fast convergence. System preconditioning that can potentially improve the spectral properties of the examined system is not considered. The techniques are tested on a randomly generated system that demonstrates the desired eigenvalue clustering properties and on a BEM system. The approximation and the efficiency of the method is compared to an alternative version of the reduced basis method that employs only a small number of system assemblies.

Comparing the wall-clock times and the accuracy of the methods, the AKR method significantly speeds up the solution of the system in a parameter sweep calculation and in comparison to the alternative RBM method employs fewer systems assemblies. The acceleration offered becomes more pronounced in case the AKR method is used in a MOR context as the cost of the system assembly is decreased. Finally, the proposed windowing technique is employed to split the parameter interval into different regions which also correspond to different reduction bases.

Acknowledgements

The authors gratefully acknowledge the anonymous reviewers for their constructive feedback that led to a significant improvement of the paper. The research of D. Panagiotopoulos is funded by an Early Stage Researcher grant within the European Project PBNv2 Marie Curie Initial Training Network (GA 721615) and the research of E. Deckers is funded by a grant from the Research Foundation – Flanders (FWO). Additionally, the Research Fund KU Leuven is gratefully acknowledged for its support.

References

[1] C. A. Brebbia, The boundary element method for engineers, Pentech press, 1980.

[2] V. Simoncini, D. B. Szyld, Recent computational developments in Krylov subspace methods for linear systems, Numerical Linear Algebra with Applications 14 (1) (2007) 1–59.

[3] G. H. Golub, C. F. Van Loan, Matrix computations, Vol. 3, JHU press, 2012.

[4] 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.

[5] 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.

[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] 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.

[10] 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. doi:10.1007/978-94-007-0089-5_6.
URL https://doi.org/10.1007/978-94-007-0089-5_6

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

[12] 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.

[13] T. F. Chan, M. K. Ng, Galerkin projection methods for solving multiple linear systems, SIAM Journal on Scientific Computing 21 (3) (1999) 836–850.

[14] 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.

[15] D. Ryckelynck, A priori hyperreduction method: an adaptive approach, Journal of computational physics 202 (1) (2005) 346–366.

[16] D. Ryckelynck, F. Chinesta, E. Cueto, A. Ammar, On the a priori model reduction: Overview and recent developments, Archives of Computational methods in Engineering 13 (1) (2006) 91–128.

[17] Y. Liang, H. Lee, S. Lim, W. Lin, K. Lee, C. Wu, Proper orthogonal decomposition and its applications—part i: Theory, Journal of Sound and vibration 252 (3) (2002) 527–544.

[18] P. Kerfriden, P. Gosselet, S. Adhikari, S. P.-A. Bordas, Bridging proper orthogonal decomposition methods and augmented Newton–Krylov algorithms: an adaptive model order reduction for highly nonlinear mechanical problems, Computer Methods in Applied Mechanics and Engineering 200 (5-8) (2011) 850–866.

[19] R. W. Freund, Krylov-subspace methods for reduced-order modeling in circuit simulation, Journal of Computational and Applied Mathematics 123 (1-2) (2000) 395–421.

[20] Y. Lin, L. Bao, Y. Wei, Model-order reduction of large-scale k th-order linear dynamical systems via ak th-order Arnoldi method, International Journal of Computer Mathematics 87 (2) (2010) 435–453.

[21] J. Wolf, Modal synthesis for combined structural-acoustic systems, AIAA Journal 14 (1) (1976) 33–38.

[22] Y. Liang, W. Lin, H. Lee, S. Lim, K. Lee, H. Sun, Proper orthogonal decomposition and its applications–part ii: Model reduction for mems dynamical analysis, Journal of Sound and Vibration 256 (3) (2002) 515–532.

[23] G. Rozza, D. B. P. Huynh, A. T. Patera, Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations, Archives of Computational Methods in Engineering 15 (3) (2007) 1.

[24] A. Buffa, Y. Maday, A. T. Patera, C. Prud'homme, G. Turinici, A priori convergence of the greedy algorithm for the parametrized reduced basis method, ESAIM: Mathematical modelling and numerical analysis 46 (3) (2012) 595–603.

[25] A. Quarteroni, G. Rozza, A. Manzoni, Certified reduced basis approximation for parametrized partial differential equations and applications, Journal of Mathematics in Industry 1 (1) (2011) 3.

[26] S. K. Baydoun, M. Voigt, C. Jelich, S. Marburg, A greedy reduced basis scheme for multifrequency solution of structural acoustic systems, International Journal for Numerical Methods in Engineering 121 (2) (2020) 187–200.

[27] O. Axelsson, Conjugate gradient type methods for unsymmetric and inconsistent systems of linear equations, Linear algebra and its applications 29 (1980) 1–16.

[28] Y. Saad, Krylov subspace methods for solving large unsymmetric linear systems, Mathematics of computation 37 (155) (1981) 105–126.

[29] V. Simoncini, Restarted full orthogonalization method for shifted linear systems, BIT Numerical Mathematics 43 (2) (2003) 459–466.

[30] D. Panagiotopoulos, E. Deckers, W. Desmet, Krylov subspaces recycling based model order reduction for acoustic BEM systems and an error estimator, Computer Methods in Applied Mechanics and Engineering 359 (2020) 112755.

[31] B. Carpentieri, I. S. Duff, L. Giraud, Experiments with sparse preconditioning of dense problems from electromagnetic applications, CERFACS, Toulouse, France, Tech. Rep. TR/PA/00/04 (2000) 9.

[32] S. Amini, C. Ke, Conjugate gradient method for second kind integral equations—applications to the exterior acoustic problem, Engineering Analysis with Boundary Elements 6 (2) (1989) 72–77.

[33] S. Kirkup, The Boundary Element Method in acoustics: A survey, Applied Sciences 9 (8) (2019) 1642.

[34] S. Amini, N. Maines, Preconditioned Krylov subspace methods for boundary element solution of the Helmholtz equation, International Journal for Numerical Methods in Engineering 41 (5) (1998) 875–898.

[35] S. Marburg, S. Schneider, Performance of iterative solvers for acoustic problems. part i. solvers and effect of diagonal preconditioning, Engineering Analysis with Boundary Elements 27 (7) (2003) 727–750.

[36] K. Chen, P. J. Harris, Efficient preconditioners for iterative solution of the boundary element equations for the three-dimensional Helmholtz equation, Applied Numerical Mathematics 36 (4) (2001) 475–489.

[37] W. E. Arnoldi, The principle of minimized iterations in the solution of the matrix eigenvalue problem, Quarterly of applied mathematics 9 (1) (1951) 17–29.

[38] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, United States Governm. Press Office Los Angeles, CA, 1950.

[39] A. C. Antoulas, Approximation of large-scale dynamical systems, Vol. 6, Siam, 2005.

[40] Å. Björck, Numerics of gram-schmidt orthogonalization, Linear Algebra and Its Applications 197 (1994) 297–316.

[41] Y. Saad, Analysis of augmented Krylov subspace methods, SIAM Journal on Matrix Analysis and Applications 18 (2) (1997) 435–449.

[42] M. D. Gunzburger, J. S. Peterson, J. N. Shadid, Reduced-order modeling of time-dependent pdes with multiple parameters in the boundary data, Computer methods in applied mechanics and engineering 196 (4-6) (2007) 1030–1047.

[43] T. Liang, J. Wang, J. Xiao, L. Wen, Coupled BE–FE based vibroacoustic modal analysis and frequency sweep using a generalized resolvent sampling

method, Computer Methods in Applied Mechanics and Engineering 345 (2019) 518–538.

[44] S. M. Kirkup, The boundary element method in acoustics, Integrated sound software, 2007.

[45] S. Marburg, Six boundary elements per wavelength: is that enough?, Journal of Computational Acoustics 10 (01) (2002) 25–51.

[46] V. C. Henriquez, P. M. Juhl, OpenBEM-an open source boundary element method software in acoustics, in: Internoise 2010, Vol. 7, Sociedade Portuguesa de Acustica (SPA) Lisbon, Portugal, 2010, pp. 5796–5805.