DELFT UNIVERSITY OF TECHNOLOGY

REPORT 18-03

Scalable Convergence Using Two-Level Deflation Preconditioning for the Helmholtz Equation

V. Dwarka and C. Vuik

ISSN 1389-6520

Reports of the Delft Institute of Applied Mathematics

Delft 2018

Abstract

Recent research efforts aimed at iteratively solving the Helmholtz equation has focused on incorporating deflation techniques for accelerating the convergence of Krylov subpsace methods. The requisite for these efforts lies in the fact that the widely used and well acknowledged Complex Shifted Laplacian Preconditioner (CSLP) shifts the eigenvalues of the preconditioned system towards the origin as the wave number increases. The two-level-deflation preconditioner combined with CSLP (DEF) showed encouraging results in moderating the rate at which the eigenvalues approach the origin. However, for large wave numbers the initial problem resurfaces and the near-zero eigenvalues reappear. Our findings reveal that the reappearance of these near-zero eigenvalues occurs if the near-singular eigenmodes of the fine-grid operator and the coarse-grid operator are not properly aligned. This misalignment is caused by accumulating approximation errors during the inter-grid transfer operations. We propose the use of higher-order approximation schemes to construct the deflation vectors. The results from Rigorous Fourier Analysis (RFA) and numerical experiments confirm that our newly proposed scheme outperforms any other deflation-based preconditioner for the Helmholtz problem. In particular, the spectrum of the adjusted preconditioned operator stays fixed near one. These results can be generalized to general shifted indefinite systems with random right-hand sides. For the first time, the convergence properties for very large wavenumbers ($k = 10^6$ in one-dimension and $k = 10^3$ in two-dimensions) have been studied, and the convergence is close to wave number independence. Wave number independence for three-dimensions has been obtained for wave numbers up to $k = 75$. The new scheme additionally shows very promising results for the more challenging Marmousi problem. Despite having a strongly varying wave number, we managed to obtain a small and constant number of iterations.

1 Introduction

From investigating the earth's layers in seismology to assessing the effect of electromagnetic scattering in the presence of human tissue through MRI, the Helmholtz equation finds its application through various applications. Many efforts have been rendered in order to obtain accurate and computationally feasible solutions. Two major problem arise in trying to solve the Helmholtz equation numerically. First of all, for large wave numbers the numerical solution suffers from the so called 'pollution error', which intrinsically is a phase difference between the exact and numerical solution. The second issue relates to the convergence behaviour of the underlying solver. For medium to large wave numbers, the linear system becomes indefinite due to the negative eigenvalues. In order to balance the accuracy for such large wave numbers the linear system becomes very large and thus preconditioned iterative solvers are preferred, especially when considering higher dimensional problems [5]. As the wave number increases the eigenvalues of the preconditioned matrix start to shift towards the origin. These near-zero eigenvalues have a detrimental effect on the convergence speed of Krylov-based iterative solvers. In order to mitigate these effects, many preconditioners for the Helmholtz problem have been proposed throughout the years. A specific class of preconditioners focuses on the operator in question and shows performance gains for medium sized wave numbers. In [2] the preconditioner matrix is equal to the discretisized Laplacian operator, and variations on this include a real and/or complex shift. A widely known preconditioner is *Complex Shifted Laplacian Preconditioner* (CSLP) ([7],[8]). Despite achieving a substantial speed-up, the small eigenvalues of the preconditioned system still rush to zero as the wave number increases, which is why ultimately a deflation strategy was proposed in [6]. Deflation, in essence, projects the unwanted eigenvalues to zero and has been studied widely ([18], [19],[1]). While being able to improve the convergence and performance significantly, the near-zero eigenvalues still reappear for large wave numbers. In this work we present an adapted deflation scheme in order to obtain an efficient and fast solver. By using a higher-order approximation scheme for the deflation vectors, we are able to reach close to wave-number independent convergence.

2 Problem Description

We start by focusing on a simple one-dimensional mathematical model using a constant wave number k:

$$\begin{aligned} -\frac{d^2u}{dx^2} - k^2\, u &= \delta(x - x'), \\ u(0) = 0, u(L) &= 0, \\ x \in \Omega &= [0, L] \subset \mathbb{R}, \\ k &\in \mathbb{N} \setminus \{0\}. \end{aligned} \tag{1}$$

We will refer to this model problem as MP 1. For the one-dimensional case, the second order difference scheme with stepsize $h = \frac{1}{n}$ leads to

$$\frac{-u_{l-1} + 2u_l - u_{l+1}}{h^2} - k^2 u_l = f_l, l = 1, 2, \ldots, n.$$

Using a lexicographic ordering, the linear system can be formulated exclusively on the internal grid points due to the homogeneous Dirichlet boundary conditions. We obtain the following system and eigenvalues

$$\begin{aligned} Au &= \frac{1}{h^2}\text{tridiag}[-1 \;\; 2 - k^2 \;\; -1]u = f, \\ \hat{\lambda}^l &= \frac{1}{h^2}\left(2 - 2\cos(l\pi h)\right) - k^2, \\ l &= 1, 2, \ldots n. \end{aligned} \tag{2}$$

In order to investigate the scalability of the linear solver in higher dimensions (section 5), we define MP 2 and MP 3 to be the 2-D and 3-D versions of the original model problem.

Therefore, on the standard two-dimensional square unit domain $\Omega = [0,1] \times [0,1]$ with constant wave number k we consider

$$\begin{aligned} -\Delta u(x,y) - k^2 u(x,y) = \delta(x - \frac{1}{2}, y - \frac{1}{2}),\ (x,y) \in \Omega \setminus \partial\Omega \subset \mathbb{R}^2, \\ u(x,y) = 0,\ (x,y) \in \partial\Omega, \end{aligned} \tag{3}$$

This will be refered to as MP 2. Similarly, on the standard three-dimensional cube unit domain $\Omega = [0,1] \times [0,1] \times [0,1]$ we have

$$\begin{aligned} -\Delta u(x,y,z) - k^2 u(x,y,z) = \delta(x - \frac{1}{2}, y - \frac{1}{2}, z - \frac{1}{2}),\ (x,y,z) \in \Omega \setminus \partial\Omega \subset \mathbb{R}^2, \\ u(x,y,z) = 0,\ (x,y,z) \in \partial\Omega, \end{aligned} \tag{4}$$

We will refer to this as MP 3. The discretization using second order finite differences goes accordingly for higher dimensions, with the resulting matrices being penta- and hepta-diagonal for 2D and 3D respectively.

The final test problem is a representation of an industrial problem and is widely referred to as the 2D Marmousi Problem. We will refer to this model problem throughout upcoming chapters as MP 4. Note that all models, except the Marmousi model, contain Dirichlet boundary conditions to simulate the worst spectral properties for convergence. The original Marmousi problem is defined on a rectangular domain $\Omega = [0, 9200] \times [0, 3000]$. There are 158 layers with velocities ranging from 1500 m/s to 5500 m/s. In [18] a slightly adapted version of the original Marmousi problem is considered. The original domain has been truncated to $\Omega = [0, 8192] \times [0, 2048]$ in order to allow for efficient geometric coarsening of the discrete velocity profiles given that the domain remains in powers of 2. The original velocity $c(x,y)$ is also adapted by considering $2587.5 \leq c \leq 3325$. We will use the adjusted domain in order to benchmark against the results from [18]. Consequently, on the adjusted domain Ω, we define

$$\begin{aligned} -\Delta u(x,y) - k(x,y)^2 u(x,y) = \delta(x - 4000, y), (x,y) \in \Omega \setminus \partial\Omega \subset \mathbb{R}^2, \\ \left(\frac{\partial}{\partial \mathbf{n}} - ik\right) u(x,y) = 0, (x,y) \in \partial\Omega, \end{aligned} \tag{5}$$

where n denotes the outward normal unit vector in the x- and y-direction respectively. Note that we now have a non-constant wave number $k(x,y) = \frac{2\pi freq}{c(x,y)}$, where in this particular case $c(x,y)$ ranges between 2587.5 and 3325. For this adjusted version of the Marmousi problem, numerical experiments have been conducted using the frequencies $1, 10, 20$ and 40 Hz, where the grid has been resolved in such a way that the maximum wave number k at $\mathit{freq} = 1$ has a grid resolution of $kh \leq 0.039$. For the remaining frequencies, a grid resolution of $kh \leq 0.39$ is utilized.

2.1 Deflated Krylov Methods

Note that equation eq. (2) reveals that the spectrum for MP 1 contains both positive and negative eigenvalues for

$$k > \frac{2\sin(\pi\frac{h}{2})}{h} \approx \pi.$$

This indefiniteness narrows the choice of potential Krylov-based solvers due to the Conjugate Gradient type methods being ineffective. Adding to this, Krylov subspace methods are adversely affected by close to zero eigenvalues. While the application of the CSLP preconditioner was successful in confining the eigenvalues between 0 and 1, the Krylov solver remains defenseless against the hampering convergence behavior caused by the small eigenvalues for large k. Deflation is a technique designed to ”deflate” these unwanted eigenvalue onto zero. By means of a projection, it is possible to alleviate the adverse effects on the Krylov solver by either explicitly modifying the operator of the linear system ([15]) or by adapting the eigenvectors corresponding to the troublesome eigenvalues ([13], [14]). For large systems, the latter option is computationally burdensome. As a consequence, most applications in the literature are based on approximations of invariant subspaces obtained from Jordan decompositions. Deflation for large scale problems relies on multiplying the linear system by a projection matrix P and applying the Krylov subspace method to the projected system PA, rendering the projection matrix P to act as a preconditioner at the same time as follows

$$\begin{aligned}&PA\widehat{u} = Pf\\ &A \in \mathbb{C}^{n\times n},\ P \in \mathbb{R}^{n\times n},\ \widehat{u} \in \mathbb{R}^{n}\\ &m = \dim(P) < n\end{aligned}$$

Consider $A \in \mathbb{R}^{n\times n}$. Then its Jordan decomposition is given by

$$A = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} J_1 & \emptyset \\ \emptyset & J_2 \end{bmatrix} \begin{bmatrix} U_1 & U_2 \end{bmatrix}^{-1}$$

where $J_1 \in \mathbb{R}^{m\times m}$ and $J_2 \in \mathbb{R}^{(n-m)\times(n-m)}$ with $m \leq n$ represent the square Jordan blocks. Letting $P_{\{U_1,U_2\}}$ denote the projection onto $U_1 \subseteq \mathbb{R}^{m\times m}$ along $U_2 \subseteq \mathbb{R}^{(n-m)\times(n-m)}$, the projected system can be decomposed as

$$P_{\{U_1,U_2\}}A = A = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \emptyset & \emptyset \\ \emptyset & J_2 \end{bmatrix} \begin{bmatrix} U_1 & U_2 \end{bmatrix}^{-1}$$

The resulting system PA will have a zero eigenvalue with algebraic multiplicity m. The spectrum contained in the Jordan block J_1 appears invisible to the Krylov solver, improving the conditions for convergence. Analytically, the invariant subspaces are based on (generalized) eigenvectors, creating the necessity for approximations to these subspaces in order to meet practical purposes. As a result, the remaining part of the spectrum will typically differ from $\sigma(J_2)$.

2.2 Deflation Based Preconditioning for GMRES

Consider a general real valued linear system. The projection matrix $\widehat{P}$ and its complementary projection P can be defined as

$$\begin{aligned} &\widehat{P} = AQ \text{ where } Q = ZE^{-1}Z^T \text{ and } E = Z^TAZ \\ &A \in \mathbb{R}^{n\times n},\ Z \in \mathbb{R}^{m\times n} \\ &P = I - AQ, \end{aligned} \tag{6}$$

where Z functions as the deflation matrix whose $m < n$ columns are considered the deflation vectors and I is the $n \times n$ identity matrix. Additionally, the coarse-grid coefficient matrix E is assumed to be invertible. Matrix P is also known as the projection preconditioner. In Algorithm 1 we present the Preconditioned Deflated GMRES algorithm.

Algorithm 1: Preconditioned Deflated GMRES for system $Au = b$

```
Choose $u_0$ and compute $r_0 = b_0$ and $v_1 = r_0/||r_0||$
for for $j = 1, 2, ...k$ or until convergence do
  $\tilde{v}_j := Pv_j$
  $w = M^{-1}A\tilde{v}_j$
  for $i := 1, 2, ..., j$ do
    $h_{i,j} := w^Tv_i$
    $w := w - h_{i,j}v_i$
  end for
  $h_{j+1,j} := ||w||$
  $v_{j+1} := w/h_{j+1,j}$
end for
Store $V_k = \left[\tilde{v}_1, ..., \tilde{v}_k\right]$; $H_k = \{h_{i,j}\}$, $1 \leq i \leq j+1$, $1 \leq j \leq m$
Compute $y_k = argmin_y||b_0 - H_ky||$ and $u_k = u_0 + V_ky_k$
The entries of upper $k+1, k$ Hessenberg Matrix $H_k$ are the scalars $h_{i,j}$
Update approximated solution $\mathbf{u_k} = \mathbf{Qb} + \mathbf{P^Tu_k}$
```

2.3 The Deflation Preconditioner (DEF)

Based on theory above, the DEF-preconditioner has been defined by taking the coarse correction operator I_h^{2h} from a multigrid setting as the deflation subspace Z in equation eq. (6). I_{2h}^h can be interpreted as interpolating from grid Ω_{2h} to grid Ω_h. As a result, the DEF-preconditioner is commonly referred to as a two-level method and we obtain

$$\begin{aligned} &\widehat{P} = A_hQ \text{ where } Q = Z{A_{2h}}^{-1}Z^T \text{ and } A_{2h} = Z^TA_hZ \\ &P = I_h - A_hQ \text{ where } Z = I_{2h}^h \end{aligned} \tag{7}$$

For spectral improvement, the DEF-preconditioner is applied to the CSLP preconditioned system. The spectra of both systems are equivalent, which leads to solving the following linear systems [18]

$$
\begin{aligned}
&M^{-1}A_h u = M^{-1} f\\
&M^{-1}PA_h u = M^{-1}Pf\\
&P^T M^{-1}A_h u = P^T M^{-1} f
\end{aligned}
\tag{8}
$$

In the literature a distinction is made with respect to the two-level deflation operator. On the one hand we have the DEF-preconditioner as defined above. On the other hand we have the ADEF-preconditioner, which is defined by taking $P_{ADEF} = P + \gamma Q$. The inclusion of the shift γ ensures that the coarse-grid solve with respect to A_{2h} can be approximated [18]. In this work we solely focus on the DEF-preconditioner, and thus we can take $\gamma = 0$.

2.3.1 Inscalability and Spectral Analysis

We now shift our focus to the study of the eigenvalues of the DEF-operator without inclusion of the CSLP-preconditioner. In [11] and [18] detailed analytical derivations and expressions for the spectrum of the DEF-operator are given which we will use here. The eigenvalues of the system PA are given by

$$
\lambda^l(PA) = \lambda^l(A)\left(1 - \frac{\lambda^l(A)\cos(l\pi\frac{h}{2})^4}{\lambda^l(A_{2h})}\right) + \lambda^{n+1-l}(A)\left(1 - \frac{\lambda^{n+1-l}(A)\sin(l\pi\frac{h}{2})^4}{\lambda^l(A_{2h})}\right), \tag{9}
$$
$$
l = 1, 2, \ldots, \frac{n}{2}.
$$

Inspection of eq. (9) leads to the observation that the eigenvalues of the deflation operator P are given by

$$
\lambda^l(P) = \left(1 - \frac{\lambda^l(A)\cos(l\pi\frac{h}{2})^4}{\lambda^l(A_{2h})}\right) + \left(1 - \frac{\lambda^{n+1-l}(A)\sin(l\pi\frac{h}{2})^4}{\lambda^l(A_{2h})}\right). \tag{10}
$$

By introducing the following coefficients, we can rewrite eq. (9) as

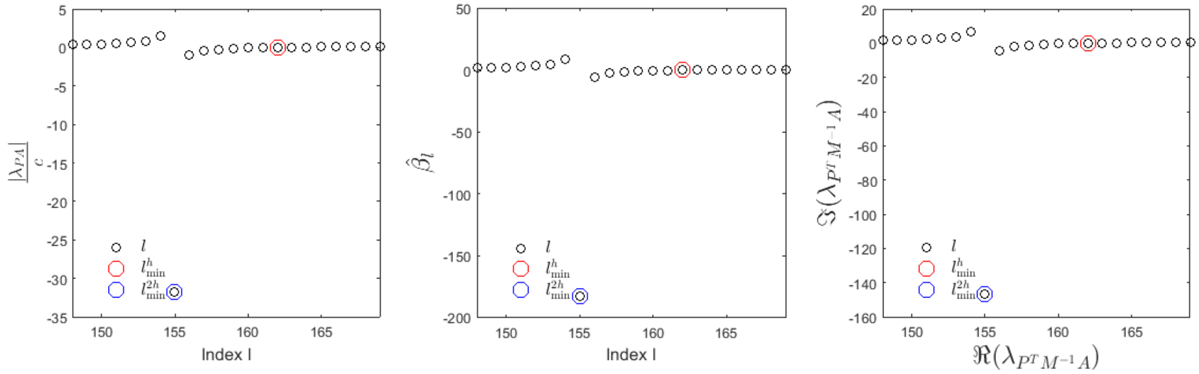
$$
\begin{aligned}
\alpha^l &= \left(1 - \frac{\lambda^l(A)\cos(l\pi\frac{h}{2})^4}{\lambda^l(A_{2h})}\right) = \frac{\lambda^{n+1-l}(A)\sin(l\pi\frac{h}{2})^4}{\lambda^l(A_{2h})},\\
\beta^l &= \left(1 - \frac{\lambda^{n+1-l}(A)\sin(l\pi\frac{h}{2})^4}{\lambda^l(A_{2h})}\right) = \frac{\lambda^l(A)\cos(l\pi\frac{h}{2})^4}{\lambda^l(A_{2h})},\\
\lambda^l(PA) &= \lambda^l(A)\alpha^l + \lambda^{n+1-l}(A)\beta^l, l = 1, 2, \ldots, \frac{n}{2}
\end{aligned}
\tag{11}
$$

Since the sine and cosine terms are always strictly less than 1, the eigenvalues of the system PA are essentially the product of eigenvalues of A multiplied by the scaled ratio of the eigenvalues of A and A_{2h}. In order to simplify the analysis, we therefore proceed by analyzing

$$\hat{\beta}^l = \left| \frac{\lambda^l(A)}{\lambda^l(A_{2h})} \right|, l = 1, 2, \ldots, \frac{n}{2}, \tag{12}$$

which provides an upperbound to the previously defined coefficients. It is easy to see that the eigenvalues of PA will approach the origin if the factor $\hat{\beta}^l$ becomes small for some l. If we define the constant c to be the magnitude of the largest eigenvalue of A, then we can scale the eigenvalues of PA by c and compare them to the eigenvalues $P^T M^{-1} A$ and $\hat{\beta}$ In fig. 1 we have plotted a selected range of eigenvalues of PA scaled by c and compared these to the eigenvalues of $P^T M^{-1} A$ (right) and $\hat{\beta}^l$ (center). On the x-axis we have the relevant indices l corresponding to the respective close to zero eigenvalues. The figure provides affirmative support for our remark that the behaviour of the eigenvalues of both PA and $P^T M^{-1} A$ are, apart from a scaling factor, determined by the behaviour of $\hat{\beta}^l$ as all three figures exhibit the same shape and pattern. $\hat{\beta}^l$ approaches the origin whenever $\left|\lambda^l(A)\right|$ becomes small, which is at $l = l^h_{\text{min}}$ (red marker). If $l^h_{\text{min}} \neq l^{2h}_{\text{min}}$ and $l^{2h}_{\text{min}} < l^h_{\text{min}}$, then we are dividing a relatively small number $\left|\lambda^{l^h_{\text{min}}}(A)\right|$ by a larger number $\left|\lambda^{l^h_{\text{min}}}(A_{2h})\right|$, which brings the resulting fraction closer to zero. The further apart l^h_{min} and l^{2h}_{min} are, the closer to zero the resulting term will be. The outlier appointed by the blue marker, is the result of exactly the opposite effect. At $l = l^{2h}_{\text{min}}$, $\left|\lambda^l(A_{2h})\right|$ will be at its smallest, while the magnitude of $\left|\lambda^l(A)\right|$ will still be large. In like manner, we get a large term, which explains the typical outliers we often encounter when the spectra of the operators PA and $P^T M^{-1} A$ are plotted. Note that the intermediate values , i.e. those between l^h_{min} and l^{2h}_{min} will be forced to become another outliers as well.

Figure 1: $kh = 0.625, k = 500$. *Left: eigenvalues of PA scaled by magnitude of the largest eigenvalue (c). Center: Ratio between eigenvalues of the fine-grid and coarse-grid operator ($\hat{\beta}$ from equation eq. (12)). Right: real part of eigenvalues $P^T M^{-1} A$.*



3 Eigenvector Perturbations

The next question which needs to be answered is what is causing the kernel of the coarse grid operator to shift. It has been reported that interpolating coarse-grid functions always introduces high-frequency modes, which can be interpreted as an aliasing phenomenon ([10], [9]). These high-frequency modes are the main cause for interpolation errors [10]. The effect becomes more severe as index l increases. If the high frequency eigenmodes are activated by interpolating from a coarse to a fine grid, then an interpolation error will start to dominate the approximation and the eigenvectors will not be approximated accurately. This affects the eigenvalues of A_{2h} as A_{2h} is obtained by first restricting the fine-grid elements onto the coarse-grid and then transferring the result back onto the fine-grid. To measure the extent of this effect, we make use of lemma 3.1 and corollary 3.1.1.

Lemma 3.1 (Intergrid Transfer I)**.** *Let B be the $\frac{n}{2} \times \frac{n}{2}$ matrix given by $B = Z^T Z$, where $Z = I_{2h}^h$ is the prolongation matrix and let $l_{\min}$ be the index of smallest eigenvalue of A in terms of magnitude. Then*

$$B\phi_{l_{\min},2h} = C_h \phi_{l_{\min},2h}, \tag{13}$$

$$\lim_{h \to 0} C_h = \lambda_{l_{\min}}(B) = 2. \tag{14}$$

where $\phi_{l,h}$ is the $l-th$ eigenvector on the fine-grid of A and $\lambda_l(B)$ is the $l-th$ eigenvalue of B.

Proof. We use the method from [9]. For $i = 1, 2, \dots n$ we have

$$\begin{aligned} Z^T \phi_{l_{\min},h} &= \frac{1}{2} \left(\sin((2i-1)h\pi l_{\min,h}) + 2\sin(2ih\pi l_{\min,h}) + \sin((2i+1)h\pi l_{\min,h}) \right), \\ &= \frac{1}{2} \left(2\sin(2ih\pi l_{\min,h}) + 2\cos(2ih\pi l_{\min,h}) \right) \sin(2ih\pi l_{\min,h}), \\ &= (1 + \cos(l_{\min,h}\pi h)) \sin(2ih\pi l_{\min,h}), \\ &= C_1(h)\phi_{l_{\min,2h}}. \end{aligned}$$

Now taking the limit as h goes to zero of the coefficient C_h gives $\lim_{h\to 0} C_1(h) = 2$. For $i = 1, 2, \dots, n$ we distinguish two cases; i is odd and i is even. We start with the first case

$$\begin{aligned} Z\phi_{l_{\min},2h} &= \frac{1}{2} \left(\sin(\frac{(i-1)h\pi l_{\min,h}}{2}) + \sin(\frac{(i+1)h\pi l_{\min,h}}{2}) \right), \\ &= \frac{1}{2} \left(\sin((i-1)h\pi l_{\min,h}) + \sin((i+1)h\pi l_{\min,h}) \right), \\ &= \cos(l_{\min,h}\phi h) \sin(ih\pi l_{\min,h}), \\ &= C_2(h)\phi_{l_{\min},h}. \end{aligned}$$

Again, taking the limit as h goes to zero of the coefficient $C_2(h)$ gives $\lim_{h\to 0} C_2(h) = 1$. For i is even, we obtain $Z\phi_{l_{\min},2h} = \sin(\frac{ih\phi l_{\min,h}}{2}) = \sin(ih\pi l_{\min,h}) = \phi_{l_{\min},h}$. We can combine both results to obtain $B\phi_{l_{\min},2h} = Z^T Z\phi_{l_{\min},2h} = Z^T(C_2(h)\phi_{l_{\min},h}) = C_1(h)C_2(h)\phi_{l_{\min},2h} = \hat{\lambda}_{l_{\min}}(B)\phi_{l_{\min},2h}$. where $\hat{\lambda}_{l_{\min}}(B)$ represents the perturbed eigenvalue of B at index l due to the approximation error. Taking the limit as h goes to zero provides

$$\begin{aligned}\lim_{h\to 0}\hat{\lambda}_{l_{\min}}(B) &= \lim_{h\to 0} C_1(h)C_2(h),\\ &= 2,\\ &= \lambda_{l_{\min,h}}(B).\end{aligned}$$

where we now have $\lambda_{l_{\min},h}(B) = 2$. □

Corollary 3.1.1 (Coarse-grid kernel)**.** *Let A_{2h} be the $\frac{n}{2}\times\frac{n}{2}$ matrix given by $A_{2h} = Z^T AZ$, where $Z = I_{2h}^h$ is the prolongation matrix and let $l_{\min}$ be the index of smallest eigenvalue of A in terms of magnitude. Then*

$$A_{2h}\phi_{l_{\min,2h}} = C_h\lambda_{l_{\min,h}}(A)\phi_{l_{\min,2h}}, , \tag{15}$$

$$\lim_{h\to 0} C_h = \lambda_{l_{\min},h}(B). \tag{16}$$

where $\phi_{j,2h}$ is the $l-th$ eigenvector on the coarse-grid of A_{2h} and $\lambda_j(A_{2h})$ is the $l-th$ eigenvalue of A_{2h}.

Proof. Using lemma 3.1 and its proof, we have

$$\begin{aligned}A_{2h}\phi_{l_{\min,2h}} &= \left(Z^T AZ\right)\phi_{l_{\min,2h}},\\ &= Z^T A\left(Z\phi_{l_{\min,2h}}\right),\\ &= Z^T A(C_2(h)\phi_{l_{\min},h}),\\ &= C_1(h)Z^T A\phi_{l_{\min},h},\\ &= C_1(h)Z^T\lambda_{l_{\min,h}}(A)\phi_{l_{\min},h},\\ &= \lambda_{l_{\min,h}}(A)C_1(h)\left(Z^T\phi_{l_{\min},h}\right),\\ &= \lambda_{l_{\min,h}}(A)C_1(h)C_2(h)\phi_{l_{\min,2h}}.\end{aligned}$$

Using lemma 3.1 it is easy to see that after taking the limit the eigenvalues of A_{2h} can be written as a product of the eigenvalues of A and the eigenvalues of B. □

From lemma 3.1 and corollary 3.1.1 it is clear that for $l_{\min}$, which is within the smooth-frequency range, the near-kernel coarse-grid eigenvalues $\lambda_{l_{\min},h}(A_{2h})$ are equal the product of $\lambda_{l_{\min},h}(A)$ and $\lambda_{l_{\min,h}}(B)$ when h goes to zero. Consequently, in the limiting case the coarse-grid kernel and the fine-grid kernel will be aligned proportionally and both A and A_{2h} will reach its smallest absolute eigenvalues at the same index $l_{\min}$.
Recall the behavior of the eigenvalues of PA can be represented by

$$\hat{\beta}^l = \left|\frac{\lambda^l(A)}{\lambda^l(A_{2h})}\right| \text{ for } l = 1, 2, \ldots, \frac{n}{2},$$

where we found that this ratio becomes very small by a mismatch of the smallest absolute eigenvalue of A and A_{2h} respectively. The index where this phenomena occurs is at $l = l_{\min,h}$, the index of the smallest absolute eigenvalue of A. We now proceed by showing that errors accumulated during interpolating and restricting the eigenvectors of A will lead to perturbations in the scaling factor $\lambda_{l_{\min},h}(B)$. As in the limit, we can write $\lambda_{l_{\min},h}(A_{2h}) = \lambda_{l_{\min},h}(B)\lambda_{l_{\min},h}(A_h)$, perturbations up to $\lambda_{l_{\min},h}(B)$ will propagate throughout the low-frequency part of the spectrum for $l \in \{1, 2, \dots, l_{\min,h}\}$, eventually resulting in the errors related to $\lambda^l(A_{2h})$ for $l = l_{\min,h}$.
To measure to what extent these perturbations to $\lambda(B)$ lead to projection errors in constructing the projection operator P, we make use of the following proposition which was mentioned in [3].

Theorem 3.2 (Projection Error I). *Let X be the deflation space spanned by column vectors of Z and let the eigenvector corresponding to the smallest eigenvalue of A be denoted by $\phi_{l_{\min,h}} \notin X$. Let $P = ZB^{-1}Z^T$ with $B = Z^TZ$ be the orthogonal projector onto X. Then the projection error E is given by*

$$E = \|(I-P)\phi_{l_{\min,h}}\|^2 = {\phi_{l_{\min,h}}}^T \phi_{l_{\min,h}} - {\phi_{l_{\min,h}}}^T ZB^{-1}Z^T\phi_{l_{\min,h}}.$$

Proof. By idempotency of the orthogonal projector, we have

$$\begin{aligned}
\|(I-P)\phi_{l_{\min,h}}\|^2 &= {\phi_{l_{\min,h}}}^T(I-P)(I-P)\phi_{l_{\min,h}}, \\
&= {\phi_{l_{\min,h}}}^T(I-P)\phi_{l_{\min,h}}, \\
&= {\phi_{l_{\min,h}}}^T\phi_{l_{\min,h}} - {\phi_{l_{\min,h}}}^T ZB^{-1}Z^T\phi_{l_{\min,h}}.
\end{aligned}$$

□

From this representation of the projection error it is difficult to see how this influences the behavior of the eigenvalues of A_{2h}. We therefore proceed by rewriting the projection error in terms of a perturbation to the eigenvalues of the operator B.

Corollary 3.2.1 (Projection Error II). *Let X be the deflation space spanned by the column vectors of Z and let the eigenvector corresponding to the smallest eigenvalue of A be denoted by $\phi_{l_{\min,h}} \notin X$. Let $P = ZB^{-1}Z^T$ with $B = Z^TZ$ be the orthogonal projector onto X. Then the projection error E is given by*

$$E = \|(I-P)\phi_{l_{\min,h}}\|^2 = \left(1 - \frac{\lambda_{l_{\min},h}(B) - \delta_1}{\lambda_{l_{\min},h}(B) - \delta_2}\right){\phi_{l_{\min,h}}}^T\phi_{l_{\min,h}},$$

where $\delta_1 = \lambda_{l_{\min},h}(B) - \frac{{\phi_{l_{\min,h}}}^T\hat{B}\phi_{l_{\min,h}}}{{\phi_{l_{\min,h}}}^T\phi_{l_{\min,h}}}$ *and* $\delta_2 = \lambda_{l_{\min},h}(B) - \frac{{\phi_{l_{\min,h}}}^T\hat{B}\phi_{l_{\min,h}}}{{\phi_{l_{\min,h}}}^T Z\left(B^{-1}Z^T\phi_{l_{\min,h}}\right)}$.

Proof. We first start by showing that in the limit, the error goes to zero and there are no perturbations to the eigenvalues of B. Using lemma 3.1 and its proof we know that in the limit $Z^T\phi_{l_{\min},h}$ is an eigenvector of B. We would thus have

$$\begin{aligned}
\|(I-P)\phi_{l_{\min},h}\|^2 &= {\phi_{l_{\min},h}}^T\phi_{l_{\min},h} - {\phi_{l_{\min},h}}^T Z\left(B^{-1}Z^T\phi_{l_{\min},h}\right),\\
&= {\phi_{l_{\min},h}}^T\phi_{l_{\min},h} - {\phi_{l_{\min},h}}^T Z\left(\frac{Z^T\phi_{l_{\min},h}}{\lambda_{l_{\min},h}(B)}\right),\\
&= {\phi_{l_{\min},h}}^T\phi_{l_{\min},h} - \frac{{\phi_{l_{\min},h}}^T ZZ^T\phi_{l_{\min},h}}{\lambda_{l_{\min},h}(B)},\\
&= {\phi_{l_{\min},h}}^T\phi_{l_{\min},h} - \frac{{\phi_{l_{\min},h}}^T\left(\hat{B}\phi_{l_{\min},h}\right)}{\lambda_{l_{\min},h}(B)}.
\end{aligned}$$

Note that $\hat{B}$ has dimension $n \times n$ and has $\frac{n}{2}$ eigenvalues equal to the eigenvalues of B and $\frac{n}{2}$ zero eigenvalues. By lemma 3.1 and its proof, we also have that $\phi_{l_{\min},h}$ is an eigenvector of $\hat{B}$, which leads to

$$\begin{aligned}
\|(I-P)\phi_{l_{\min},h}\|^2 &= {\phi_{l_{\min},h}}^T\phi_{l_{\min},h} - \frac{{\phi_{l_{\min},h}}^T\left(\hat{B}\phi_{l_{\min},h}\right)}{\lambda_{l_{\min},h}(B)}, \qquad (17)\\
&= {\phi_{l_{\min},h}}^T\phi_{l_{\min},h} - \frac{{\phi_{l_{\min},h}}^T\left(\lambda_{l_{\min},h}(\hat{B})\phi_{l_{\min},h}\right)}{\lambda_{l_{\min},h}(B)},\\
&= 0.
\end{aligned}$$

Now, in the non-limiting case, we have two sources of errors; the factor containing $\lambda_{l_{\min},h}(B)$ both in the numerator and denominator will be subjected to perturbations. Starting with the denominator, if we let $\tilde{\lambda}_{l_{\min},h}(B)$ denote the perturbed eigenvalue of B, we can have

$${\phi_{l_{\min},h}}^T Z\left(B^{-1}Z^T\phi_{l_{\min},h}\right) = {\phi_{l_{\min},h}}^T Z\left(\frac{Z^T\phi_{l_{\min},h}}{\tilde{\lambda}_{l_{\min},h}(B)}\right) \neq {\phi_{l_{\min},h}}^T Z\left(\frac{Z^T\phi_{l_{\min},h}}{\lambda_{l_{\min},h}(B)}\right).$$

Reordering leads to

$$\begin{aligned}
\tilde{\lambda}_{l_{\min},h}(B) &= \frac{{\phi_{l_{\min},h}}^T ZZ^T\phi_{l_{\min},h}}{{\phi_{l_{\min},h}}^T Z\left(B^{-1}Z^T\phi_{l_{\min},h}\right)},\\
&= \frac{{\phi_{l_{\min},h}}^T\hat{B}\phi_{l_{\min},h}}{{\phi_{l_{\min},h}}^T Z\left(B^{-1}Z^T\phi_{l_{\min},h}\right)}.
\end{aligned}$$

Using $\tilde{\lambda}_{l_{\min},h}(B)$, we can now write

$$\tilde{\lambda}_{l_{\min},h}(B){\phi_{l_{\min},h}}^T Z\left(B^{-1}Z^T\phi_{l_{\min},h}\right) = {\phi_{l_{\min},h}}^T\hat{B}\phi_{l_{\min},h},$$

and the perturbation to $\lambda_{l_{\min},h}(B)$ is

$$\begin{aligned}
\delta_2 &= \lambda_{l_{\min},h}(B) - \tilde{\lambda}_{l_{\min},h}(B),\\
&= \lambda_{l_{\min},h}(B) - \frac{{\phi_{l_{\min},h}}^T\hat{B}\phi_{l_{\min},h}}{{\phi_{l_{\min},h}}^T Z\left(B^{-1}Z^T\phi_{l_{\min},h}\right)}.
\end{aligned}$$

The second source of error is due to $\hat{B}\phi_{l_{\min,h}} \neq \lambda_{l_{\min,h}}(B)\phi_{l_{\min,h}}$. If we let η denote the error vector, i.e. $\eta = \hat{B}\phi_{l_{\min,h}} - \lambda_{l_{\min,h}}(B)\phi_{l_{\min,h}}$, then $\hat{B}\phi_{l_{\min,h}} = \lambda_{l_{\min,h}}(B)\phi_{l_{\min,h}} + \eta$ and substitution gives

$$\begin{aligned}\tilde{\lambda}_{l_{\min,h}}(B)\phi_{l_{\min,h}}{}^T Z\left(B^{-1}Z^T\phi_{l_{\min,h}}\right) &= \phi_{l_{\min,h}}{}^T \hat{B}\phi_{l_{\min,h}}, \\ &= \phi_{l_{\min,h}}{}^T\left(\lambda_{l_{\min,h}}(B)\phi_{l_{\min,h}} + \eta\right).\end{aligned}$$

Letting $\delta_1 = -\frac{\phi_{l_{\min,h}}{}^T\eta}{\phi_{l_{\min,h}}{}^T\phi_{l_{\min,h}}}$, we obtain

$$\begin{aligned}\tilde{\lambda}_{l_{\min,h}}(B)\phi_{l_{\min,h}}{}^T Z\left(B^{-1}Z^T\phi_{l_{\min,h}}\right) &= \phi_{l_{\min,h}}{}^T\left(\lambda_{l_{\min,h}}(B)\phi_{l_{\min,h}} + \eta\right), \\ &= \left(\lambda_{l_{\min,h}}(B) - \delta_1\right)\phi_{l_{\min,h}}{}^T\phi_{l_{\min,h}}.\end{aligned}$$

Finally, we can now rewrite the projection error E in terms of perturbations to the eigenvalues of B;

$$\begin{aligned}\|(I-P)\phi_{l_{\min,h}}\|^2 &= \phi_{l_{\min,h}}{}^T\phi_{l_{\min,h}} - \phi_{l_{\min,h}}{}^T Z\left(B^{-1}Z^T\phi_{l_{\min,h}}\right), \\ &= \left(1 - \frac{\lambda_{l_{\min,h}}(B) - \delta_1}{\lambda_{l_{\min,h}}(B) - \delta_2}\right)\phi_{l_{\min,h}}{}^T\phi_{l_{\min,h}},\end{aligned}$$

which gives the statement. □

corollary 3.2.1 reveals that the projection error due to the inaccurate approximations of the eigenvectors can be represented by deviations from $\lambda_{l_{\min,h}}(B)$. In table 1 we present the projection error for various k. The results illustrate that the projection error increases linearly with k. Along with the projection error, the misalignment between $l_{\min,h}$ and $l_{\min,2h}$ increases. As a result, the kernel of A and A_{2h} are separated causing the eigenvalues of the preconditioned system to move towards the origin. If we let $kh = 0.3125$, the projection error is reduced. However, already for $k = 1000$, the error regains magnitude, which explains why, despite resorting to a finer grid, the near-zero eigenvalues eventually reappear for higher wave numbers when the DEF-preconditioner is used.

Table 1: Projection Error for $\phi_{l_{\min,h}}$ for various values of k. $j_{\min,h}$ and $l_{\min,2h}$ denote the index for the smallest absolute eigenvalue of A and A_{2h} respectively.

k	E	$l_{\min,h}$	$l_{\min,2h}$	E	$l_{\min,h}$	$l_{\min,2h}$
	$kh = 0.625$			$kh = 0.3125$		
10	0.0672	3	3	0.0077	3	3
50	0.4409	16	15	0.0503	16	16
100	0.8818	32	31	0.0503	32	32
500	4.670	162	155	0.5031	162	158
1000	9.2941	324	310	1.0062	324	316

In Section section 2.3.1 we have shown that the spectrum of PA and $PM^{-1}A$ is (apart from a scaling factor) equivalent to

$$\hat{\beta}^l = \left| \frac{\lambda^l(A)}{\lambda^l(A_{2h})} \right|,$$
$$l = 1, 2, \ldots, \frac{n}{2}.$$

From lemma 3.1 and corollary 3.1.1 we additionally found that in the limit near $l = l_{\min,h}$ we can express the eigenvalues of the coarse-grid operator A_{2h} in terms of $\lambda_{l_{\min},h}(B)$

$$\lambda^{l_{\min,h}}(A_{2h}) = \lambda^{l_{\min,h}}(A)\lambda_{l_{\min},h}(B). \tag{18}$$

Thus in the vicinity of the kernel, we can write

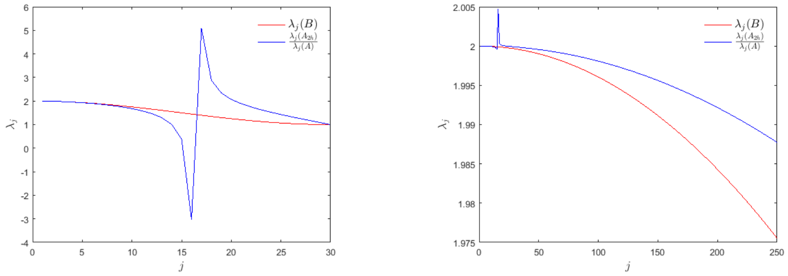
$$\hat{\beta}^l = \left| \frac{\lambda^l(A)}{\lambda^l(A_{2h})} \right| = \frac{1}{\lambda^l(B)}. \tag{19}$$

corollary 3.2.1 reflects that errors in projecting the eigenvectors onto the coarse-grid lead to errors in the eigenvalues of the operator B. These errors accumulate and increase as index l increases, due to the eigenvectors becoming more oscillatory. If we account for these errors, then (eq. (19)) becomes

$$\hat{\beta}^l = \left| \frac{\lambda^l(A)}{\lambda^l(A_{2h})} \right| = \frac{1}{\hat{\lambda}^l(B)}.$$

for some perturbed $\hat{\lambda}^l(B)$. These perturbations to the eigenvalues of B cause inaccurate scaling of the eigenvalues of A, eventually leading to the kernel of A_{2h} being located at a different index $l_{\min,2h} \neq l_{\min,h}$.
In fig. 3(a) and fig. 3(b) we have plotted the eigenvalues of B and the ratio between the eigenvalues of A_{2h} and A according to equation eq. (19). Note that the latter essentially represents the perturbed $\lambda^l(B)$ due to errors accumulated during prolongating and restricting the eigenvectors of A. It can be noted that as h becomes smaller, the ratio slowly converges to $\lambda^l(B)$. This observation is also in line with the projection error decreasing. In many literature surveys and works, it has been stated that the deflation vectors should be close approximations of the eigenvectors. However, a more accurate statement would be that the prolongated coarse-grid eigenvectors should be sufficiently accurate approximations of the eigenvectors. This can either be achieved by setting the grid resolution kh very small or incorporating an approximation scheme, other than the current interpolation scheme to obtain more accurate approximations of the eigenvectors. In order to combat this effect without resorting to solving large linear systems, we propose the use of higher-order deflation vectors. In the next section, we will show that the use of these higher-order deflation vectors significantly decreases the projection error, leading to the kernels of A and A_{2h} remaining proportional to each other.

Figure 2: $k = 50$. *Plot of the ratio between the fine-grid and coarse-grid eigenvalues (equation (eq. (19))) and the eigenvalues of B. $l_{\min,h} = 16$ and $l_{\min,2h} = 15$ for $kh = 0.825$. For $kh = 0.01$, $l_{\min,h} = l_{\min,2h} = 16$.*



(a) $kh = 0.825$,

(b) $kh = 0.01$

4 Higher-order Deflation

4.1 Quadratic Approximation

We have seen that the current geometric multigrid vectors are not sufficiently warranting against the near zero eigenvalues reappearing. So far, the main objective for using geometric multigrid vectors is that they are easily implemented in a multi-level setting. Moreover, the vectors are sparse and easy to compute. The problem, however, seems to be mapping of the fine-grid near kernel to the coarse-grid near kernel. If the latter two are misaligned, the eigenvalues of the deflation preconditioned operator will approach the origin. Consequently, while the deflation preconditioner has been designed to alleviate the near zero eigenvalues of the CSLP preconditioned system, the coarse-grid operator itself seems to be another source for near zero eigenvalues.

So far, we have used the bilinear interpolation and prolongation operator to construct the matrices I_h^{2h} and I_{2h}^h. An effective alternative should keep the simplicity of the geometric multigrid vectors, while attaining higher accuracy at mapping the kernel onto the coarse grid. Recall that the grid transfer functions $u_2h = [u_{2h_1}, \dots, u_{2h_n}]$ from Ω_{2h} to the fine grid Ω_h using standard linear interpolation are given by

$$I_{2h}^h : \Omega_{2h} \to \Omega_h, \quad u_{2h} \to I_{2h}^h u_{2h} \tag{20}$$

such that

$$\begin{cases} [u_{2h}]_{i/2} & \text{if } i \text{ is even,} \\ \frac{1}{2}\left([u_{2h}]_{(i-1)/2} + [u_{2h}]_{(i-1)/2}\right) & \text{if } i \text{ is odd,} \end{cases} \quad i = 1, \dots, n-1 \tag{21}$$

A closer look reveals that the current transfer functions are only reinforced at the odd components, leaving the even components unchanged. In fact, these components are mapped to linear combination of their fine-grid counterparts ϕ_{h_l} and a complimentary mode $\phi_{h_{n+1-l}}$ with first order accuracy [10]. A more general representation of the linear interpolation operator for the even components can be given by using rational *Bézier* curves. The use of these curves within the context of multigrid methods has been studied in [4] and [12]. Using these vectors as vectors for the input of the prolongation and restriction matrices in a multigrid setting is referred to as a *monotone multigrid method*. The monotonicity comes from the construction of the coarse-grid approximations, which ensures that the coarse-grid functions approximate the fine-grid functions monotonically [12], [16]. We will use the theory from [12] and [16] below to introduce these concepts.

Definition 4.1 (*Bézier* curve)**.** A *Bézier* curve of degree n is a parametric curve defined by

$$\begin{aligned} B(t) &= \sum_{j=0}^{n} b_{j,n}(t) P_j, \;\; 0 \leq t \leq 1, \;\; \text{where the polynomials} \\ b_{j,n}(t) &= (n, j)\, t^j (1-t)^{n-j}, \;\; j = 0, 1, \ldots, n, \end{aligned}$$

are known as the Bernstein basis polynomials of order n. The points P_j are called control points for the *Bézier* curve.

Definition 4.2 (Rational *Bézier* curve)**.** A rational *Bézier* curve of degree n with control points $P_0, P_1, \ldots, P_n$ and scalar weights $w_0, w_1, \ldots, w_n$ is defined as

$$C(t) = \frac{\sum\limits_{j=0}^{n} w_j b_{j,n}(t) P_j}{\sum\limits_{j=0}^{n} w_j b_{j,n}(t)}.$$

Definition 4.3 (Linear Interpolation)**.** Let $[u_{2h}]_{(j-1)/2}$ and $[u_{2h}]_{(j+1)/2}$, be the end points within a component span defined on the coarse grid. Then the prolongation scheme for the even nodes can be characterized by a Rational *Bézier* curve of degree 1 with polynomials

$$\begin{aligned} b_{0,1}(t) &= 1 - t, \\ b_{1,1}(t) &= t, \end{aligned}$$

whenever j is odd by taking the weights $w_0 = w_1 = 1$ and $t = \frac{1}{2}$. Note that in case $w_0 = w_1$ and non-rational we obtain the original *Bézier* curve.

$$\begin{aligned} C(\frac{1}{2}) &= \frac{\frac{1}{2}[u_{2h}]_{(j-1)/2} + (1 - \frac{1}{2})[u_{2h}]_{(j+1)/2}}{\frac{1}{2} + (1 - \frac{1}{2})}, && (22) \\ &= \frac{1}{2}\left([u_{2h}]_{(j-1)/2} + [u_{2h}]_{(j+1)/2}\right). && (23) \end{aligned}$$

When j is even, we take the middle component $[u_{2h}]_{j/2}$, which itself gets mapped onto the fine grid.

For large k, the prolongation operator working on the even components is not sufficiently accurate to map the near kernels to adjacent modes on Ω_{2h} and Ω_h. Consequently, we wish to find a higher order approximation scheme, which takes the even components into account. We thus consider a quadratic rational *Bézier* curve in order to find appropriate coefficients to yield a higher order approximation of the fine-grid functions by the coarse grid functions.

Definition 4.4 (Quadratic Approximation)**.** Let $[u_{2h}]_{(j-2)/2}$ and $[u_{2h}]_{(j+2)/2}$, be the end points within a component span defined on the coarse grid. Then the prolongation operator can be characterized by a Rational *Bézier* curve of degree 2 with polynomials

$$\begin{aligned} b_{0,2}(t) &= (1-t)^2, \\ b_{1,2}(t) &= 2t(1-t), \\ b_{2,2}(t) &= t^2, \end{aligned}$$

and control point $[u_{2h}]_{j/2}$, whenever j is even. Because we wish to add more weight to the center value, we take weights $w_0 = w_2 = \frac{1}{2}$, $w_1 = \frac{3}{2}$ and $t = \frac{1}{2}$ to obtain

$$\begin{aligned} C(t) &= \frac{\frac{1}{2}(1-t)^2[u_{2h}]_{j-1} + \frac{3}{2}2t(1-t)[u_{2h}]_j + \frac{1}{2}(t)^2[u_{2h}]_{j+1}}{\frac{1}{2}(1-t)^2 + \frac{3}{2}2t(1-t) + \frac{1}{2}(t)^2} \\ &= \frac{\frac{1}{2}(1-\frac{1}{2})^2[u_{2h}]_{j-1} + \frac{3}{2}(2)(\frac{1}{2})(1-\frac{1}{2})[u_{2h}]_j + \frac{1}{2}(\frac{1}{2})^2[u_{2h}]_{j+1}}{\frac{1}{2}(1-\frac{1}{2})^2 + \frac{1}{2}(2)(\frac{1}{2})(1-\frac{1}{2}) + \frac{1}{2}(\frac{1}{2})^2} \\ &= \frac{\frac{1}{8}[u_{2h}]_{j-1} + \frac{3}{4}[u_{2h}]_j + \frac{1}{8}[u_{2h}]_{j+1}}{1} \\ &= \frac{1}{8}\left([u_{2h}]_{j-1} + 6[u_{2h}]_j + [u_{2h}]_{j+1}\right). \end{aligned} \tag{24}$$

When j is odd, $[u_{2h}]_{(j-1)/2}$ and $[u_{2h}]_{(j+1)/2}$ have an even component and we are in the same scenario as is the case with linear interpolation.

As mentioned earlier, we use the Galerkin approach to construct the restriction operator. We finally obtain the following two higher order grid transfer operators

$$I_{2h}^{h} = I_{h}^{2h^T}.$$

Using the new matrices I_{2h}^h and I_h^{2h}, we can now construct similar analytical expressions for the eigenvalues of A_{2h}, PA and $P^TM^{-1}A$, where we following the same approach as [10], [9] and [11]. We therefore first consider the mapping of the eigenvectors on the same basis by these new operators.

Based on the upper scheme, we can now redefine the prolongation and restriction operator as follows

$$I_{2h}^h : \Omega_{2h} \to \Omega_h, \quad u_{2h} \to I_{2h}^h u_{2h} \tag{25}$$

such that

$$I_{2h}^{h}\left[u_{2h}\right]_{i}=\left\{\begin{array}{cc}\frac{1}{8}\left(\left[u_{2h}\right]_{(i-2)/2}+6\left[u_{2h}\right]_{(i)/2}+\left[u_{2h}\right]_{(i+2)/2}\right) & \text{if } i \text{ is even,}\\ \frac{1}{2}\left(\left[u_{2h}\right]_{(i-1)/2}+\left[u_{2h}\right]_{(i+1)/2}\right) & \text{if } i \text{ is odd,}\end{array}\right\}, \tag{26}$$

for $i=1,\dots,n-1$ and

$$I_{h}^{2h}\,:\,\Omega_h\to\Omega_{2h},\quad u_h\to I_h^{2h}\,u_h \tag{27}$$

such that

$$I_{h}^{2h}\left[u_{h}\right]_{i}=\frac{1}{8}\left(\left[u_{h}\right]_{(2i-2)}+4\left[u_{h}\right]_{(2i+1)}+6\left[u_{h}\right]_{(2i)}+4\left[u_{h}\right]_{(2i+1)}+\left[u_{h}\right]_{(2i+2)}\right),$$

for $i=1,\dots,\frac{n}{2}$.

We now analyze the mapping properties of these operators with respect to the eigenvectors. We will use the same method from [9] and start with the prolongation operator over the first part of the index set $j=1,2,\dots\frac{n}{2}$. The prolongation operator maps the coarse-grid eigenvectors for indices $j,l=1,2,\dots\frac{n}{2}$ to

$$\begin{aligned}[I_h^{2h}\phi_{2h}]_j^l &= \frac{1}{8}\left[\sin((j-2)/2)l\pi 2h)+6\sin((j)/2)l\pi 2h)+\sin((j+2)/2)l\pi 2h)\right],\\ &= \frac{1}{8}\left[\sin((j-2)l\pi h)+6\sin((j)l\pi h)+\sin((j+2)l\pi h)\right],\\ &= \frac{1}{8}\left[2\cos(2l\pi h)+6\right]\sin(lj\pi h),\\ &= \left[\frac{1}{4}\cos(2l\pi h)+\frac{3}{4}\right]\sin(lj\pi h),\end{aligned}$$

for j is even and

$$\begin{aligned}[I_h^{2h}\phi_{2h}]_j^l &= \frac{1}{8}\left[4\sin((j-1)/2)l\pi 2h)+4\sin((j+1)/2)l\pi 2h)\right],\\ &= \frac{1}{2}\left[\sin((j-1)l\pi h)+\sin((j+1)l\pi h)\right],\\ &= \frac{1}{2}\left[2\cos(l\pi h)\right]\sin(lj\pi h)\right],\\ &= \left[\cos(l\pi h)\right]\sin(lj\pi h),\end{aligned}$$

for j is odd. With respect to the remaining part of the index set containing j, we use that

$$\begin{aligned}\phi_h^{n+1-l_j} &= -(-1)^j\sin(lj\pi h),\\ j &= 1,2,\dots n-1,\\ l &= 1,2,\dots\frac{n}{2}.\end{aligned} \tag{28}$$

Note that eq. (28) is only positive when j is odd. Consequently for even j such that $j \in \left\{\frac{n}{2}, \dots, n-1\right\}$ is even, we obtain

$$\begin{aligned}
[I_h^{2h}\phi_{2h}]_j^l &= \frac{1}{8}\left[-\sin((j-2)/2)l\pi 2h) - 6\sin((j)/2)l\pi 2h) - \sin((j+2)/2)l\pi 2h)\right], \\
&= \frac{1}{8}\left[-\sin((j-2)l\pi h) - 6\sin((j)l\pi h) - \sin((j+2)l\pi h)\right], \\
&= \frac{1}{8}\left[-2\cos(2l\pi h) - 6\right]\sin(lj\pi h), \\
&= \left[-\frac{1}{4}\cos(2l\pi h) - \frac{3}{4}\right]\sin(lj\pi h),
\end{aligned}$$

whereas for j is odd, we now have

$$\begin{aligned}
[I_h^{2h}\phi_{2h}]_j^l &= \frac{1}{8}\left[4\sin((j-1)/2)l\pi 2h) + 4\sin((j+1)/2)l\pi 2h)\right], \\
&= \frac{1}{2}\left[\sin((j-1)l\pi h) + \sin((j+1)l\pi h)\right], \\
&= \frac{1}{2}\left[2\cos(l\pi h)\right]\sin(lj\pi h)\right], \\
&= \left[\cos(l\pi h)\right]\sin(lj\pi h).
\end{aligned}$$

With respect to our basis, we therefore obtain the following 2×1 block for the prolongation operator

$$[I_{2h}^h]^l = \begin{bmatrix} \cos(l\pi h) + \frac{1}{4}\cos(2l\pi h) + \frac{3}{4} \\ \cos(l\pi h) - \frac{1}{4}\cos(2l\pi h) - \frac{3}{4} \end{bmatrix}.$$

Similarly, the restriction operator works on the first part of the the basis by mapping the fine-grid eigenvectors according to

$$\begin{aligned}
\left[I_{2h}^h\phi_h\right]_j^l &= [I_{2h}^h]^l\sin(lj\pi h) \\
&= \frac{1}{8}[\sin((2j-2)l\pi h) + 4\sin((2j-1)l\pi h) + 6\sin(2jl\pi h) \\
&+ 4\sin((2j+1)l\pi h) + \sin((2j+2)l\pi h)], \\
&= \frac{1}{8}\left[2\cos(2l\pi h) + 8\cos(l\pi h) + 6\right]\sin(2lj\pi h), \\
&= \left[\cos(l\pi h) + \frac{1}{4}\cos(2l\pi h) + \frac{3}{4}\right]\sin(2lj\pi h),
\end{aligned}$$

for $j, l = 1, 2, \dots \frac{n}{2}$. For the second part of the basis corresponding to the eigenvectors $\phi_h^{n+1-l_j}$, we again use that

$$\begin{aligned}\phi_h^{n+1-l_j} &= -(-1)^j \sin(lj\pi h),\\ j &= 1,2,\dots n-1,\\ l &= 1,2,\dots \frac{n}{2}.\end{aligned}$$

Thus, the restriction operator maps the eigenvectors $\phi_h^{n+1-l_j}$ for $j = \frac{n}{2},\dots,n$ to

$$\begin{aligned}\left[I_{2h}^h \phi_h\right]_j^{n+1-l} &= [I_{2h}^h]_j^{n+1-l} \sin((n+1-l)j\pi h)\\ &= \frac{1}{8}[-\sin((2j-2)l\pi h) + 4\sin((2j-1)l\pi h) - 6\sin(2jl\pi h)\\ &+ 4\sin((2j+1)l\pi h) - \sin((2j+2)l\pi h)],\\ &= \frac{1}{8}\left[-2\cos(2l\pi h) + 8\cos(l\pi h) - 6\right]\sin(2lj\pi h),\\ &= \left[\cos(l\pi h) - \frac{1}{4}\cos(2l\pi h) - \frac{3}{4}\right]\sin(2lj\pi h).\end{aligned}$$

We therefore obtain the following 1×2 block for the restriction operator

$$[I_{2h}^h]^l = \begin{bmatrix}\cos(l\pi h) + \frac{1}{4}\cos(2l\pi h) + \frac{3}{4}\\ \cos(l\pi h) - \frac{1}{4}\cos(2l\pi h) - \frac{3}{4}\end{bmatrix}^T.$$

For ease of notation, we now define

$$\begin{aligned}v^l &= \cos(l\pi h) + \frac{1}{4}\cos(2l\pi h) + \frac{3}{4},\\ v^{n+1-l} &= \cos(l\pi h) - \frac{1}{4}\cos(2l\pi h) - \frac{3}{4}.\end{aligned}$$

Using these expressions, we can now compute the eigenvalue of the Galerkin coarse grid operator, which is given by the 1×1 diagonal block

$$\begin{aligned}\lambda^l(A_{2h}) &= [I_{2h}^h]^l A^l [I_h^{2h}]^l,\\ &= \left(v^l\right)^2 \lambda^l(A) + \left(v^{n+1-l}\right)^2 \lambda^{n+1-l}(A).\end{aligned} \qquad (29)$$

In order to obtain the eigenvalues of PA, we have to compute the 2×2 diagonal blocks of the projection operator P first. Recall that P is defined by

$$P^l = I - (I_{2h}^h)^l (A_{2h}^l)^{-1} (I_h^{2h})^l A^l.$$

We thus obtain the following block system

$$\begin{aligned}
P^l &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \frac{1}{\lambda^l(A_{2h})} \begin{bmatrix} (v^l)^2 & v^l v^{n+1-l} \\ v^{n+1-l} v^l & (v^{n+1-l})^2 \end{bmatrix}, \\
&= \begin{bmatrix} 1 - \frac{(v^l)^2}{\lambda^l(A_{2h})} & \frac{v^l v^{n+1-l}}{\lambda^l(A_{2h})} \\ \frac{v^{n+1-l} v^l}{\lambda^l(A_{2h})} & 1 - \frac{(v^{n+1-l})^2}{\lambda^l(A_{2h})} \end{bmatrix} \begin{bmatrix} \lambda^l(A) & 0 \\ 0 & \lambda^{n+1-l}(A) \end{bmatrix}, \\
&= \begin{bmatrix} \lambda^l(A) \left(1 - \frac{(v^l)^2}{\lambda^l(A_{2h})}\right) & \lambda^{n+1-l}(A) \left(\frac{v^l v^{n+1-l}}{\lambda^l(A_{2h})}\right) \\ \lambda^l(A) \left(\frac{v^{n+1-l} v^l}{\lambda^l(A_{2h})}\right) & \lambda^{n+1-l}(A) \left(1 - \frac{(v^{n+1-l})^2}{\lambda^l(A_{2h})}\right) \end{bmatrix}. \qquad (30)
\end{aligned}$$

From here, we retrieve the eigenvalues of PA by multiplying eq. (30) again with the 2×2 block containing the eigenvalues of A with respect to index l on our defined basis.

$$\begin{aligned}
[PA]^l &= \begin{bmatrix} \lambda^l(A) \left(1 - \frac{(v^l)^2}{\lambda^l(A_{2h})}\right) & \lambda^{n+1-l}(A) \left(\frac{v^l v^{n+1-l}}{\lambda^l(A_{2h})}\right) \\ \lambda^l(A) \left(\frac{v^{n+1-l} v^l}{\lambda^l(A_{2h})}\right) & \lambda^{n+1-l}(A) \left(1 - \frac{(v^{n+1-l})^2}{\lambda^l(A_{2h})}\right) \end{bmatrix} \begin{bmatrix} \lambda^l(A) & 0 \\ 0 & \lambda^{n+1-l}(A) \end{bmatrix}, \\
&= \begin{bmatrix} (\lambda^l(A))^2 \left(1 - \frac{(v^l)^2}{\lambda^l(A_{2h})}\right) & (\lambda^{n+1-l}(A))^2 \left(\frac{v^l v^{n+1-l}}{\lambda^l(A_{2h})}\right) \\ (\lambda^l(A))^2 \left(\frac{v^{n+1-l} v^l}{\lambda^l(A_{2h})}\right) & (\lambda^{n+1-l}(A))^2 \left(1 - \frac{(v^{n+1-l})^2}{\lambda^l(A_{2h})}\right) \end{bmatrix}. \qquad (31)
\end{aligned}$$

Note that each 2×2 block has a non-zero and zero eigenvalue. As a result, we obtain the non-zero eigenvalues of the system PA by computing the trace of each respective block of eq. (31) as in [11]

$$\begin{aligned}
\lambda^l(PA) &= (\lambda^l(A))^2 \left(1 - \frac{(v^l)^2}{\lambda^l(A_{2h})}\right) + (\lambda^{n+1-l}(A))^2 \left(1 - \frac{(v^{n+1-l})^2}{\lambda^l(A_{2h})}\right), \qquad (32) \\
l &= 1, 2, \ldots, \frac{n}{2}.
\end{aligned}$$

Similarly, the eigenvalues of $P^T M^{-1} A$ are obtained by simply multiplying eq. (30) with the 2×2 block containing the eigenvalues of $M^{-1}A$ instead of A and computing the trace. This operation leads to the following analytical expressions for the eigenvalues of $P^T M^{-1} A$

$$\begin{aligned}
\lambda^l(P^T M^{-1} A) &= \frac{(\lambda^l(A))^2}{\lambda^l(M)} \left(1 - \frac{(v^l)^2}{\lambda^l(A_{2h})}\right) + \frac{(\lambda^{n+1-l}(A))^2}{\lambda^l(M)} \left(1 - \frac{(v^{n+1-l})^2}{\lambda^l(A_{2h})}\right), \qquad (33) \\
l &= 1, 2, \ldots, \frac{n}{2}.
\end{aligned}$$

Using these expressions, we proceed with the spectral analysis of the DEF-preconditioner.

4.2 Spectral Analysis

In order to keep track of both deflation based preconditioned systems, we will use the following notation

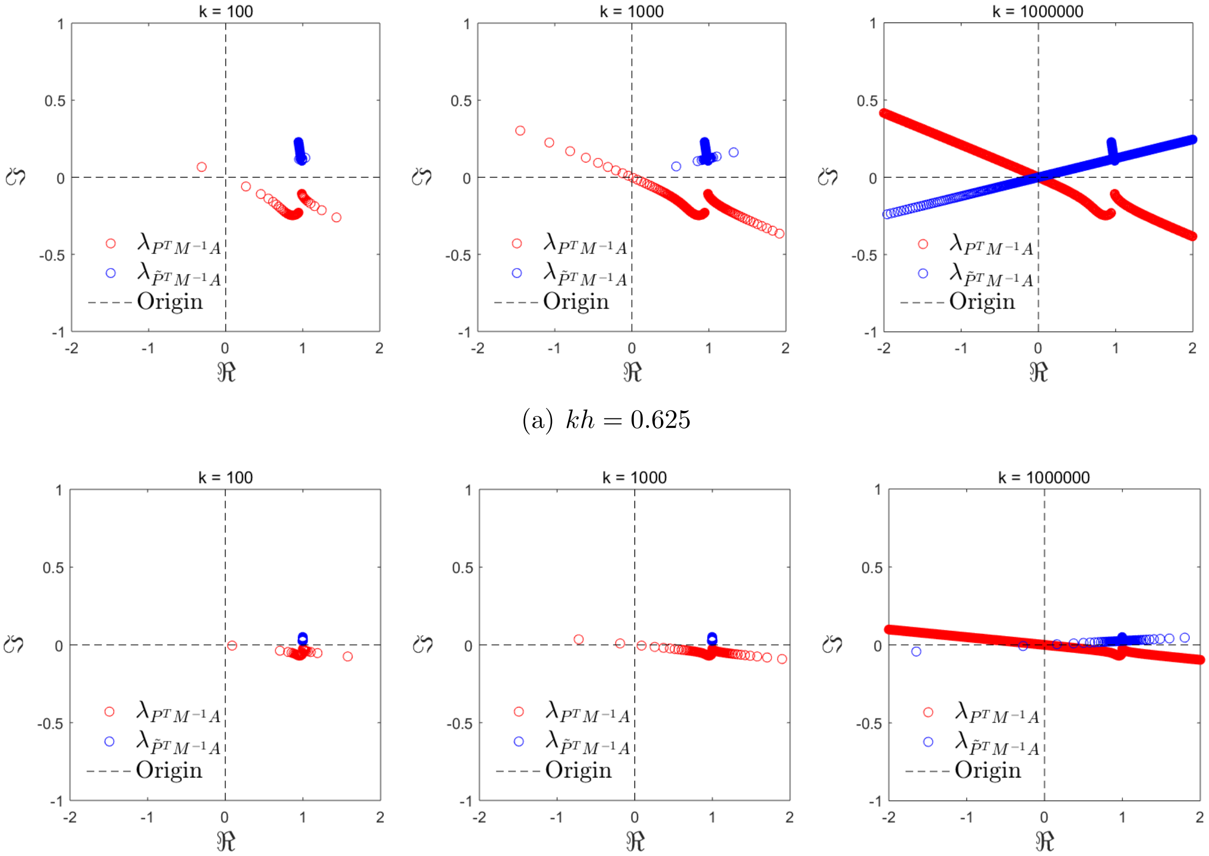
$$\begin{aligned}
I &= \text{Original prolongation/restriction,} \\
\tilde{I} &= \text{Adapted prolongation/restriction,} \\
A_{2h} &= \text{Original coarse-grid operator,} \\
\tilde{A}_{2h} &= \text{Adapted coarse-grid operator,} \\
PA &= \text{DEF,} \\
\tilde{P}A &= \text{Adapted Deflation,} \\
P^T M^{-1} A &= \text{DEF + CSLP,} \\
\tilde{P}^T M^{-1} A &= \text{Adapted Deflation + CSLP.}
\end{aligned}$$

We will now compare the spectrum of the DEF + CSLP preconditioned system ($P^T M^{-1} A$), with the adapted Deflation + CSLP precondtioned system ($\tilde{P}^T M^{-1} A$) for MP 1. In fig. 3 we have plotted the spectrum of both $P^T M^{-1} A$ (red) and $\tilde{P}^T M^{-1} A$ (blue) for MP 1 using large to very large wave numbers. In the top row we have plotted the eigenvalues for $kh = 0.625$, whereas the bottom row contains the eigenvalues for $kh = 0.3125$. Starting with the results for $kh = 0.625$, we note that incorporating the new deflation scheme leads to a significant reduction in the near-zero eigenvalues. For example for $k = 10^4$, there are almost no near-zero eigenvalues. However, as k increases to 10^6, we see the near-zero eigenvalues reappearing. Compared to the original DEF-scheme, the spectrum of the adapted scheme is more densely spread near the point $(1, 0)$. As a result, the spectrum of the adapted scheme has shorter tails. If we switch to a finer grid using $kh = 0.3125$, fig. 3 (b) illustrates that the new scheme almost completely dissolves the clustering spectrum near the origin. For $k = 10^6$ we do see a few eigenvalues moving slightly towards the origin, however these results are negligible compared to the magnitude of the wave number. Based on the spectral analysis and comparison between $P^T M^{-1} A$ (red) and $\tilde{P}^T M^{-1} A$ (blue), we expect the Krylov-based solver to converge much faster. This will be examined in the next section. One reason for the improvement in the spectrum of $\tilde{P}^T M^{-1} A$ is due to the better accuracy of the approximation of the eigenvectors when moving from a coarse-grid to a fine-grid and vice versa. In order to confirm this, table 2 contains the projection error according to corollary 3.2.1 for both schemes for large values of k. The projection error is reduced by a factor of 100 compared to the case where we use the old approximation scheme. However, for large k we again note that the projection error increases. These results are in line with the spectral analysis and explain why, even for the new scheme, we see some small eigenvalues reappearing near zero for $k = 10^6$ in fig. 3 (a).

4.2.1 Parameter Sensitivity

We have seen that for very large k such as $k = 10^6$, the adapted scheme using $\tilde{P}$ still has a small number of near-zero eigenvalues. This result is supported by the increasing

Figure 3: Eigenvalues of $P^T M^{-1} A$ and $\tilde{P}^T M^{-1} A$. The top row contains the spectrum of $P^T M^{-1} A$ and $\tilde{P}^T M^{-1} A$ for $kh = 0.625$. The bottom row contains the eigenvalues for $kh = 0.3125$.



(a) $kh = 0.625$

(b) $kh = 0.3125$

Table 2: Projection error for the old scheme E and the adapted scheme $\tilde{E}$.

k	E	$\tilde{E}$	E	$\tilde{E}$
	$kh = 0.625$		$kh = 0.3125$	
10^1	0.0672	0.0049	0.0077	0.0006
10^2	0.8818	0.0154	0.1006	0.0008
10^3	9.2941	0.1163	1.0062	0.0014
10^4	92.5772	1.1021	10.0113	0.007
10^5	926.135	10.9784	100.1382	0.0635
10^6	9261.7129	109.7413	1001.3818	0.6282

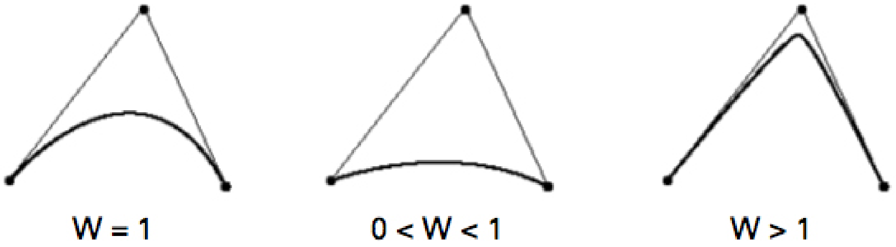
projection error for $kh = 0.625$ (see table 2), One explanation is that for these large wave numbers, the low-frequency eigenmode corresponding to l^h_{min} for A and l^{2h}_{min} for $\tilde{A}_{2h}$ are still very oscillatory vectors. Furthermore, apart from these eigenmodes themselves being relatively oscillatory, the high frequency modes which get activated are again a source for approximation errors when prolonging the coarse-grid eigenvectors. Necessarily, at some point, the scheme based on the adapted deflation vectors will again suffer from accumulation errors as their approximation power reduces as k increases. This on its term affects the location of the near-kernel modes of $\tilde{A}_{2h}$, i.e. resulting in $l^h_{\text{min}} \neq l^{2h}_{\text{min}}$ and inevitably a large gap between the smallest eigenvalues of A and A_{2h} respectively. As we have shown in section 2.3.1, $\hat{\beta}^l$ gives an accurate reflection of the behavior of the near-zero eigenvalues. As soon as these errors start accumulating, this will lead to $\hat{\beta}^l$ being small for $l = l^h_{\text{min}}$ and large for $l = l^{2h}_{\text{min}}$, causing the outliers and small eigenvalues near zero to reappear.

A very obvious yet in the long run expensive remedy would be to increase the number of grid points per wave length. For example, we observe in table 2 that using $kh = 0.3125$ leads to a severe reduction in the projection error for very large k. However, we are interested in a solution which does not require increasing the problem size.

In order to find such a solution, we note that the piecewise character of Bézier curves imply that at systematic intervals some discontinuities appear as sharp corners at certain points [17]. We have seen that as long as the grid is well-resolved, then even at high wave numbers the eigenvectors will be approximated accurately, and these discontinuities do not get amplified. If the eigenvectors become oscillatory due to the wave number being very large, then keeping the grid resolution constant, these discontinuities become a source of approximation error. This is exactly the phenomena we notice with respect to the linear interpolation scheme, however, the latter already suffers from inaccurate curvature approximation at relatively smaller wave numbers due to the use of linear piecewise segments.

Instead of diverting to higher-order approximation schemes, the use of rational Bézier curves allow simple modifications which can alter the shape and movement of the utilized curve segments. In fact, the weights of the rational Bézier curve are shape parameters, which allow control over the curve segments. For example, increasing the weight corresponding to a control point forces the curvature to move more closely and sharply to that control point. Decreasing the weight of a control point, on the other hand, results in the curve flattening and expanding more towards its endpoints. An illustration of this effect is provided by fig. 4 [1]. In our case, the quadratic approximation using the rational Bézier curve has one control point per segment. This would lead to the following redefinition

Figure 4: *Effect of changing the weight W with respect to a control point.*



$$I_{2h}^{h}\,:\,\Omega_{2h}\rightarrow\Omega_{h},\quad u_{2h}\rightarrow I_{2h}^{h}u_{2h}$$

such that

$$I_{2h}^{h}\left[u_{2h}\right]_{i}=\left\{\begin{array}{ll}\left(\frac{1}{8}\left[u_{2h}\right]_{(i-2)/2}+(\frac{3}{4}-\varepsilon)\left[u_{2h}\right]_{(i)/2}+\frac{1}{8}\left[u_{2h}\right]_{(i+2)/2}\right) & \text{if } i \text{ is even,}\\ \frac{1}{2}\left(\left[u_{2h}\right]_{(i-1)/2}+\left[u_{2h}\right]_{(i+1)/2}\right) & \text{if } i \text{ is odd,}\end{array}\right) \tag{34}$$

for $i=1,\dots,n-1$, and $\varepsilon>0$ The new scheme eq. (34) alters the expressions for the eigenvalues of $\tilde{P}^{T}M^{-1}A$ according to

$$\begin{aligned} v^{l} &= \cos(l\pi h)+\frac{1}{4}\cos(2l\pi h)+(\frac{3}{4}-\varepsilon),\\ v^{n+1-l} &= \cos(l\pi h)-\frac{1}{4}\cos(2l\pi h)-(\frac{3}{4}-\varepsilon),\end{aligned}$$

where now the expressions for $\tilde{P}^{T}M^{-1}A$ are again given by

$$\begin{aligned}\lambda^{l}(\tilde{P}^{T}M^{-1}A)=\frac{(\lambda^{l}(A))^{2}}{\lambda^{l}(M)}\left(1-\frac{(v^{l})^{2}}{\lambda^{l}(A_{2h})}\right)+\frac{(\lambda^{n+1-l}(A))^{2}}{\lambda^{l}(M)}\left(1-\frac{(v^{n+1-l})^{2}}{\lambda^{l}(A_{2h})}\right), \\ l=1,2,\dots,\frac{n}{2}.\end{aligned} \tag{35}$$

The next question which needs to be answered is, given a fixed kh, how do we find ε? ε should be chosen such that the projection error E is minimized. In order to find this value, we can use two approaches. The first approach is straightforward; our ultimate aim is to have the eigenvalue of $\lambda^{l}(\tilde{P}^{T}M^{-1}A)$ at index $l_{\min,h}$ to be equal to 1. Recall from the proof

[1] Image source: https://docs.derivative.ca/index.php?title=Spline.

of corollary 3.1.1 that in the absence of errors the eigenvalues of A_{2h} can be written as a product of the eigenvalues of A and the eigenvalues of B. Thus, using equation eq. (29), we can write

$$\begin{aligned} \lambda^l(A_{2h}) &= [I_{2h}^h]^l A^l [I_h^{2h}]^l, \\ &= \left(v^l\right)^2 \lambda^l(A) + \left(v^{n+1-l}\right)^2 \lambda^{n+1-l}(A), \\ &= \lambda^l(A)\lambda^l(B). \end{aligned} \tag{36}$$

Note that the sum of $\left(v^l\right)^2$ and $\left(v^{n+1-l}\right)^2$ in expression eq. (36) are exactly equal to $\lambda^l(B)$. If we want eq. (36) to hold at index $l_{\min,h}$ in the presence of errors, we need to pick ε such that $\left(v^{n+1-l}\right)^2 = 0$, which is equivalent to

$$\varepsilon = \frac{3}{4} - (\cos(l\pi h) - \frac{1}{4}\cos(2l\pi h)). \tag{37}$$

This way the near-zero eigenvalue of A_{2h} will always be proportional to the near-zero eigenvalue of A. Fortunately, the eigenvalues of B containing the term ε are independent of the eigenvalues of A. Therefore, finding ε primarily depends on the approximation scheme which determines the eigenvalues of B. An interesting observation is that ε is completely determined by the step-size h and therefore by the grid resolution kh. Thus, once we find the right ε for a given kh for a very large k, we can expect the solver to be scalable up to that k. Another method to find ε is to use the heuristic in Algorithm 2. The algorithm finds the minimizing ε with respect to projection error. The latter method provides a practical alternative to computing analytical expressions for the eigenvalues of B. However, if time permits we will extend the spectral analysis for MP 2 and MP 3 in order to determine the exact ε. We then expect to obtain a scalable solver in higher dimensions irrespective of k and kh. For now we proceed by using Algorithm 2 for the higher dimensional model problems.

Algorithm 2: Projection Error Minimizer

```
Initialize $k = 1 : \tilde{k}$ and an initial $\varepsilon_0 > 0$
for $k = 1, 2, ..\tilde{k}$ do
    Compute $E_{\varepsilon_0}$ using corollary 3.2.1
    Compute mean $E_{\varepsilon_0} = \bar{E}_{\varepsilon_0}$
    while $\bar{E}_{\varepsilon_j} > \bar{E}_{\varepsilon_{j-1}}$ do
        Pick an $\varepsilon_{j+1}$ and repeat until $\bar{E}_{\varepsilon_{j+1}} < \bar{E}_{\varepsilon_j}$
    end while
end for
```

As mentioned earlier, once we find ε, this will hold for any k as the accuracy of deflation based projection methods only depend on the grid resolution and step-size [5].

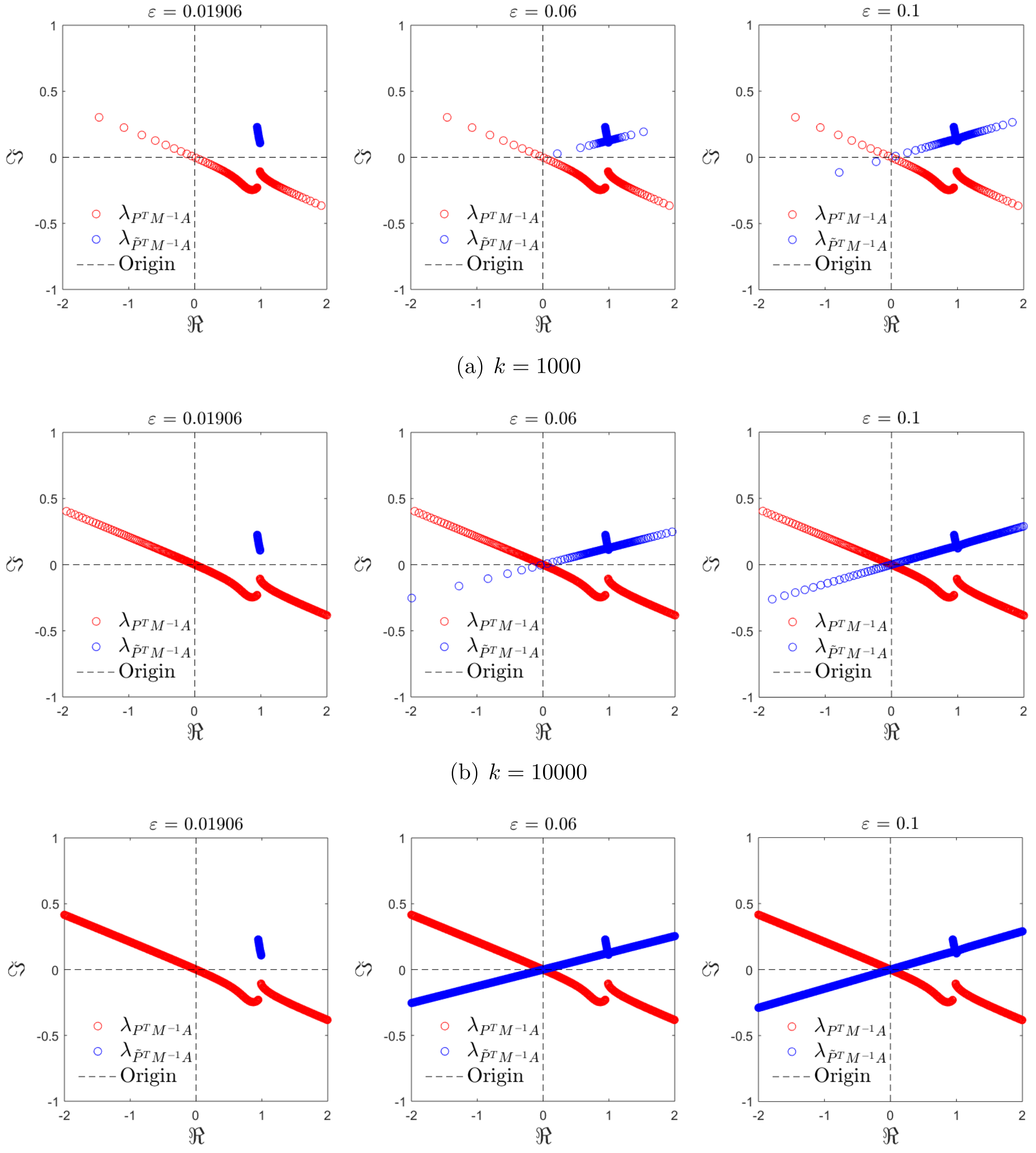
We proceed by re-examining the spectrum after introducing the weight-parameter. We have plotted the eigenvalues for $kh = 0.625$ for $\varepsilon = 0.05$ (left), $\varepsilon = 0.01906$ (center) and $\varepsilon = 0$ (right) in fig. 5. It immediately becomes apparent that using the right parameter to minimize the projection error completely shifts the spectrum. Particularly, the left column contains the results where the optimal ε has been used and it can be noted that the spectrum stays clustered near $(1, 0)$ independent of the wave number k. In all cases, the altered spectrum has a more favorable distribution relative to the original spectrum of the DEF-operator (red).

5 Numerical Experiments

5.1 One-dimensional Constant Wave Number Model

We start by examining the convergence behavior of the adapted solver using various kh. Note that unless kh is fine enough, the numerical solution suffers from pollution error. For now, we solely test for theoretical purposes using both coarse and fine grid resolutions in order to assess the scalability of the solver. In table 3 we have reported the number of preconditioned GMRES-iterations using a zero initial guess. In all scenarios we have used the CSLP-preconditioner with $(\beta_1, \beta_2) = (1, 0.5)$ and the tolerance level for the relative residual has been set to 10^{-7}. ε has been determined by using eq. (37). We start with the case where $\varepsilon = 0$ (bold) and no weight-parameter is incorporated. In case of $k = 10^6$, it takes 509 iterations to reach convergence. These results are in line with with the spectral analysis from fig. 3 and the reported increasing projection error from table 2. In particular, we observed small near-zero eigenvalues reappearing for $k = 10^6$. Also, the projection error for the new scheme was 109.7. As soon as we allow for corrections by means of introducing the weight-parameter, we observe that for each of the reported grid resolutions, we obtain a scalable solver. An interesting observation is that even at coarse-grids we obtain a constant number of iterations. However, on these coarser levels, the number of iterations is higher. For example, compare the 11 iterations for $kh = 1.25$ with the 5 iterations for $kh = 0.825$. In order to put these results into perspective, we report the projection error for each scheme in table 4. The results from table 4 confirm that the projection error has been reduced significantly. If we compare this to the previous results from table 1 and table 2, the projection error in all cases is now strictly smaller than 1 and constant for increasing k. These results are consistent with the spectral analysis for $k = 10^6$ and various kh in fig. 6. For all kh, the real part of the eigenvalues of the ADP-scheme stay clustered near 1. We now have a way of assessing the convergence properties of the Krylov-based solver by examining the behavior of the projection error. What is also interesting is that we may now use a higher-order finite difference scheme, combined with a coarser grid resolution in order to solve large scale problems. Another interesting property is that the significance of using a weight-parameter ε decreases with the grid resolution, i.e ε goes to zero as h becomes smaller.

Figure 5: Eigenvalues of $P^T M^{-1} A$ and $\tilde{P}^T M^{-1} A$ using $kh = 0.625$ for various weight-parameters ε.



(a) $k = 1000$

(b) $k = 10000$

(c) $k = 100000$

Table 3: Number of GMRES-iterations for the one-dimensional constant wave number problem for various kh using the Adapted Preconditioned Deflation scheme APD(ε). ε has been determined using eq. (37). The shift in CSLP has been set to $(1, 0.5)$.

k	APD(0.3050)	APD(0.1250)	APD(0.0575)	APD(0.01906)	**APD(0)**	APD(0.00125)
	$kh = 1.25$	$kh = 1$	$kh = 0.825$	$kh = 0.625$	$kh = 0.625$	$kh = 0.3125$
10^1	2	2	3	4	**4**	3
10^2	9	6	5	4	**4**	3
10^3	11	6	5	4	**6**	3
10^4	11	6	5	4	**12**	3
10^5	11	6	5	4	**59**	3
10^6	11	6	5	4	**509**	3

Table 4: Projection error E(ε) for various kh. ε has been determined using eq. (37).

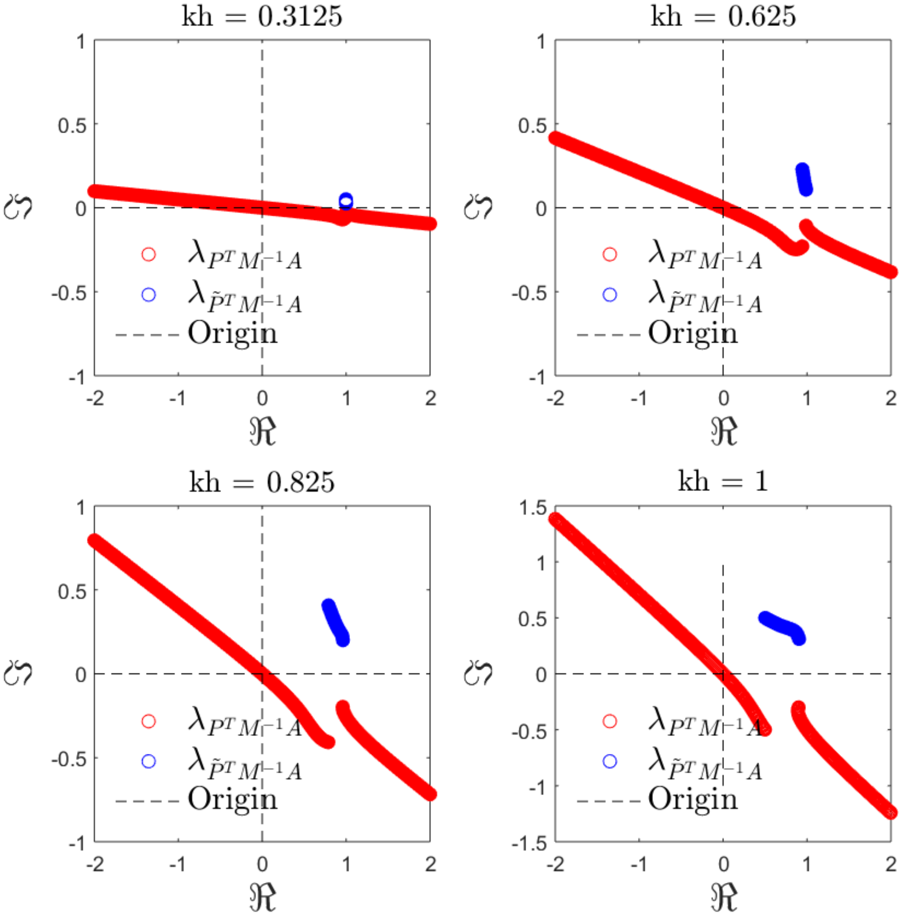
k	E(0.3050)	E(0.1250)	E(0.0575)	E(0.01906)	E(0.00125)
	$kh = 1.25$	$kh = 1$	$kh = 0.825$	$kh = 0.625$	$kh = 0.3125$
10^1	0.0699	0.0127	0.0075	0.0031	0.0006
10^2	0.1884	0.0233	0.0095	0.0036	0.0007
10^3	0.2215	0.0245	0.0095	0.0038	0.0007
10^4	0.2197	0.0246	0.0095	0.0038	0.0007
10^5	0.2200	0.0246	0.0095	0.0038	0.0007
10^6	0.2200	0.0246	0.0095	0.0368	0.0007

5.2 Two-dimensional Constant Wave Number Model

In this section perform numerical experiments for the two-dimensional model problem using a constant wave number k. This model problem is referred to as MP 2, see section 2. The results are presented in table 5 for $kh = 0.625$. The weight-parameter ε has been determined using Algorithm 2. Similar to the results for MP 1, we note that the number of iterations are again more or less independent of the wave number k. As k gets very large, we do see that the number of iterations increases. We expect that computing the analytical ε similar to eq. (37) for the one-dimensional case, will provide a wave number independent iteration count. However, compared to the DEF-scheme, the reported number of iterations is more favorable. A noteworthy result is that for the two-dimensional case, the use of the adapted scheme seems less sensitive to changes in the complex shift of the CSLP preconditioner.

We now repeat the same analysis for $kh = 0.3125$. Note that in this case we do not include an adjusted weight coefficient parameter, i.e. we set $\varepsilon = 0$. The reason for this is that increasing the problem size already results in more accuracy and faster convergence [18], [19]. We also compare the performance of the adapted scheme with and without the inclusion of the CSLP-preconditioner as the analysis from the previous part showed that the scheme is less sensitive to changes in shifts of the CSLP-preconditioner. For the inclusion of the CSLP-preconditioner, we use the shift $(1, 1)$.

Figure 6: *Eigenvalues for* $k = 10^6$ *of* $P^T M^{-1} A$ *and* $\tilde{P}^T M^{-1} A$ *using various* kh*. The weight-parameter* ε *has been determined using equation eq.* (37).



Results are reported in table 6. If we compare these results to the ones obtained from table 5, we note that increasing the problem size leads to faster convergence. However, compared to the results for the DEF-solver, the effect is much more subtle. This can be explained by the fact that already on a coarse grid, the transfer of the underlying eigenvectors are approximated with higher accuracy compared to the DEF-scheme which relies on an approximation using the standard interpolation and restriction scheme. We also note that the influence of the parameter ε becomes diminishable compared to the case where we used $kh = 0.625$. Thus, it seems that the inclusion of ε may in particular be more useful when using coarser grids. Also, the number of iterations with and without the CSLP-preconditioner is almost the same for all reported values of k and it may be argued that for fine grid resolutions, the inclusion of the CSLP-preconditioner becomes redundant. As the inclusion of the CSLP-preconditioner comes at a heavy computational cost, it can be efficient to exclude it from the solver. While utilizing a finer grid leads to less number of

Table 5: Number of iterations for the **two-dimensional** constant wave number problem for $kh = 0.625$. ε refers to the weight-parameter. The shift in CSLP has been set to $(1, 0.5)$ (column 4) and $(1, 1)$ (column 5) resp.

k	n^2	APD(0)	APD(0.0187)	APD(0.0187)
50	6400	4	4	5
100	25600	5	4	5
250	160000	10	5	5
500	640000	15	5	6
750	1440000	37	7	8
1000	2560000	53	8	9

iterations and more accurate numerical solutions, it inevitably leads to very large problem sizes.

Table 6: Number of iterations for the **two-dimensional** constant wave number problem for $kh = 0.3125$. AD contains no CSLP-preconditioner.

k	n^2	AD(0)	APD(0.00125)
		Iterations	Iterations
25	6400	4	4
50	25600	4	4
100	102400	3	4
250	640000	4	4
500	2560000	5	5
750	5760000	5	5
1000	10240000	7	8

5.3 Two-dimensional Non-constant Marmousi Model

In this section we present the numerical results for the industrial two-dimensional Marmousi problem (MP 4) eq. (5), section 2. Results are reported in table 7. We present results using the two-level method implemented in Matlab R2015b on a machine with processor i7-4790K at 4.00Ghz for both the DEF- and APD-schemes. Inversion is obtained by using the backslash solver in Matlab. With respect to the APD-scheme we implement no correction using ε given that the grid for this model problem has been resolved such that $kh \leq 0.39$ on average and the maximum wave number is approximately 400. [2] The

[2] If we use the dimensionless model we obtain a wave number of $\sqrt{\frac{2\pi 40}{2587.5}^2 \times 2048 \times 8192} \approx 398$.

first four rows of table 7 contain the results for frequencies $f = 1, 10, 20$ and 40 using 10 grids points per wave length for the largest wave number k. The last four rows use 20 grid points per wave length. The results show that even for this challenging problem, the APD-scheme leads to very satisfactory results. First of all, the number of iterations again seems fairly consistent irrespective of the grid points per wave length used. In both cases the number of iterations for these test problems are independent of frequencies. If we start comparing the results between DEF-TL and APD-TL, we note an improved performance in terms of both metrics; solve time and iterations. For $f = 1$, the number of iterations for APD-TL are larger than DEF-TL. The latter method takes 6 iterations, while the former takes 3 iterations, which is obviously reflected in the lower solve time. Once we start increasing the frequency, we note that the APD-TL scheme quickly catches up in terms of both iterations and solve time. For example for $f = 40$, we obtain 5 iterations and a total solve time of 111.78 seconds compared to the 1175.99 seconds DEF-TL method.

The last four rows present the results using 20 grid points per wave length. In terms of iterations, the results for ADP-TL are not much different. In both cases we note that the number of iterations do not grow with the frequency. However, given that utilizing a finer grid leads to a larger problem size, the solve time in general increases. Compared to DEF-TL, in terms of solve time, both methods at first do not seem to differ significantly. At $f = 40$, ADP-TL is approximately 300 seconds faster than DEF-TL.

Table 7: Results for the Marmousi problem using 10 gpw (upper) and 20 gpw respectively (lower). All solvers are combined with the CSLP-preconditioner using shifts (1,1). TL denotes two-level.

f	DEF-TL	APD-TL	DEF-TL	APD-TL
	Iterations (s)		Solve Time	
1	3	6	1.72	4.08
10	16	5	7.30	3.94
20	31	5	77.34	19.85
40	77	5	1175.99	111.78
1	3	4	9.56	15.45
10	7	6	19.64	3.83
20	10	6	155.70	122.85
40	15	6	1500.09	1201.45

5.4 Three-dimensional Constant Wave Number Model

In this section we present some three-dimensional numerical results for MP 3 (equation eq. (4)). The algorithm is still to be terminated when the relative residual has been reduced by order 10^7. Furthermore, we have used the same weight-parameter ε from the two-dimensional test problem MP 2. From table 8 we can see that even without the weight-parameter ε, the 3D-results show promising features for scalability. These results are also in line with the previous results obtained for the two-dimensional constant wave number model from table 6. We expect the importance of ε to decrease along with kh.

Table 8: Number of iterations for the **three-dimensional** constant wave number problem for $kh = 0.625$. AD contains no CSLP-preconditioner. APD contains the CSLP with shift $(1, 0.5)$.

k	n^3	APD(0)	APD(0.00125)
		Iterations	Iterations
5	512	4	4
10	4096	4	4
25	64000	5	4
50	512000	5	4
75	1728000	6	4

6 Conclusion

We have shown that the near-zero eigenvalues for deflation based preconditioners are related to the near-kernel eigenmodes of the fine-grid operator A and coarse-grid operator A_{2h} being misaligned. A very simple yet concise red flag is whether the indices of the smallest absolute eigenvalue of A and A_{2h} are different. This effect can be attributed to accumulating interpolation errors, due to the interpolation scheme not being able to sufficiently approximate the transferring of the grid functions at very large wave numbers. The root cause of this phenomena lies in the high frequency modes being activated.

We have presented the first scheme to analytically measure the effect of these errors on the construction of the projection operator. The latter operator defines the deflation preconditioner. The error can be measured by computing the projection error. Our results indicate that the quality of the deflation vectors determine whether the projection error dominates. To minimize the projection error, we propose the implementation of a higher order approximation scheme to construct the deflation vectors. For the first time, the spectral properties of the Helmholtz problem at such large wave numbers (the largest being 10^6) have been studied. In terms of spectral properties, the eigenvalues based on the adapted deflation scheme, even at large wave numbers for the one-dimensional case, are close to 1. However, at very large wave numbers, the near zero-eigenvalues reappear. Adjusting the weight-parameter within the approximation scheme seems to provide counterbalance to mitigate these near-zero eigenvalues. This only remedies computations at coarse grid resolutions as reducing kh diminishes the importance and necessity for incorporating a weight-parameter. Two options are available for determining the weight-parameter. The first option is to use the analytical eigenvalues of B at the smallest index $l_{\min,h}$ and solve for

ε. This approach is fairly straightforward to use as it primarily depends on the eigenvalues of B, which can be computed independently of the eigenvalues of A. The second approach is to use the projection error minimizing algorithm, which computes the projection error for various k and finds the ε which minimizes the norm on average.

Even without adjusting the weight-parameter, the spectrum of our proposed operator is still the most favourable compared to other preconditioning operators based on deflation for the Helmholtz equation. We have performed numerical testing and simulation of our model problems ranging from the simple one-dimensional constant wave number problem to the challenging industrial Marmousi problem. The numerical results are in line with the theoretical results as the number of iterations for both the one-, two- and three-dimensional constant wave number model problem are more or less wave number independent. For the one-dimensional case in particular, we determined the exact value for ε, which results in a wave number independent solver and a constant but small projection error. With respect to the higher-dimensions, ε can be approximated very closely by finding the value which minimizes the projection error. In the future, we will study the exact value of ε in higher-dimensions and test whether numerical results corroborate these presumptions.

The numerical tests were performed for very large wave numbers (10^6 for one-dimension and 10^3 for two-dimensions). In the two-dimensional case (MP 2), the inclusion of the CSLP-preconditioner becomes superfluous. For the three-dimensional model problem, the maximum wave number has been set to $k = 75$ due to memory constraints. Especially the results for the Marmousi problem are very encouraging as the number of iterations has been reduced significantly. A challenging yet exciting topic for future research would be to study the parallel implementation of the adapted deation based solver and/or the multilevel krylov implementation and apply it to the three-dimensional test problem. While these results on scalability of the solver are promising, they do not solve the problem of the pollution error. A simultaneous and better understanding of both topics is currently another subject of our research.

References

[1] Domenico Lahaye Abdul H Sheikh and Cornelis Vuik. On the convergence of shifted Laplace preconditioner combined with multilevel deflation. *Numerical Linear Algebra with Applications*, 20:645–662, 2013.

[2] Alvin Bayliss, Charles I Goldstein, and Eli Turkel. An iterative method for the helmholtz equation. *Journal of Computational Physics*, 49(3):443–457, 1983.

[3] Mohammed Bellalij, Yoursef Saad, and Hassane Sadok. Analysis of some krylov subspace methods for normal matrices via approximation theory and convex optimization. *Electronic Transactions on Numerical Analysis*, 33(17-30):2009, 2008.

[4] Marco Donatelli. A note on grid transfer operators for multigrid methods. *arXiv preprint arXiv:0807.2565*, 2008.

[5] Yogi Erlangga and Eli Turkel. Iterative schemes for high order compact discretizations to the exterior helmholtz equation. *ESAIM: Mathematical Modelling and Numerical Analysis*, 46(3):647–660, 2012.

[6] Yogi A Erlangga and Reinhard Nabben. On a multilevel krylov method for the helmholtz equation preconditioned by shifted laplacian. *Electronic Transactions on Numerical Analysis*, 31(403-424):3, 2008.

[7] Yogi A Erlangga, Cornelis W Oosterlee, and Cornelis Vuik. A novel multigrid based preconditioner for heterogeneous helmholtz problems. *SIAM Journal on Scientific Computing*, 27(4):1471–1492, 2006.

[8] Yogi A Erlangga, Cornelis Vuik, and Cornelis W Oosterlee. Comparison of multigrid and incomplete lu shifted-laplace preconditioners for the inhomogeneous helmholtz equation. *Applied numerical mathematics*, 56(5):648–666, 2006.

[9] Oliver G Ernst and Martin J Gander. Multigrid methods for helmholtz problems: A convergent scheme in 1d using standard components.

[10] Oliver G Ernst and Martin J Gander. Why it is difficult to solve helmholtz problems with classical iterative methods. In *Numerical analysis of multiscale problems*, pages 325–363. Springer, 2012.

[11] Luis Garcia Ramos and Reinhard Nabben. On the spectrum of deflated matrices with applications to the deflated shifted laplace preconditioner for the helmholtz equation. *SIAM Journal on Matrix Analysis and Applications*, 39(1):262–286, 2018.

[12] Markus Holtz and Angela Kunoth. B-spline-based monotone multigrid methods. *SIAM Journal on Numerical Analysis*, 45(3):1175–1199, 2007.

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

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

[15] Roy A Nicolaides. Deflation of conjugate gradients with applications to boundary value problems. *SIAM Journal on Numerical Analysis*, 24(2):355–365, 1987.

[16] Allan Pinkus. *On L1-approximation*, volume 93. Cambridge University Press, 1989.

[17] David F Rogers. *An introduction to NURBS: with historical perspective.* Elsevier, 2000.

[18] Abdul H Sheikh. *Development Of The Helmholtz Solver Based On A Shifted Laplace Preconditioner And A Multigrid Deflation Technique.* TU Delft, Delft University of Technology, 2014.

[19] Abdul H. Sheikh, Domenico Lahaye, L Garcia Ramos, Reinhard Nabben, and Cornelis Vuik. Accelerating the shifted laplace preconditioner for the helmholtz equation by multilevel deflation. *Journal of Computational Physics*, 322:473–490, 2016.