DELFT UNIVERSITY OF TECHNOLOGY

REPORT 20-04

Scalable multi-level deflation preconditioning for the highly indefinite Helmholtz equation

V. Dwarka and C. Vuik

ISSN 1389-6520

Reports of the Delft Institute of Applied Mathematics

Delft 2020

Scalable multi-level deflation preconditioning for the highly indefinite Helmholtz equation

Vandana Dwarka* Cornelis Vuik †

June 2020

Abstract

Recent research efforts aimed at iteratively solving the Helmholtz equation have focused on incorporating deflation techniques for accelerating the convergence of Krylov subpsace methods. In this work, we extend the two-level deflation method in [6] to a multilevel deflation method. By using higher-order deflation vectors, we show that up to the level where the coarse-grid linear systems remain indefinite, the near-zero eigenvalues of the these coarse-grid operators remain aligned with the fine-grid operator keeping the spectrum of the preconditioned system fixed away from the origin. Combining this with the well-known CSLP-preconditioner, we obtain a scalable solver with theoretical linear complexity for the highly indefinite Helmholtz equation. This can be attributed to a fixed number of iterations independent of the wave number and an optimal use of the CSLP-preconditioner. We approximate the CSLP-preconditioner, while allowing the complex shift to be small. The proposed configuration additionally shows very promising results for the more challenging Marmousi problem.

1 Introduction

The Helmholtz equation has puzzled the minds of many mathematicians and numerical analysts throughout the years. Its wide application, ranging from seismology to medical tomography, has kept its relevance even till this day. As a result, many efforts have and are still being been rendered in order to obtain accurate and computationally feasible solutions.
A large branch within this research has focused on developing preconditioners, such as the (Complex) Shifted Laplacian [10, 9, 13, 4]. In order to apply the preconditioner, one multigrid cycle is used to approximate its inverse. The latter serves as an alternative to using multigrid as a stand-alone solver as the method is generally known to diverge for

*Department of Applied Mathematics, Delft University of Technology, Delft, the Netherlands (v.n.s.r.dwarka@tudelft.nl).

†Department of Applied Mathematics, Delft University of Technology, Delft, the Netherlands (c.vuik@tudelft.nl, `homepage.tudelft.nl/d2b4e/`).

the Helmholtz equation once coarser levels are reached [12]. Some works have focused on obtaining a stand-alone multigrid solver [7, 20, 22, 11], with success for either practical wavenumbers and/or one-dimensional model problems.

A recent and promising branch of research has combined its efforts towards preconditioning techniques based on domain decomposition methods applied to the corresponding (shifted) problem [14]. These methods split the computational domain in subdomains and solve a local subproblem of smaller dimension using a direct method [5, 19, 17, 1, 18]. The performance of these preconditioners depends on the accuracy of the transmission conditions, which currently is robust for constant wave number model problems [16, 15]. While the domain decomposition preconditioners have resulted in a reduced number of iterations and higher computational efficiency by exploiting parallelization strategies, the number of iterations still grows with the wavenumber k.

As a result, some have studied the use of deflation techniques (combined with the CSLP-preconditioner) in order to accelerate the convergence of the Krylov subspace method, which we will denote DEF. [24, 25, 26]. Incorporating the deflation preconditioner has improved the convergence, but taxed the efficiency in terms of memory and computational cost. For a two-level deflation preconditioner, the direct solve on the second level takes up most of the computational power and memory. Consequently, multilevel variants of the two-level method have been proposed in order to counter this effect [8, 25]. A multilevel extension replaces the direct solve in the two-level method by applying a similar two-level extension recursively combined with an outer Flexible GMRES (FGMRES) solver. The CSLP-preconditioner is then applied on each level through one multigrid cycle.

In both variants, however, the number of iterations still slowly grows with the wave number k. In this work, we build on our recent work from [6] where we developed and tested a two-level deflation preconditioner which rendered close to wavenumber independent convergence for large wavenumbers in all spatial dimensions. We will refer to this method as the Adapted Deflation Preconditioner (ADP), where the adaption is realized through the use of higher-order interpolation polynomials. A natural question which arises is whether we can extend the wavenumber independent convergence to a multilevel setting, thereby combining both the gain in computational efficiency with our previous scalability results.

The structure of this paper is as follows. We start by introducing our model problems in section 2. We then discuss the deflated Krylov methods and the multilevel algorithm in section 3. We then proceed by extensively developing theory for the multilevel deflation operator in section 4. We perform Rigorous Fourier Analysis (RFA) by block-diagonalizing the resulting operators and inspecting the spectral properties. Finally we present numerical results for benchmark problems in section 5.

2 Problem Description

We start by focusing on a one-dimensional mathematical model using a constant wave number $k > 0$

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

We will refer to this model problem as MP 1-A. To allow for more practical examples, we introduce MP 1-B as the model problem where Sommerfeld radiation conditions have been implemented. In this case, the boundary conditions become

$$\left(\frac{\partial}{\partial \mathbf{n}} - ik\right) u(x) = 0, x \in \partial[0, L].$$

If we define $h = \frac{1}{n}$, where n is chosen according to $kh = \frac{2\pi}{c}$, where c is the number of grid points per wavelength, then discretization on the unit interval using second order finite differences leads to

$$\frac{-u_{j-1} + 2u_j - u_{j+1}}{h^2} - k^2 u_j = f_j, j = 1, 2, \ldots, n.$$

Lexicographic ordering leads to the following linear system and eigenvalues for MP 1-A with indices $j = 1, 2, \ldots n$

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

Similarly, we define the 2-D and 3-D versions of model problem MP 1-B as above eq. (1). The discretization using second order finite differences goes accordingly for higher dimensions with the needed alterations at the boundary when using Sommerfeld conditions.
The final test problem is a representation of an industrial problem and is widely referred to as the 2D Marmousi Problem, which we denote by MP-4. We consider an adapted version of the original Marmousi problem developed in [24]. 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 a power of 2. The original velocity $c(x, y)$ is also adapted by letting $2587.5 \leqslant c \leqslant 3325$. On the adjusted domain Ω, we define

$$\begin{aligned} -\Delta u(x, y) - k(x, y)^2 u(x, y) &= \delta(x - 4000, y), (x, y) \in \Omega \backslash \partial\Omega \subset \mathbb{R}^2, \\ \left(\frac{\partial}{\partial \mathbf{n}} - ik\right) u(x, y) &= 0, (x, y) \in \partial\Omega, \end{aligned} \tag{3}$$

where $\mathbf{n}$ denotes the outward normal unit vector. Note that we now have a non-constant wave number $k(x, y) = \frac{2\pi f}{c(x,y)}$, where the frequency f is given in Hertz.

3 Deflated Krylov Methods

We start by briefly explaining the two-level deflation preconditioning technique to solve the resulting linear system. We then proceed by extending the two-level method recursively to a multilevel Krylov method.

3.1 Two-level Deflation

For a linear system $Au = f$ we construct the deflation preconditioner P where the column space of Z is used as the deflation subspace. Z can be interpreted as interpolating from the coarse grid to the fine grid.

$$P = I - AQ \text{ where } Q = ZE^{-1}Z^T \text{ and } E = Z^TAZ$$

In [6], we used higher-order Bezier curves to construct Z. Using these higher-order polynomials, the prolongation and restriction operator act on a grid function as follows

$$Z\left[u_{2h}\right]_i = \left\{ \begin{array}{cc} \frac{1}{8}\left(\left[u_{2h}\right]_{(i-2)/2} + 6\left[u_{2h}\right]_{(i)/2} + \left[u_{2h}\right]_{(i+2)/2}\right) & \text{if } i \text{ is even,} \\ \frac{1}{2}\left(\left[u_{2h}\right]_{(i-1)/2} + \left[u_{2h}\right]_{(i+1)/2}\right) & \text{if } i \text{ is odd,} \end{array} \right\}, \tag{4}$$

for $i = 1, \ldots, n-1$ and for $i = 1, \ldots, \frac{n}{2}$. To obtain even better convergence, the CSLP-preconditioner was included, which is given by

$$M = L - (\beta_1 + \sqrt{-1}\beta_2)k^2 I,$$

where $(\beta_1, \beta_2) \in [0, 1]$ and L is the discretisized Poisson equation. In compliance with the literature, we keep $\beta_1 = 1$. The system to be solved becomes $M^{-1}PAu = M^{-1}Pf$.

By allowing higher-order interpolation schemes, the near-zero eigenspace of the fine- and coarse-grid coefficient matrix remains perfectly aligned. As a result, the smallest eigenvalue in magnitude of both A and E is located at the same index. This prevents the eigenvalues of the deflated system from shifting towards the origin. While the method provides close to wavenumber independent convergence in one- and two-dimensions for fairly large wavenumbers $k = 10^6$ (1D) and $k = 10^3$ (2D). The method requires the exact solve of the coarse-grid coefficient matrix E, adding to the computational complexity in 3D, where we obtained wave number independent convergence up to $k = 75$ (3D). In order to circumvent the direct solve, we extend the two-level to a multilevel deflation method.

3.2 Multilevel Deflation

We start by noting that the inexact inversion requires the addition of an extra term Q in order to prevent synthetic close-to-zero eigenvalues from obstructing the convergence of the Krylov solver [21, 8, 23]. The multilevel deflation algorithm is given below

Algorithm 1: Multilevel ADP Implementation

Initialization;
Construct $A^{(1)}, M^{(1)}$
for $i = 1, 2, ...m$ *the coarsest level* **do**
  Construct $Z^{(i,i+1)}$ and $Z^{(i,i+1)T}$
  Construct $A^{(i+1)} = Z^{(i,i+1)}A^{(i)}Z^{(i,i+1)T}$
  Construct $M^{(i+1)} = Z^{(i,i+1)}M^{(i)}Z^{(i,i+1)^T}$
end
Start $i = 1$
Solve: $A^{(1)}u^{(1)} = b^{(1)}$ with Krylov preconditioned by
 $P^{(1)}$, where $P^{(m)} = I^{(m)} - A^{(m)}Q^{(m)} + Q^{(m)}$
$u^{(1)}$ vector to be preconditioned
Restrict: $\hat{u}^{(2)} = Z^{(1,2)T}u^{(1)}$
if $m = 1$ **then**
  $u^{(2)} = (A^{(2)})^{-1}\hat{u}^{(2)}$ using direct solver
else
  $i = 2$
  Solve: $A^{(2)}u^{(2)} = b^{(2)}$ with Krylov preconditioned by $P^{(2)}$
  $u^{(2)}$ vector to be preconditioned
  Restrict: $\hat{u}^{(3)} = Z^{(2,3)T}u^{(2)}$
  if $m = 2$ **then**
    $u^{(3)} = (A^{(3)})^{-1}\hat{u}^{(3)}$ using direct solver
  else
    $i = 3$
    Solve: $A^{(3)}u^{(3)} = b^{(3)}$ with Krylov preconditioned by $P^{(3)}$
    ...
    ...
    Interpolate: $q^{(2)} = Z^{(2,3)}u^{(3)}$
    $\hat{t}^{(2)} = u^{(2)} - A^{(2)}q^{(2)}$
    $t^{(2)} = (M^{(2)})^{-1}\hat{t}^{(2)}$
    $w^{(2)} = t^{(2)} + q^{(2)}$
    $p^{(2)} = A^{(2)}w^{(2)}$
  end
  Interpolate: $q^{(1)} = Z^{(1,2)}u^{(2)}$
  $\hat{t}^{(1)} = u^{(1)} - A^{(1)}q^{(1)}$
  $t^{(1)} = (M^{(1)})^{-1}\hat{t}^{(1)}$
  $w^{(1)} = t^{(1)} + q^{(1)}$
  $p^{(1)} = A^{(1)}w^{(1)}$
end

So far, recent works have used one multigrid cycle to approximate the preconditioning step on the vector t as this is an $\mathcal{O}(n)$ operation. However, the shift β_2 in the CSLP-preconditioner has to be kept large enough for multigrid to converge [12, 11, 4]. Another option is by allowing a few GMRES-iterations to approximate the preconditioner. For example, in the context of using multigrid as a preconditioner, the standard smoothing step is replaced by a few GMRES-iterations on coarse grids in order to replace the unstable Jacobi and Gauss-Seidel smoother [7, 2, 3]. Within our configuration, this can be beneficial as it enables a small shift β_2, which has been shown to accelerate convergence [13]. We will show in section 5 that we can gain tremendous efficiency by using a very low tolerance (10^{-1}) and a fixed number of iterations.

4 Inscalability

In this section we will extend the theoretical results of the two-level ADP-scheme to a multilevel setting for MP 1-A. Given that the coefficient matrix remains normal, spectral analysis can be performed to assess the convergence behavior. We have provided a detailed summary of the literature as regards the role of the eigenvalues when the matrix is non-normal in [6].

4.1 Multilevel mapping

We start with the following theorem

Theorem 1. *Multilevel Prolongation and Restriction (linear) Let Z_m be the $n_{m-1} \times n_m$ prolongation matrix based on linear interpolation for $m = 1, 2, \ldots m_{\max}$, with $n_m = \frac{n}{2^m}$. If we define $v_m^j = \sin(2^m hi\pi j)$, and $v_m^{j'} = \sin(2^m hi\pi(n_m + 1 - j))$, where on the finest level we have $m = 0$. Then there exist constants C_1^j and C_2^j depending on h such that restriction operator maps the eigenvectors to*

$$\prod_{l=m}^{1} Z_l^T v_0^j = C_1^j v_m^j, \ j = 1, 2, \ldots, n_m,$$
$$\prod_{l=m}^{1} Z_l^T v_0^{j'} = C_2^j v_m^j, \ j = 1, 2, \ldots, n_m.$$

where $C_1^j = \left(\frac{1}{2}\right)^m \prod_{l=1}^m \left(1 + \cos(j\pi 2^{l-1} h)\right)$ and $C_2^j = \left(\frac{1}{2}\right)^m \prod_{l=1}^m \left(\cos(j\pi 2^{l-1} h) - 1\right)$. Similarly, the prolongation operator maps the eigenvectors to

$$\prod_{l=1}^{l} Z_l [v_m]_i = C_1^j [v_0^j]_i, \ \textit{for } i \textit{ is odd.},$$
$$\prod_{l=1}^{l} Z_l [v_m]_i = C_2^j [v_0^j]_i, \ \textit{for } i \textit{ is even.}.$$

Finally, if we let $B_m = \prod_{l=1}^m Z_l \prod_{l=m}^1 Z_l^T$ and $\hat{B}_m = Z_m Z_m^T$ for $m = 1, 2, \ldots, m_{\max}$, then B_m has dimension n_0 with n_m non-zero eigenvalues.

Proof. This proof is structured as follows. First we will define the mapping operators and the respective vector spaces and their bases to which they are applied. Then we will continue by showing the action of the restriction operator on the basis for these vectors spaces. To keep an overview of what is happening between the vector spaces on an abstract level, we use both the analytical operator and their matrix representations in the proof. We then do the same for the prolongation operator. Finally, we show that the kernel and range of the composite mapping consisting of the restriction and prolongation operator span a subspace containing the eigenvectors. We use this to show that the eigenvalues of B_m are related to the eigenvalues of $\hat{B}_m$.

<u>Basis and ordering</u>
We start by defining $n_m = \frac{n}{2^m}$ and rearranging the space spanned by the eigenvectors at each level such that we obtain the following subspace

$$\mathcal{V}_m^j = \text{span}\,\{v_m^j, v_m^{n_m+1-j}\},$$

for $j = 1, 2, \ldots, n_{m+1}$. Moreover let

$$V_{m+1}^j = \bigoplus_{j=1}^{n_{m+1}} \text{span}\,\{v_{m+1}^j\},$$

denote the space spanned by the eigenvectors at a coarser level $m+1$. Note that the basis spans $\mathbb{C}^{n_m}$ and $\mathbb{C}^{n_{m+1}}$ as we can write

$$\mathbb{C}^{n_m} = \bigoplus_{j=1}^{n_{m+1}} \mathcal{V}_m^j \text{ and } \mathbb{C}^{n_{m+1}} = \bigoplus_{j=1}^{n_{m+1}} V_{m+1}^j,$$

and at each subsequent level $m+1$ we re-order the basis to obtain $\mathcal{V}_{m+1}$. Thus, on each level we define the automorphism such that we can bring the basis of V_m in to the order of $\mathcal{V}_m$

$$\alpha_{\pi(j)}^m : V_m \to V_m : j \mapsto n_m + 1 - (j-1) \text{ for } j \text{ is even.}$$

For $m = 0, 1, 2 \ldots m_{\max}$, the linear interpolation and restriction operator maps between subsequent vector spaces

$$\begin{aligned}
\mathcal{I}_m^{m+1} &: \mathcal{V}_m \to V_{m+1}, \text{ such that } \mathcal{V}_m^j \mapsto \mathcal{I}_m^{m+1} V_m^j \\
\mathcal{I}_{m+1}^m &: V_{m+1} \to \mathcal{V}_m, \text{ such that } v_{m+1}^j \mapsto \mathcal{I}_{m+1}^m v_{m+1}^j.
\end{aligned}$$

<u>Restriction operator</u>
We will now apply the corresponding matrices to the respective eigenvectors on each level, where we let $\mathcal{I}_m^{m+1} = Z_{m+1}$. We start by taking $m = 0$. Using the basis of eigenvectors for $\mathcal{V}_0$ we have for index j

$$\begin{aligned}
\left[Z_1^T v_0^j\right]_i &= \frac{1}{4}\left(\sin((2i-1)h\pi j) + 2\sin(2ih\pi j) + \sin((2i+1)h\pi j)\right), \\
&= \frac{1}{2}\left(1 + \cos(j\pi h)\right)\sin(2hi\pi j), \\
&= C_{1,h}^j \left[v_1^j\right]_i.
\end{aligned}$$

Now, for the complementary mode on level $m = 0$ corresponding to index j we define $j' = n_0 + 1 - j$. Note that we can write

$$
\begin{aligned}
&[v_0^{j'}]_i = -(-1)^j \sin(ihj\pi), \\
&i = 1, 2, \dots n_m, \text{and } j = 1, 2, \dots n_{m+1}.
\end{aligned}
\tag{5}
$$

Applying the restriction operator to the complementary eigenvector gives

$$
\begin{aligned}
\left[Z_1^T v_0^{j'}\right]_i &= \frac{1}{4}\left(\cos(j\pi h)\sin(2hi\pi j) - (-1)^{2i}\sin(2hi\pi j)\right), \\
&= \frac{1}{4}\left(\cos(j\pi h) - 1\right)\sin(2hi\pi j), \\
&= C_{2,h}^j \left[v_1^j\right]_i.
\end{aligned}
$$

We thus have that at level $m = 1$, the fine-grid eigenvectors from level $m = 0$ are mapped by the restriction operator Z_1^T according to

$$
Z_1^T v_0^j = C_{1,h}^j v_1^j, \ \ j = 1, 2, \dots, n_1, \tag{6}
$$

$$
Z_1^T v_0^{n_0+1-j} = C_{2,h}^j v_1^j, \ \ j = 1, 2, \dots, n_1. \tag{7}
$$

Note that $v_1^j \in V_1 \ \forall j$. Additionally, note that n_1 vectors from $\mathcal{V}_0$ are mapped to zero which implies that the nullspace of Z_1^T has $\dim \mathcal{N}(Z_1^T) = n_1$. In order to move from $m = 1$ to $m = 2$, which maps $\mathcal{V}_1 \to V_2$, we apply Z_2^T. The mapping trajectory is given by the following diagram

$$
\mathcal{I}_1^2 \circ \mathcal{I}_0^1 : \mathcal{V}_0 \xrightarrow{\mathcal{I}_0^1} \mathcal{V}_1 \xrightarrow{\mathcal{I}_1^2} V_2, \ \text{where} \
\begin{tikzcd}
\mathcal{V}_0 \arrow[r, "\mathcal{I}_0^1"] \arrow[dr, "\mathcal{I}_1^2\circ\mathcal{I}_0^1"'] & V_1 \circlearrowleft^{\alpha^1_{\pi(j)}}_{\mathcal{V}_1} \arrow[d, "\mathcal{I}_1^2"] \\
& V_2
\end{tikzcd}
$$

We obtain $\mathcal{V}_0$ by first applying $\alpha^0_{\pi(j)}$ such that we get the ordering of the basis in pairs j, j'. The restriction operator $\mathcal{I}_0^1$ maps these basis vectors to V_1. Then in order to move to the second coarse space V_2, we again have to reorder the basis on V_1 by applying the automorphism $\alpha^1_{\pi(j)}$. After permuting the elements of the basis, we can apply $\mathcal{I}_1^2$. Consequently, the range of $\mathcal{I}_1^2$ is V_2. This is equivalent to having a composition of the linear transformations $\mathcal{I}_1^2 \circ \mathcal{I}_0^1$. Thus, in terms of the matrix representations, applying Z_2^T gives

$$
\begin{aligned}
\left[Z_2^T \left[Z_1^T v_0^j\right]\right]_i &= C_{1,h}^j \left(Z_2^T \left[v_1^j\right]_i\right), \\
&= \frac{1}{2}\left(1 + \cos(j\pi h)\right)\left(Z_2^T \sin(2hi\pi j)\right), \\
&= \frac{1}{2}\left(1 + \cos(j\pi h)\right)\left(\frac{1}{4}\sin((2i-1)2h\pi j) + 2\sin((2i)2h\pi j) + \sin((2i+1)2h\pi j)\right), \\
&= \left(\frac{1}{2}\left(1 + \cos(j\pi h)\right)\right)\left(\frac{1}{2}\left(1 + \cos(j\pi 2h)\right)\right)\sin(4hi\pi j), \\
&= C_{1,h}^j C_{1,2h}^j \left[v_2^j\right]_i.
\end{aligned}
$$

As regards the complementary modes on level $m = 1$ note that $\alpha^1_{\pi(j)} : V_1 \mapsto \mathcal{V}_1$ enables us to redefine $j' = n_1 + 1 - j$, where

$$
\begin{aligned}
&[v_1^{j'}]_i = -(-1)^j \sin(i2hj\pi), \\
&i = 1, 2, \ldots n_1, \text{and } j = 1, 2, \ldots n_2.
\end{aligned}
\tag{8}
$$

Thus, applying the restriction operator to the complementary modes on $m = 1$ gives

$$
\begin{aligned}
\left[Z_2^T\left[Z_1^T v_0^{j'}\right]\right]_i &= C^j_{2,h}\left(Z_2^T\left[v_1^j\right]_i\right), \\
&= \frac{1}{2}\left(\cos(j\pi h) - 1\right)\left(Z_2^T\left[v_1^j\right]_i\right), \\
&= \frac{1}{2}\left(\cos(j\pi h) - 1\right)\left(\frac{1}{4}\left(\cos(j\pi h)\sin(2hi\pi j) - (-1)^{2i}\sin(2hi\pi j)\right)\right), \\
&= \left(\frac{1}{2}\left(\cos(j\pi h) - 1\right)\right)\left(\frac{1}{2}\left(\cos(j\pi 2h) - 1\right)\right)\sin(4hi\pi j), \\
&= C^j_{2,h}C^j_{2,2h}\left[v_2^j\right]_i.
\end{aligned}
$$

Note that $v_2^j \in V_2\ \forall j$. Consequently, using Z_1^T to map from level $m = 0$ to $m = 1$ and Z_2^T to map from level $m = 1$ to $m = 2$, results in the fine-grid eigenvectors being mapped in a nested application according to

$$
\begin{aligned}
Z_2^T\left(Z_1^T v_0^j\right) &= C_1^j v_2^j,\ j = 1, 2, \ldots, n_2, \\
Z_2^T\left(Z_1^T v_0^{n+1-j}\right) &= C_2^j v_2^j,\ j = 1, 2, \ldots, n_2,\ \text{where,} \\
C_1^j &= \left(\frac{1}{2}\right)^m \prod_{l=1}^{m}\left(1 + \cos(j\pi 2^{l-1}h)\right)\ \text{and,} \\
C_2^j &= \left(\frac{1}{2}\right)^m \prod_{l=1}^{m}\left(\cos(j\pi 2^{l-1}h) - 1\right).
\end{aligned}
$$

In this case, n_2 vectors from $\mathcal{V}_1$ are mapped to zero which implies that the nullspace of Z_2^T has $\dim \mathcal{N}(Z_2^T) = n_2$. Consequently, in order to move to $m = 3$ which maps $\mathcal{V}_2 \to V_3$, we can continue applying Z_3^T. From here, it is easy to see that for each subsequent level $m > 2$, consecutive application of the matrices Z_m^T is equivalent to the following linear mapping between the vector spaces $\mathcal{V}_m$

$$
\mathcal{I}^m_{m-1} \circ \mathcal{I}^{m-1}_{m-2} \circ \ldots \circ \mathcal{I}^1_0 : \mathcal{V}_0 \overset{\mathcal{I}^1_0}{\to} \mathcal{V}_1 \overset{\mathcal{I}^2_1}{\to} \mathcal{V}_2 \ldots \mathcal{V}_{m-1} \overset{\mathcal{I}^m_{m-1}}{\to} \mathcal{V}_m,
$$

which can be represented by the following diagram

$$
\begin{tikzcd}
\mathcal{V}_0 \arrow[r, "\mathcal{I}^1_0"] \arrow[drr, "\mathcal{I}^m_{m-1}\circ\mathcal{I}^{m-1}_{m-2}\circ\ldots\circ\mathcal{I}^1_0"'] & V_1 \circlearrowleft^{\alpha^1_\pi}_{\mathcal{V}_1} \arrow[r, "\mathcal{I}^2_1"] & V_2 \circlearrowleft^{\alpha^2_\pi}_{\mathcal{V}_2} \ldots \arrow[r, "\mathcal{I}^{m-1}_{m-2}"] & V_{m-1} \circlearrowleft^{\alpha^{m-1}_\pi}_{\mathcal{V}_{m-1}} \arrow[dl, "\mathcal{I}^m_{m-1}"] \\
 & & V_m &
\end{tikzcd}
$$

We thus have $v_m^j \in V_m\ \forall j$, and in terms of the matrices, we therefore obtain

$$\begin{aligned}
\left[\prod_{l=m}^{1} Z_l^T v_0^j\right]_i &= \left[Z_m^T Z_{m-1}^T \dots \left[Z_2^T \frac{1}{2}\left(1+\cos(j\pi h)\right) v_1\right]\right]_i, \\
&= \left[Z_m^T Z_{m-1}^T \dots \left[Z_3^T \frac{1}{4}\left(1+\cos(j\pi h)\right)\left(1+\cos(j\pi 2h)\right) v_2\right]\right]_i, \\
&= \left(\frac{1}{2}\right)^m \prod_{l=1}^{m} \left(1+\cos(j\pi 2^{l-1}h)\right)[v_m]_i = C_1^j [v_m^j]_i,
\end{aligned}$$

for $j = 1, 2, \dots, n_m$. Similarly, for the complementary part corresponding to $j' = n_{m-1} + 1 - j$ we obtain

$$\left[\prod_{l=m}^{1} Z_l^T v_0^{j'}\right]_i = \left(\frac{1}{2}\right)^m \prod_{l=1}^{m} \left(\cos(j\pi 2^{l-1}h) - 1\right)[v_m]_i = C_2^j [v_m^j]_i.$$

To conclude, we obtain

$$\prod_{l=m}^{1} Z_l^T v_0^j = C_1^j v_m^j,\ \ j = 1, 2, \dots, n_m, \tag{9}$$

$$\prod_{l=m}^{1} Z_l^T v_0^{j'} = C_2^j v_m^j,\ \ j = 1, 2, \dots, n_m. \tag{10}$$

where $C_1^j = \left(\frac{1}{2}\right)^m \prod_{l=1}^m \left(1 + \cos(j\pi 2^{l-1}h)\right)$ and $C_2^j = \left(\frac{1}{2}\right)^m \prod_{l=1}^m \left(\cos(j\pi 2^{l-1}h) - 1\right)$.

Prolongation operator

The restriction operator was defined as the transpose of $\mathcal{I}_{m+1}^m$, and thus we have that the matrix representation of the prolongation operator is given by Z_m. For the prolongation operator, we again start with $m = 1$ and take the basis V_1 as the prolongation operator works on a coarse-grid eigenvector on level m and maps it to a fine-grid counterpart on level $m - 1$. We distinguish two cases; i is odd and i is even. We start with the first case

$$\begin{aligned}
[Z_1 v_1^j]_i &= \frac{1}{4}\left(\sin(\frac{(i-1)2h\pi j}{2}) + \sin(\frac{(i+1)2h\pi j}{2})\right), \\
&= \frac{1}{4}\left(\sin((i-1)h\pi j) + \sin((i+1)h\pi j)\right), \\
&= \frac{1}{2}\cos(j\pi h)\sin(ih\pi j),
\end{aligned} \tag{11}$$

for $j = 1, 2, \dots, n_1$. For i is even, we obtain

$$[Z_1 v_1^j]_i = \frac{1}{2}\sin(\frac{2hi\pi j}{2}) = \frac{1}{2}\sin(hi\pi j) = \frac{1}{2}[v_0^j]_i. \tag{12}$$

Using eq. (8), if we define $j' = n_{m-1} + 1 - j$, we can write eq. (12) as

$$[Z_1 v_1^j]_i = \sin(hi\pi j) = -(-1)^i \sin(j\pi hi) = [v_0^{j'}]_i, \tag{13}$$

for i = odd. Thus, for i is odd, combining eq. (8) and eq. (13), gives

$$[Z_1 v_1^j]_i = \frac{1}{2}[v_0^{j'}]_i + \frac{1}{2}\cos(j\pi h)[v_0^j]_i = C_{1,h}^j[v_0^j, v_0^{j'}]_i,$$

for $j = 1, 2, \ldots, n_1$. Similarly, for i is even, we obtain

$$[Z_1 v_1^j]_i = -\frac{1}{2}[v_0^{j'}]_i + \frac{1}{2}\cos(j\pi h)[v_0^j]_i = C_{2,h}^j[v_0^j, v_0^{j'}]_i,$$

for $j = 1, 2, \ldots, n_1$. Note that $[v_0^j, v_0^{j'}]_i$ is an element of $\mathcal{V}_0$ and the coarse-grid eigenvectors are mapped by the interpolation operator Z_1 according to

$$\mathcal{I}_1^0 : V_1 \xrightarrow{\mathcal{I}_1^0} \mathcal{V}_0.$$

Also note that $\mathcal{R}(Z_1) \subset V_0$, and we have $V_0 = \mathcal{N}(Z_1^T) \bigoplus \mathcal{R}(Z_1)$. We now take $m = 2$, using the basis V_2. From the above, it follows that

$$[Z_2 v_2^j]_i = \frac{1}{2}[v_1^{j'}]_i + \frac{1}{2}\cos(j\pi 2h)[v_1^j]_i = C_{1,2h}^j[v_1^j, v_1^{j'}]_i, \ i \text{ is odd} \tag{14}$$

$$[Z_2 v_2^j]_i = -\frac{1}{2}[v_1^{j'}]_i + \frac{1}{2}\cos(j\pi 2h)[v_1^j]_i = C_{2,2h}^j[v_1^j, v_1^{j'}]_i, \ i \text{ is even}, \tag{15}$$

for $j = 1, 2, \ldots, n_2$ and $j' = n_1 + 1 - 1$. As the v_1^j's are the eigenvectors on level $m = 1$, we can rewrite the complementary indices j' in terms of j again by using

$$\begin{aligned} &[v_1^{j'}]_i = -(-1)^i \sin(i2hj\pi), \\ &i = 1, 2, \ldots n_1, \text{and } j = 1, 2, \ldots n_2. \end{aligned} \tag{16}$$

Substituting eq. (16) into eq. (14) and eq. (15) gives

$$[Z_2 v_2^j]_i = \frac{1}{2}[v_1^j]_i + \frac{1}{2}\cos(j\pi 2h)[v_1^j]_i = C_{1,2h}^j[v_1^j]_i, \ i \text{ is odd} \tag{17}$$

$$[Z_2 v_2^j]_i = -\frac{1}{2}[v_1^j]_i + \frac{1}{2}\cos(j\pi 2h)[v_1^j]_i = C_{2,2h}^j[v_1^j]_i, \ i \text{ is even}, \tag{18}$$

and $\mathcal{R}(Z_2) \subset V_1$, and we have $V_1 = \mathcal{N}(Z_2^T) \bigoplus \mathcal{R}(Z_2)$. Moving from $m = 1$ to $m = 0$ by left-multiplying eq. (17) and eq. (18) with Z_1 is now straightforward as we get the coefficient $C_{1,h}^j$ and $C_{2,h}^j$ times $[Z_1 v_1^j]_i$ from above. This corresponds to a composition of the linear transformations where at $\mathcal{V}_1$ we reorder the basis to V_1 using eq. (16)

$$\mathcal{I}_1^0 \circ \mathcal{I}_2^1 : V_2 \xrightarrow{\mathcal{I}_2^1} \mathcal{V}_1 \xrightarrow{\mathcal{I}_1^0} \mathcal{V}_0, \text{ where } \begin{tikzcd} V_2 \arrow[r, "\mathcal{I}_2^1"] \arrow[dr, "\mathcal{I}_1^0 \circ \mathcal{I}_2^1"'] & \mathcal{V}_1 \arrow[loop right, "V_1"] \arrow[d, "\mathcal{I}_1^0"] \\ & \mathcal{V}_0 \end{tikzcd}.$$

From here it is easy to see that for $m > 2$ successive application gives

$$\begin{aligned}
\left[\prod_{l=1}^{l} Z_l v_m\right]_i &= \left[Z_1 Z_2 \ldots \frac{1}{2}\left(1+\cos(j\pi 2^m h)\right)\left[Z_{m-1} v_{m-1}^j\right]\right]_i, \\
&= \left[Z_1 Z_2 \ldots \frac{1}{4}\left(1+\cos(j\pi 2^m h)\right)\left(1+\cos(j\pi 2^{m-1} h)\right)\left[Z_{m-2} v_{m-2}^j\right]\right]_i, \\
&= \left(\frac{1}{2}\right)^m \prod_{l=m}^{1}\left(1+\cos(j\pi 2^l h)\right)[v_0^j]_i = C_1^j [v_0^j]_i, \text{ for } i \text{ is odd.} \qquad (19)
\end{aligned}$$

Finally, for i is even we get $\left[\prod_{l=1}^{l} Z_l v_m\right]_i = \left(\frac{1}{2}\right)^m \prod_{l=m}^{1}\left(\cos(j\pi 2^l h) - 1\right)[v_0^j]_i = C_2^j [v_0^j]_i$ and $\mathcal{R}(Z_{m+1}) \subset V_m$, and we have $V_m = \mathcal{N}(Z_{m+1}^T) \bigoplus \mathcal{R}(Z_{m+1})$.

Composite mapping subspaces

Let us now take $B_m = \prod_{l=1}^{m-1} Z_l \prod_{l=m-1}^{1} Z_l^T$, and $\hat{B}_m = Z_m Z_m^T$. We furthermore let

$$\begin{aligned}
&{}^t f^m : \mathcal{V}_0 \to V_m : \mathcal{I}_{m-1}^m \circ \mathcal{I}_{m-2}^{m-1} \circ \ldots \circ \mathcal{I}_0^1, \text{ and} \\
&f^m : V_m \to \mathcal{V}_0, \text{ and} \\
&g^m : \mathcal{V}_{m-1} \to \mathcal{V}_{m-1} : \mathcal{I}_m^{m-1} \circ \mathcal{I}_{m-1}^m
\end{aligned}$$

where ${}^t f^m$ is the transpose of the linear map f^m. Note that g^m is a automorphism. We can define

$$h^m : \mathcal{V}_0 \to \mathcal{V}_0 : f^m \circ {}^t f^m, \; f^m \in V_m,$$

to denote the composite linear mapping along the m-vectors spaces. Here ${}^t f^m$ maps elements of $\mathcal{V}_0$ to V_m and we can write $h^m : f^{m-1} \circ (g^m \circ {}^t f^{m-1})$. This gives

$$\begin{aligned}
\ker g^m &= \{v_0^{j'} \in \mathcal{V}_0, : {}^t f^{m-1} v_0^j = 0\} \subset \mathcal{V}_{m-1}, \text{ and} \\
\operatorname{Im} g^m &= \{v_0^j \in \mathcal{V}_0 : {}^t f^{m-1} v_0^j \neq 0\} = V_{m-1}/\ker g^m \subset \mathcal{V}_{m-1},
\end{aligned}$$

where j' are the complementary indices corresponding to $n_0 + 1 - j$. But then by definition and the fact that g^m is an automorphism, ${}^t f^{m-1} v_0^j$ must be an eigenvector of g^m. Given that we can write $V_{m-1} = \ker g^m \bigoplus \operatorname{Im} g^m$, the rank-nullity theorem furthermore tells us that $\dim(V_{m-1}) = \dim(\ker g^m) + \dim(\operatorname{Im} g^m) = n_m + n_m = n_{m-1}$. Thus, g^m must have n_m zero eigenvalues and n_m non-zero eigenvalues as the kernel of g^m is non-trivial. This leads to

$$\begin{aligned}
(g^m \circ {}^t f^{m-1}) v_0^j &= g^m({}^t f^{m-1} v_0^j), \\
&= \lambda(g^m)({}^t f^{m-1} v_0^j) = \lambda(g^m) v_{m-1}^j,
\end{aligned}$$

where $\lambda(g^m)$ denotes the scalar eigenvalue corresponding to g^m. Applying f^{m-1}, finally gives

$$\begin{aligned}
f^{m-1} \circ (g^m \circ {}^t f^{m-1}) v_0^j &= f^{m-1}\left(g^m({}^t f^{m-1})\, v_0^j\right), \\
&= \lambda(g^m) f^{m-1}({}^t f^{m-1} v_0^j) = \lambda(g^m)\lambda(h^{m-1}) v_{m-1}^j.
\end{aligned}$$

Eigendecomposition of B_m

If B_{m-1} and $\hat{B}_m$ are the matrix representations of h^{m-1} and g^m respectively, then $\dim(\ker g^m) = \dim(\mathcal{N}(\hat{B}_m)) = n_m$, and $\dim(\operatorname{Im} g^m) = \dim(\mathcal{R}(\hat{B}_m)) = n_m$, and thus $\hat{B}_m$ has only n_m non-zero eigenvalues. But then B_m must also have n_m non-zero eigenvalues as well. $\square$

We similarly extend the multilevel operators for the higher-order deflation vectors.

Corollary 1.1 (Multilevel Prolongation and Restriction (quadratic))**.** *Let Z_m be the $n_{m-1} \times n_m$ prolongation matrix based on rational Bezier curves for $m = 1, 2, \ldots m_{\max}$, with $n_m = \frac{n}{2^m}$. If we define $v_m^j = \sin(2^m hi\pi j)$, and $v_m^{j'} = \sin(2^m hi\pi(n_m + 1 - j))$, where on the finest level we have $m = 0$. Then there exist constants C_1^j and C_2^j depending on h such that the* <u>*restriction*</u> *operator maps the eigenvectors to*

$$\prod_{l=m}^{1} Z_l^T v_0^j = C_1^j v_m^j, \;\; j = 1, 2, \ldots, n_m,$$

$$\prod_{l=m}^{1} Z_l^T v_0^{j'} = C_2^j v_m^j, \;\; j = 1, 2, \ldots, n_m.$$

where $C_1^j = \left(\frac{1}{2}\right)^m \prod_{l=1}^m C_{1,lh}^j$ and $C_2^j = \left(\frac{1}{2}\right)^m \prod_{l=1}^m C_{2,lh}^j$. Similarly, the <u>*prolongation*</u> *operator maps the eigenvectors to*

$$\prod_{l=1}^{l} Z_l [v_m]_i = C_1^j [v_0^j]_i, \;\; \textit{for } i \textit{ is odd.},$$

$$\prod_{l=1}^{l} Z_l [v_m]_i = C_2^j [v_0^j]_i, \;\; \textit{for } i \textit{ is even.}.$$

Finally, if we let $B_m = \prod_{l=1}^m Z_l \prod_{l=m}^1 Z_l^T$ and $\hat{B}_m = Z_m Z_m^T$ for $m = 1, 2, \ldots, m_{\max}$, then B_m has dimension n_0 with n_m non-zero eigenvalues.

Proof. The proof is exactly the same as the proof of theorem 1, however we now have

$$C_{1,mh}^j = \left(\cos(j\pi 2^m h) + \cos(j\pi 2^{m+1} h)\frac{1}{4} + \frac{3}{4}\right),$$

$$C_{2,mh}^j = \left(\cos(j\pi 2^m h) - \cos(j\pi 2^{m+1} h)\frac{1}{4} - \frac{3}{4}\right).$$

For a detailed proof of deriving $C_{1,mh}^j$ and $C_{2,mh}^j$ see [6]. The statement is obtained by substituting these coefficients into the proof of theorem 1. $\square$

Using this result we can approximate where the near-zero eigenvalues of the coarse-grid matrix E_m will be located. This is expressed in the following corollary.

Corollary 1.2 (Coarse-grid near-zero eigenvalues)**.** *Let Z_m be the $n_{m-1} \times n_m$ prolongation matrix for $m = 0, 1, 2, \ldots m_{\max}$, with $n_m = \frac{n}{2^m}$. We define the symmetric coarse-grid coefficient matrix $E_m = \prod_{l=m}^1 Z_l^T A \prod_{l=1}^m Z_m$. If we let $[v_m^j]_i = \sin(2^m hi\pi j)$ be the eigenvectors of E_m, where for $m = 0$ we have the finest level, then $\exists \tilde{m}$: for $m > \tilde{m}$ E_m is negative definite. For $m \leqslant \tilde{m}$ E_m is indefinite.*

Proof. Let $\Lambda(A)$ denotes the $n_0 \times n_0$ diagonal matrix containing the eigenvalues of A, then using theorem 1 for each i, either odd or even, we have

$$\lim_{h\to 0}\left|E_m[v_m^j]_i\right| \leqslant \lim_{h\to 0}\left|\prod_{l=m}^{1} Z_l^T\Lambda(A)\prod_{l=1}^{m} Z_l[v_m^j]_i\right| \leqslant \lim_{h\to 0}\left|\lambda_A^j(C_1^j)^2[v_m^j]_i\right| \leqslant 4^m\left|\lambda_A^j[v_m^j]_i\right|,$$

where we used that by definition of C_1^j and C_2^j, for all j we have $\left|C_1^jC_2^j\right| \leqslant \left|(C_1^j)^2\right| \leqslant 4^m$. Note that for i is even we would have $C_1^jC_2^j$ instead of ${C_1^j}^2$. Thus, in the limit as h goes to zero, we can bound the expression for $\lambda_{E_m}^j$ from above by $\left|\lambda_{E_m}^j\right| \leqslant 4^m\lambda_A^j$ for each j. Now to find a bound for the smallest eigenvalue in magnitude of E_m, we need to minimize the right-hand side of the upper-inequality over all indices j. This is achieved at $j = j_{\min}$, corresponding to the smallest eigenvalue in magnitude of A as this eigenvalue is the closest eigenvalue to zero. We thus have $\left|\lambda_{E_m}^{j_{\min}}\right| \leqslant 4^m\lambda_A^{j_{\min}}$. We now need to find the level m at which the matrix E_m becomes negative definite. Recall that

$$j_{\min} = \left\lceil \frac{\cos^{-1}\left(\frac{1-k^2h^2}{2}\right)}{\pi h}\right\rceil = \left\lceil \frac{n\cos^{-1}\left(\frac{1-k^2h^2}{2}\right)}{\pi}\right\rceil.$$

Therefore, to find the level $\tilde{m}$ which still contains index $j_{\min}$, for $j = 1, 2, \ldots n_m$, we have to find $m : n_m = \frac{n}{2^m} > j_{\min}$. Note $j_{\min}$ is unaffected by h as h goes to zero and thus we can assess how many times $j_{\min}$ fits into n. Additionally, coarsening leads to the problem size being halved for each m, and thus need to divide by 2 as well.

$$\left\lfloor \frac{n}{2j_{\min}}\right\rfloor = \left\lfloor \frac{\cos^{-1}\left(\frac{1-k^2h^2}{2}\right)}{2\pi}\right\rfloor = \tilde{m}.$$

Consequently, for $m > \tilde{m}$, $j_{\min}$ is no longer within the range of n_m. Therefore, all eigenvalues of $E_{m>\tilde{m}}$ for $j = 1, 2, \ldots n_{m>\tilde{m}} \leqslant j_{\min}$ must have the same sign, due to the fact that $\lambda_A^{j_{\min}}$ is an upperbound and the only eigenvalue of A where a sign-change can occur. □

corollary 1.2 shows that for $m \leqslant \tilde{m}$, the resulting coarse-grid coefficient matrices E_m are indefinite. Thus, on these subsequent levels, it is important that the near-zero eigenvalues are reduced and aligned in coherence with the fine-grid level. In order to analytically assert this, we proceed by defining the multilevel deflation operator and block-diagonalizing it using a similar basis as we used for the two-level ADP scheme. This will allow us to perform spectral analysis of the multilevel deflation operator as the latter reduces to applying the two-level ADP scheme recursively.

4.2 Block-diagonal systems

Using the matrices Z_m and Z_m^T to denote the prolongation and restriction operator on level m, and using the theory developed so far, we can construct similar analytical expressions for the eigenvalues of the preconditioner applied to the coefficient matrix. We will perform the analysis for MP 1-A. We define the $n \times n$ projection operator $P_{h,m}$ to be

$$P_{h,m} = I - AQ_m, \text{ where } Q_m = \prod_{l=1}^{m} Z_l E_m^{-1} \prod_{l=m}^{1} Z_l^T \text{ and } E_0 = A, \tag{20}$$

$$P_m = I_m - E_m Q_m, \text{ where } Q_m = Z_m E_m^{-1} Z_m^T \text{ and } E_m = Z_m^T E_{m-1} Z_m \tag{21}$$

Note that this is equivalent to constructing P by solving E_m directly on the m-th level and then prolonging the inverse back to the fine grid in order to proxy the effect of having an approximate inversion of E_1 in the two-level method. We will refer to $P_{h,m}$ as the **global** multilevel deflation preconditioner and P_m as the **local** level deflation preconditioner.

4.2.1 Global system block-diagonalization

In order to extend the spectral analysis of the two-level ADP-scheme to a multilevel setting, we will use the bases and operators defined in the first part of the proof of theorem 1. We start with the following lemma.

Lemma 2 (Block-diagonalization I). *Let Z_m be the $n_{m-1} \times n_m$ interpolation matrix with $n_m = \frac{n}{2^m}$ for $m = 0, 1, 2, \ldots, m_{\max}$. Let $B_m = \prod_{l=m}^{1} Z_l \prod_{l=1}^{m} Z_l^T$ and $\hat{B}_m = Z_m Z_m^T$ for $m = 1, 2, \ldots, m_{\max}$. Defining the basis*

$$\mathcal{V}_m = \bigoplus_{j=1}^{n_{m+1}} \operatorname{span}\{v_m^j v_m^{n_{m+1}+1-j}\},$$

where $v_m^j = [\sin(j\pi h i 2^m]_{i=1}^{n_m}$, the eigenvalues of B_m are given by

$$\lambda_{B_m}^j = \left(\frac{1}{2}\right)^m \prod_{l=m}^{1} \left((r_l^j)^2 + (p_l^j)^2\right),$$

where $r_l^j = \frac{1}{2}\left(1 + \cos(j\pi 2^{l-1}h)\right)$, $p_l^j = \frac{1}{2}\left(\cos(j\pi 2^{l-1}h) - 1\right)$ for $j = 1, 2, \ldots n_{m-1}$.

Proof. We can start by using the results from theorem 1. To keep the notation compact we let $r_m^j = C_{1,mh}^j = \frac{1}{2}(1 + \cos(j\pi 2^{m-1}h))$ and $p_m^j = C_{2,mh}^j = \frac{1}{2}(\cos(j\pi 2^{m-1}h) - 1)$. We start with the case where $m = 1$. Using the basis $\mathcal{V}_0, V_1$, Z_1 and Z_1^T have the block form

$$[Z_1]_{V_1}^j = \begin{bmatrix} r_1^j \\ p_1^j \end{bmatrix}, \tag{22}$$

$$[Z_1^T]_{\mathcal{V}_0}^j = \begin{bmatrix} r_1^j & p_1^j \end{bmatrix}, \tag{23}$$

for $j = 1, 2, \ldots, n_1$. In block-diagonal form we can write Z_1 as

$$\begin{bmatrix} \boxed{\begin{matrix} r_1^1 \\ p_1^1 \end{matrix}} & & & \mathbf{0} \\ & \boxed{\begin{matrix} r_1^2 \\ p_1^2 \end{matrix}} & & \\ & & \ddots & \\ \mathbf{0} & & & \boxed{\begin{matrix} r_1^{n_1} \\ p_1^{n_1} \end{matrix}} \end{bmatrix}$$

To block-diagonalize $\hat{B}_1$, we therefore multiply the respective blocks for each j

$$[Z_1[Z_1^T]^j_{\mathcal{V}_0}]^j_{V_1} = \begin{bmatrix} r_1^j \\ p_1^j \end{bmatrix} \begin{bmatrix} r_1^j & p_1^j \end{bmatrix} = \begin{bmatrix} (r_1^j)^2 & (r_1^j p_1^j) \\ (r_1^j p_1^j) & (p_1^j)^2 \end{bmatrix}.$$

Now, $\hat{B}_1$ has n_1 non-zero eigenvalues given by the trace of each respective block and n_1 zero eigenvalues, which was also discussed in the proof of theorem 1. The non-zero eigenvalues are thus given by the 1×1 block $\lambda^j_{\hat{B}_1} = (r_1^j)^2 + (p_1^j)^2$ for $j = 1, 2, \ldots, n_1$ and $\hat{B}_1 = B_1$ has the block-diagonal form

$$[B_1]_{\mathcal{V}_0} = \left[\begin{array}{ccc|ccc} \boxed{\lambda^1_{\hat{B}_1}} & & & & & \\ & \ddots & & & \mathbf{0} & \\ & & \boxed{\lambda^{n_1}_{\hat{B}_1}} & & & \\ \hline & & & 0 & & \\ & \mathbf{0} & & & \ddots & \\ & & & & & 0 \end{array}\right].$$

We now take $m = 2$ and block-diagonalize $\hat{B}_2$. Using the same steps as above we have

$$[Z_2 Z_2^T]^j_{\mathcal{V}_1} = \begin{bmatrix} r_2^j \\ p_2^j \end{bmatrix} \begin{bmatrix} r_2^j & p_2^j \end{bmatrix} = \begin{bmatrix} (r_2^j)^2 & (r_2^j p_2^j) \\ (r_2^j p_2^j) & (p_2^j)^2 \end{bmatrix},$$

for $j = 1, 2, \ldots, n_2$. Computing the trace of each block gives $\lambda^j_{\hat{B}_2} = (r_2^j)^2 + (p_2^j)^2$ with block-diagonal form

$$[\Lambda(\hat{B}_2)]_{\mathcal{V}_1} = \left[\begin{array}{ccc|ccc} \boxed{\lambda^1_{\hat{B}_2}} & & & & & \\ & \ddots & & & \mathbf{0} & \\ & & \boxed{\lambda^{n_2}_{\hat{B}_2}} & & & \\ \hline & & & 0 & & \\ & \mathbf{0} & & & \ddots & \\ & & & & & 0 \end{array}\right]. \tag{24}$$

Note that we have $n_2 = \frac{n}{4}$ zero and non-zero eigenvalues and the dimension of $\hat{B}_2$ is $n_1 \times n_1$. This is equivalent to having n_2 blocks of dimension 1×1 containing the non-zero eigenvalues and n_2 blocks, also with dimension 1×1 containing the zero eigenvalues. We now apply Z_1 to the left and Z_1^T to the right of eq. (24), where we use the block-diagonal form of Z_1 and Z_1^T given by eq. (22) and eq. (23) respectively. Z_1 has n_1 blocks of dimension 2×1 and Z_1^T has n_1 blocks of dimension 1×2. Thus, Z_1 works on each non-zero 1×1 block of $\hat{B}_2$, and then Z_1^T is applied to the resulting 2×1 block. However, only the first n_2 blocks of $\Lambda(\hat{B}_2)$ contain non-zero terms as we can see from eq. (24) and thus only the indices $j = 1, 2, \dots n_2$ in Z_1 and Z_1^T lead to non-zero terms. Thus, for $j = 1, 2, \dots, n_2$ we obtain $[\Lambda(B_2)]_{\mathcal{V}_0} = [\Lambda(Z_1 \hat{B}_2 Z_1^T)]_{\mathcal{V}_0}$, which is given by the following matrix representation

$$
\begin{bmatrix}
\boxed{\begin{matrix} r_1^1 \\ p_1^1 \end{matrix}} & & & \mathbf{0} \\
& \boxed{\begin{matrix} r_1^2 \\ p_1^2 \end{matrix}} & & \\
& & \ddots & \\
\mathbf{0} & & & \boxed{\begin{matrix} r_1^{n_1} \\ p_1^{n_1} \end{matrix}}
\end{bmatrix}
\left[\begin{array}{ccc|ccc}
\boxed{\lambda^1_{\hat{B}_2}} & & & & & \\
& \ddots & & & \mathbf{0} & \\
& & \boxed{\lambda^{n_2}_{\hat{B}_2}} & & & \\
\hline
& & & 0 & & \\
& \mathbf{0} & & & \ddots & \\
& & & & & 0
\end{array}\right]
\begin{bmatrix}
\boxed{r_1^1 \quad p_1^1} & & & \mathbf{0} \\
& \boxed{r_1^2 \quad p_1^2} & & \\
& & \ddots & \\
\mathbf{0} & & & \boxed{r_1^{n_1} \quad p_1^{n_1}}
\end{bmatrix}
$$

Thus, at the level of each respective j-th block we have

$$
[\Lambda(B_2)]^j_{\mathcal{V}_0} = \begin{bmatrix} r_1^j \\ p_1^j \end{bmatrix} \lambda^j_{\hat{B}_2} \begin{bmatrix} r_1^j & p_1^j \end{bmatrix} = \lambda^j_{\hat{B}_2} \begin{bmatrix} (r_1^j)^2 & (r_1^j p_1^j) \\ (r_1^j p_1^j) & (p_2^j)^2 \end{bmatrix},
$$

for $j = 1, 2, \dots, n_2$. Computing the trace of each respective block gives

$$
\lambda^j_{B_2} = \left((r_1^j)^2 + (p_1^j)^2\right)(\lambda^j_{\hat{B}_2}) = \left((r_1^j)^2 + (p_1^j)^2\right)\left((r_2^j)^2 + (p_2^j)^2\right). \tag{25}
$$

Thus, we obtain the following block-diagonal form

$$
[B_2]_{\mathcal{V}_0} = \left[\begin{array}{ccc|ccc}
\boxed{\lambda^1_{B_2}} & & & & & \\
& \ddots & & & \mathbf{0} & \\
& & \boxed{\lambda^{n_2}_{B_2}} & & & \\
\hline
& & & 0 & & \\
& \mathbf{0} & & & \ddots & \\
& & & & & 0
\end{array}\right],
$$

where $\lambda^j_{B_2}$ is given by eq. (25). From here it is easy to see that successive application of Z_m and Z_m^T for $m > 2$ gives

$$
[\Lambda(B_m)]^j_{\mathcal{V}_0} = \begin{bmatrix} \prod_{l=m-1}^{1} r_l^j \\ \prod_{l=m-1}^{1} p_l^j \end{bmatrix} \lambda^j_{\hat{B}_m} \begin{bmatrix} \prod_{l=m-1}^{1} r_l^j & \prod_{l=m-1}^{1} p_l^j \end{bmatrix},
$$

for $j = 1, 2, \dots, n_m$ with $\lambda^j_{B_m} = \prod_{l=m}^{1} \left((r_l^j)^2 + (p_l^j)^2\right)$. $\square$

Using the results from theorem 2, we can start by block-diagonalizing the Galerkin coarse-grid operator E_m, where m again denotes the level. On the basis $\mathcal{V}_0$ defined with respect to the finest level $m = 0$, we can block-diagonalize the coefficient matrix A in terms of a total of n_1 blocks with size 2×2. If we define the complementary index $j' = n_m + 1 - j = n_0 + 1 - j$, then each j-th respective block has the form

$$[\Lambda(A)]^j_{\mathcal{V}_0} = \begin{bmatrix} \lambda^j_A & 0 \\ 0 & \lambda^{j'}_A \end{bmatrix},$$

for $j = 1, 2, \dots, n_1$. Moving to $m = 1$, we now start using $\mathcal{V}_1$ as E_1 resides in the coarse-space. After applying Z_1^T and Z_1, we obtain, for $j = 1, 2, \dots n_1$, the 1×1 block

$$\begin{aligned} [\Lambda(E_1)]^j_{\mathcal{V}_1} &= \left[Z_1^T A_0 Z_1\right]^j_{\mathcal{V}_1}, \\ &= \begin{bmatrix} r^j_1 & p^j_1 \end{bmatrix} \begin{bmatrix} \lambda^j_A & 0 \\ 0 & \lambda^{j'}_A \end{bmatrix} \begin{bmatrix} r^j_1 \\ p^j_1 \end{bmatrix}, \\ &= \left(r^j_1\right)^2 \lambda^j_A + \left(p^j_1\right)^2 \lambda^{j'}_A. \end{aligned}$$

Thus, if we define $\lambda^j_{E_1} = \left(r^j_1\right)^2 \lambda^j_A + \left(p^j_1\right)^2 \lambda^{j'}_A$ for $j = 1, 2, \dots, n_1$, then E_1 has block-diagonal form

$$[\Lambda(E_1)]_{V_1} = \begin{bmatrix} \boxed{\lambda^1_{E_1}} & & & \mathbf{0} \\ & \boxed{\lambda^2_{E_1}} & & \\ & & \ddots & \\ \mathbf{0} & & & \boxed{\lambda^{n_1}_{E_1}} \end{bmatrix}.$$

Note that E_1 has no zero eigenvalues and dimension $n_1 \times n_1$. Consequently, we have a total of n_1 blocks with size 1×1 corresponding to each index j at level $m = 1$. To apply Z_2^T and Z_2 to E_1, we now need the 2×2 blocks. We can apply the permutation matrix corresponding to α_π with respect to V_1 such that we get the ordered basis $\mathcal{V}_1$. On this basis the block-diagonal form of E_1 is form

$$[\Lambda(E_1)]_{\mathcal{V}_1} = \begin{bmatrix} \boxed{\begin{matrix} \lambda^1_{E_1} & 0 \\ 0 & \lambda^{1'}_{E_1} \end{matrix}} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \boxed{\begin{matrix} \lambda^{n_2}_{E_1} & 0 \\ 0 & \lambda^{n_2'}_{E_1} \end{matrix}} \end{bmatrix}.$$

for $j = 1, 2, \dots n_2$. Now, applying the block-diagonal form of Z_2^T and Z_2 to $[\Lambda(E_1)]_{\mathcal{V}_1}$

gives

$$\begin{bmatrix} \boxed{r_2^1 \;\; p_2^1} & & & \mathbf{0} \\ & \boxed{r_2^2 \;\; p_2^2} & & \\ & & \ddots & \\ \mathbf{0} & & & \boxed{r_2^{n_2} \;\; p_2^{n_2}} \end{bmatrix} \begin{bmatrix} \boxed{\begin{matrix} \lambda_{E_1}^1 & 0 \\ 0 & \lambda_{E_1}^{1'} \end{matrix}} & & & \mathbf{0} \\ & \boxed{r_2^2 \;\; p_2^2} & & \\ & & \ddots & \\ \mathbf{0} & & & \boxed{\begin{matrix} \lambda_{E_1}^{n_2} & 0 \\ 0 & \lambda_{E_1}^{n_2'} \end{matrix}} \end{bmatrix} \begin{bmatrix} \boxed{\begin{matrix} r_2^1 \\ p_2^1 \end{matrix}} & & & \mathbf{0} \\ & \boxed{\begin{matrix} r_2^2 \\ p_2^2 \end{matrix}} & & \\ & & \ddots & \\ \mathbf{0} & & & \boxed{\begin{matrix} r_2^{n_2} \\ p_2^{n_2} \end{matrix}} \end{bmatrix}.$$

Note that $[\Lambda(E_1)]_{\mathcal{V}_1}$ has size $(n_1 \times n_1)$ and Z_2^T has size $(n_2 \times n_1)$. Thus, for $j = 1, 2, \ldots, n_2$ and $j' = n_1 + 1 - j$, each respective j-th block leads to the (1×1) block containing

$$[\Lambda(E_2)]_{\mathcal{V}_1}^j = \begin{bmatrix} r_2^j & p_2^j \end{bmatrix} \begin{bmatrix} \lambda_{E_1}^j & 0 \\ 0 & \lambda_{E_1}^{j'} \end{bmatrix} \begin{bmatrix} r_2^j \\ p_2^j \end{bmatrix} = (r_2^j)^2 \lambda_{E_1}^j + (p_2^j)^2 \lambda_{E_1}^{j'}.$$

From here it is easy to see that for $m > 2$, application of Z_m^T and Z_m recursively gives a $j-$th (1×1) block with $\lambda_{E_m}^j = (r_m^j)^2 \lambda_{E_{m-1}}^j + (p_m^j)^2 \lambda_{E_{m-1}}^j$ for $j = 1, 2, \ldots, n_m$ and $j' = n_{m-1} + 1 - j$, where each j-th block has the form

$$[\Lambda(E_m)]_{\mathcal{V}_m}^j = \begin{bmatrix} \lambda_{E_m}^j & 0 \\ 0 & \lambda_{E_m}^{j'} \end{bmatrix}.$$

We can now combine theorem 2 and the previous expression for the eigenvalues of E_m to block-diagonalize Q_m. We can now use the result from theorem 2. This gives

$$[\Lambda(Q_m)]_{\mathcal{V}_0}^j = [\Lambda(\prod_{l=1}^{m} Z_l E_m^{-1} \prod_{l=m}^{1} Z_l^T)]_{\mathcal{V}_0}^j = \lambda_{E_m{}^j}^{-1} [\Lambda(B_m)]_{\mathcal{V}_0}^j = \lambda_{E_m{}^j}^{-1} \prod_{l=m}^{1} \left((r_l^j)^2 + (p_l^j)^2 \right),$$

for $j = 1, 2, \ldots, n_m$. We can now easily block-diagonalize P_m as follows

$$\begin{aligned} [\Lambda(P_m)]_{\mathcal{V}_0}^j &= [I - AQ_m]_{\mathcal{V}_0}^j, \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}_{\mathcal{V}_0}^j - \frac{\lambda_{B_m}^j}{\lambda_{E_m}^j} \begin{bmatrix} \lambda_A^j & 0 \\ 0 & \lambda_A^{j'} \end{bmatrix}_{\mathcal{V}_0}^j, \\ &= \begin{bmatrix} 1 - \frac{\lambda_A^j \lambda_{B_m}^j}{\lambda_{E_m}^j} & \frac{\lambda_A^j \lambda_{B_m}^j}{\lambda_{E_m}^j} \\ \frac{\lambda_A^{j'} \lambda_{B_m}^j}{\lambda_{E_m}^j} & 1 - \frac{\lambda_A^{j'} \lambda_{B_m}^j}{\lambda_{E_m}^j} \end{bmatrix}_{\mathcal{V}_0}^j \end{aligned}$$

Including the CSLP-preconditioner M^{-1} and applying the multilevel-deflation preconditioner P_m to the coefficient matrix A finally gives the block-diagonal expressions of the preconditioned system

$$[\Lambda(P_m M^{-1} A)]_{\mathcal{V}_0}^j = \frac{\lambda_A^j}{\lambda_M^j} \begin{bmatrix} 1 - \frac{\lambda_A^j \lambda_{B_m}^j}{\lambda_{E_m}^j} & \frac{\lambda_A^j \lambda_{B_m}^j}{\lambda_{E_m}^j} \\ \frac{\lambda_A^{j'} \lambda_{B_m}^j}{\lambda_{E_m}^j} & 1 - \frac{\lambda_A^{j'} \lambda_{B_m}^j}{\lambda_{E_m}^j} \end{bmatrix}_{\mathcal{V}_0}^j.$$

At last, we obtain the eigenvalues of $P_mM^{-1}A$ for $j = 1, 2, \ldots, n_1$ and $j' = n_0 + 1 - j$, by computing the trace of each respective block

$$\lambda^j(P_mM^{-1}A) = \frac{\lambda_A^j}{\lambda_M^j}\left(1 - \frac{\lambda_A^j\lambda_{B_m}^j}{\lambda_{E_m}^j}\right) + \frac{\lambda_A^{j'}}{\lambda_M^j}\left(1 - \frac{\lambda_A^{j'}\lambda_{B_m}^j}{\lambda_{E_m}^j}\right), \tag{26}$$

with $\lambda_{B_m}^j = \prod_{l=m}^{1}\left((r_l^j)^2 + (p_l^j)^2\right)$.

4.3 Spectral analysis

Using these expressions, we proceed by analyzing the various operators involved in the multi-level deflation operator.

4.3.1 Global near-zero eigenvalues

We start with $P_{h,m}$ and the spectrum of the operators up to the level where the coefficient matrix becomes negative definite, which according to corollary 1.2 is at $\tilde{m} = 3$. For $k = 100$ and MP 1-A, we define $P_{h,1}, P_{h,2}$ and $P_{h,3}$ according to eq. (20). We keep the shift $\beta_2 = 1$ for this part of the analysis. fig. 1 contains the results using linear interpolation, whereas fig. 2 illustrates the spectrum when the deflation space is constructed using higer-order deflation vectors. We observe that using linear interpolation, already on the first level (thus moving from n to $\frac{n}{2}$), near-zero eigenvalues start to appear. This is in fact the DEF-TL operator. As we move to the second level (from $\frac{n}{2}$ to $\frac{n}{4}$), the number of near-zero eigenvalues increases. Note that the at the third level (from $\frac{n}{4}$ to $\frac{n}{8}$), the spectrum completely resembles the spectrum obtained from solely applying the CSLP-preconditioner. We have proved that starting from the third level, the resulting coarse-grid coefficient matrix E_3 is completely negative definite. Consequently, the problem of the near-zero eigenvalues of $E_{m>3}$ resolves itself at these levels given that the location of the smallest eigenvalue in terms of magnitude is now fixed away from zero due to the matrix being negative-definite. Moreover, the further down the levels we move, the smaller the number of eigenvalues become which get projected away.

We repeat the analysis for $k = 1000$, where fig. 3 is based on linear interpolation and fig. 4 on higher-order deflation vectors. We observe the same effect; when the coefficient matrix remains indefinite, the eigenvalues of the first and second level deflation-preconditioner approach the origin as the wavenumber k increases for the linear interpolation scheme. While we notice some near-zero eigenvalues appear for the second level $P_{h,2}$ preconditioner, the spectrum of the first level preconditioner $P_{h,1}$ remains away from the origin when using the higher-order deflation scheme. This is in line with the spectral analysis of the two-level ADP-scheme.

Spectrum of the global deflation + CSLP preconditioned system.

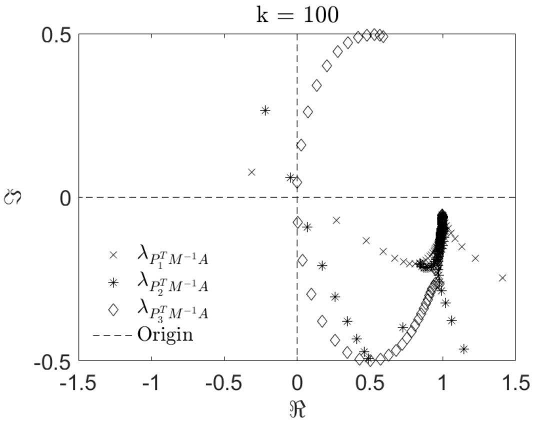


Figure 1: Linear Interpolation

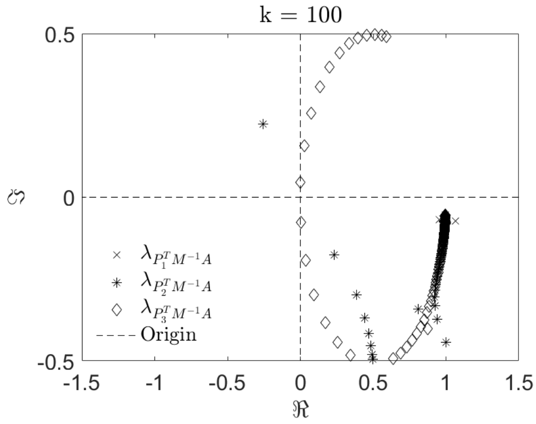


Figure 2: Quadratic Rational Bezier

Spectrum of the global deflation + CSLP preconditioned system.

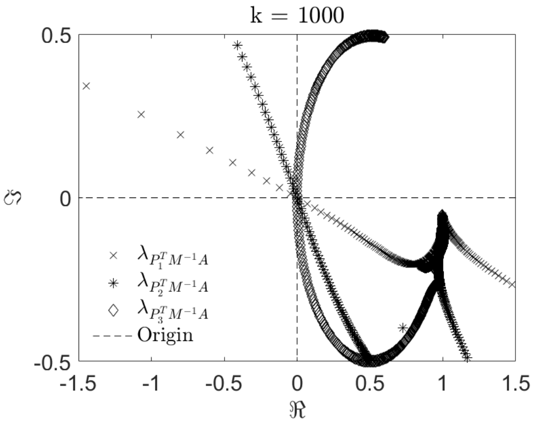


Figure 3: Linear Interpolation

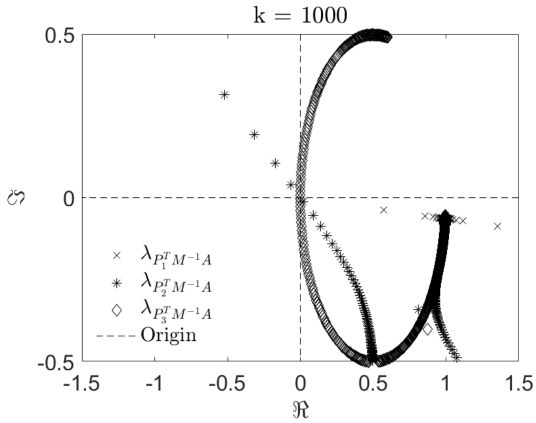


Figure 4: Quadratic Rational Bezier

4.3.2 Local deflated near-zero eigenvalues

Here we start by plotting the local near-zero eigenvalues for $k = 100$ of P_2 and P_3 and compare them to $P_{h,2}$ and $P_{h,3}$ respectively. We start with the linear interpolation scheme in fig. 5 and fig. 6. We observe that the eigenvalues of the local and global operator are similar. If we use a higher-order scheme the largest gain in terms of removing the near-zero eigenvalues is realized at level $m \leqslant 2$. At these levels, comparing fig. 7 to fig. 5, we observe that we have no near-zero eigenvalues both globally and locally. As soon as the matrix becomes negative definite, the spectrum is fully determined by the spectrum of CSLP applied to the global and/or local coefficient matrix.

Spectrum of global and local deflation + CSLP preconditioned system using linear interpolation.

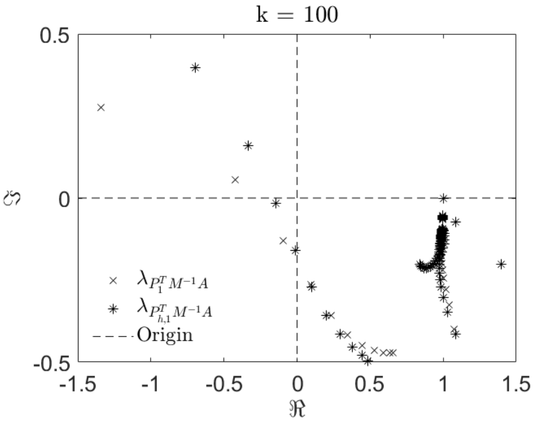


Figure 5: Level $m = 2$

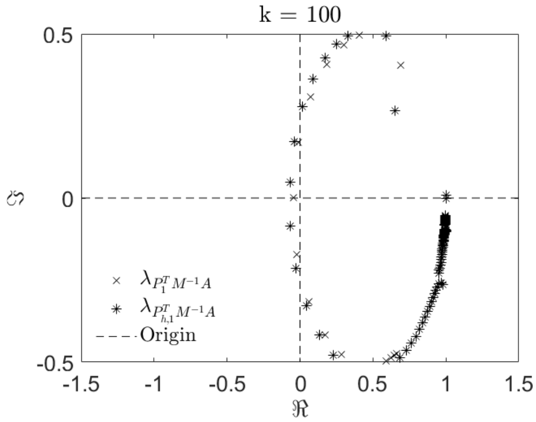


Figure 6: Level $m = 3$

Spectrum of global and local deflation + CSLP preconditioned system using quadratic rational Bezier interpolation.

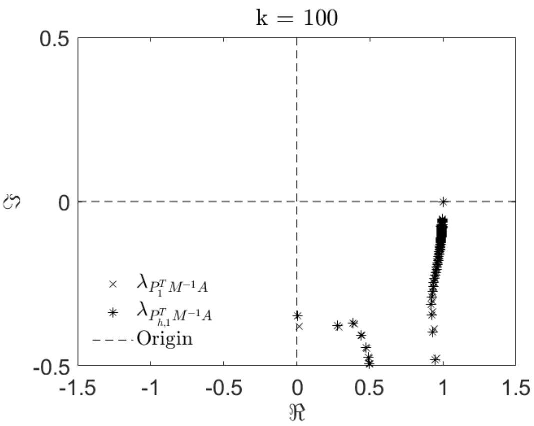


Figure 7: Level $m = 2$

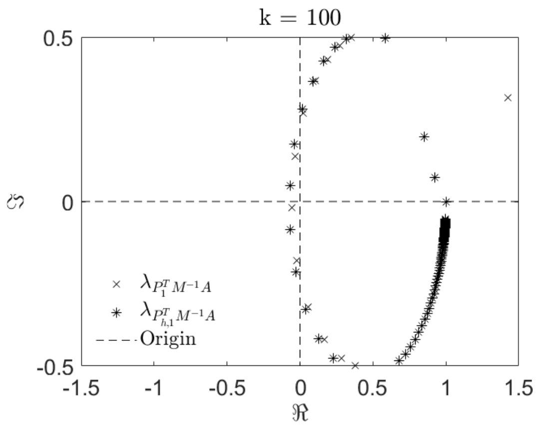


Figure 8: Level $m = 3$

4.3.3 Local near-zero eigenvalues

Here we proceed by plotting the eigenvalues of the coarse-grid systems for levels $m \leqslant 3$. The results are comparable to the ones obtained for the two-level ADP preconditioner. The near-zero eigenvalues for all levels where the coefficient matrices are indefinite remain aligned, see fig. 10. Comparing this to fig. 9 for the linear interpolation case, the near-zero eigenvalues start shifting as we move from $m = 0$ to $m = 2$. Note that at $m = 3$ all eigenvalues are negative, which follows from corollary 1.2.

Spectrum of the coarse linear systems for $k = 100$ and $m \leqslant 3$.

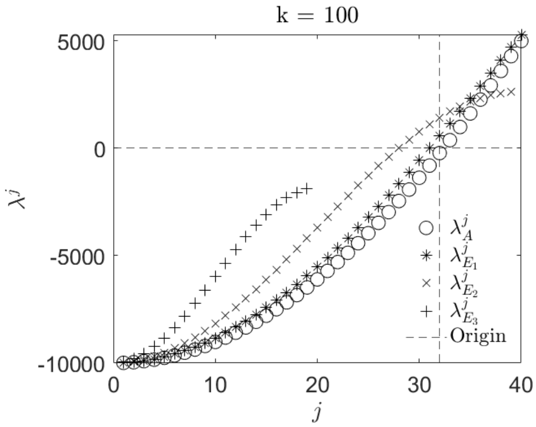


Figure 9: Linear Interpolation

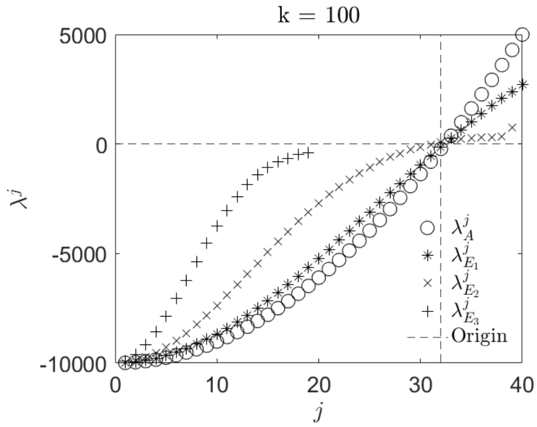


Figure 10: Quadratic Rational Bezier

5 Numerical Experiments

In this section we will provide numerical experiments to study the convergence behavior of our multilevel preconditioner. One advantage we have, is that we are able to use a small shift within the CSLP-preconditioner given that we have substituted the full multigrid cycle on each level with a few GMRES-iterations. Thus, we will use $\beta_2 = \frac{1}{k}$ as it has been shown to provide optimal convergence [13]. This is especially beneficial as the spectral analysis from section 4.3 has shown that at subsequent levels in the chain, the spectrum is predominantly determined by the spectrum of the local CSLP-preconditioned linear system. The tolerance level for the relative residual of the outer method has been set to 10^{-7} and the grid resolution kh is kept at $kh = 0.625$ unless stated otherwise. Note that in general, kh is set to $k^3h^2 < 1$ in order to minimize the pollution error. We have shown in [6] that this is equivalent to letting h go to zero, which for our method lowers the number of iterations. Thus, if the method performs well for a larger h, then it will perform even better for a smaller h. However, the coefficient matrices become very large, especially for MP-3, which is why we keep $kh = 0.625$. For the inner GMRES-iterations we have a tolerance level of 10^{-1}, but we do not require convergence as we are interested in a low accuracy approximation of the CSLP-preconditioner. In all cases unless stated otherwise, we additionally set the maximum number of iterations $n^{\frac{1}{8}}$. All experiments are implemented sequentially on a Dell laptop using 8GB RAM and a i7-8665U processor. For the one- and two-dimensional model problem respectively, an exact solve is performed at the coarsest level with problem size $n = 10$. For the three-dimensional model problem we perform an exact solve when $n = 100$. Moreover, we only allow one FGMRES-iteration on each level.

5.1 One-dimensional Constant Wave Number Model

For MP 1-A and MP 1-B the results are presented in table 1, which contains the number of FGMRES-iterations. The results from table 1 show that the multilevel deflation approach exhibits similar behavior compared to the two-level ADP-scheme as regards wave number independent convergence. In the classical multilevel-Krylov setting, the CSLP-preconditioner is applied by allowing one multigrid cycle. One drawback of this approach is the requisite to choose the shift β_2 large [4]. Thus, we have replaced the approximation of the preconditioner with a few restarted GMRES iterations preconditioned by the diagonal of the CSLP-preconditioner using a tolerance of 10^{-1}. We observe that the use of the higher-order deflation vectors, enables us to obtain scalable convergence given that the number of iterations remains fairly constant, despite allowing for a low-accuracy approximation of the preconditioner. These results furthermore illustrate the theory from section 4 and are coherent with the spectral analysis from section 4.3. For the levels where the coefficient matrices remain indefinite, the deflation preconditioner maps the near-zero eigenvalues to the origin and keeps the subsequent near-zero eigenvalues aligned (see fig. 1 to fig. 4). Once the coefficient matrices become negative definite, the remaining smallest eigenvalues for subsequent levels will be located at the same index, which makes the resulting system similar to the CSLP-preconditioned system. As a result, keeping the shift β_2 small according to [13] allows us to tackle these eigenvalues by using a few GMRES-iterations to approximate the CSLP preconditioner.

Table 1: Number of outer FGMRES-iterations for MP 1-A and MP 1-B using $kh = 0.625$. $\odot$ indicates that the number of iterations has exceeded 125.

	n	MP 1-A		MP 1-B	
k		ADP-ML	DEF-ML	ADP-ML	DEF-ML
100	160	16	19	16	19
250	400	16	27	16	23
500	800	16	36	16	31
1000	1.600	16	67	16	56
5000	8.000	17	$\odot$	16	$\odot$
10000	16.000	19	$\odot$	16	$\odot$

5.2 Two-dimensional Constant Wave Number Model

table 2 contains the results for MP-2. These results are again similar to the two-level variant in the sense that the number of iterations remains constant, even for large k.

We note that DEF-ML already exceeds the maximum number of iterations (125) after $k > 250$. Similar results were reported and observed in [26].

Table 2: Number of outer FGMRES-iterations for MP 2 using $kh = 0.625$. $\odot$ indicates that the number of iterations has exceeded 125. $\oslash$ indicates memory has been exceeded.

	$kh = 0.625$			$kh = 0.3125$		
k	n	ADP-ML	DEF-ML	n	ADP-ML	DEF-ML
50	6.241	18	34	25.281	14	37
100	25.281	18	41	368.449	14	39
250	159.201	18	53	638401	14	48
500	638.401	18	$\odot$	2.556.801	14	$\odot$
750	1.437.601	18	$\odot$	5.755.201	$\oslash$	$\odot$
1000	2.556.801	18	$\odot$	10.233.601	$\oslash$	$\odot$

5.3 Two-dimensional Non-constant Marmousi Model

For the industrial Marmousi problem (MP-4), results are reported in table 3. These results

Table 3: Number of outer FGMRES-iterations for the Marmousi problem MP-4, where f denotes the frequency in Hertz.

f	ADP-ML	DEF-ML
	Iterations	
1	12	10
10	12	15
20	12	20
40	12	33

again resemble the results from the two-level method; we obtain a constant number of iterations irrespective of the frequency. Thus, even for a varying wavenumber through a heterogeneous media we are able to obtain wavenumber independent convergence using ADP-ML.

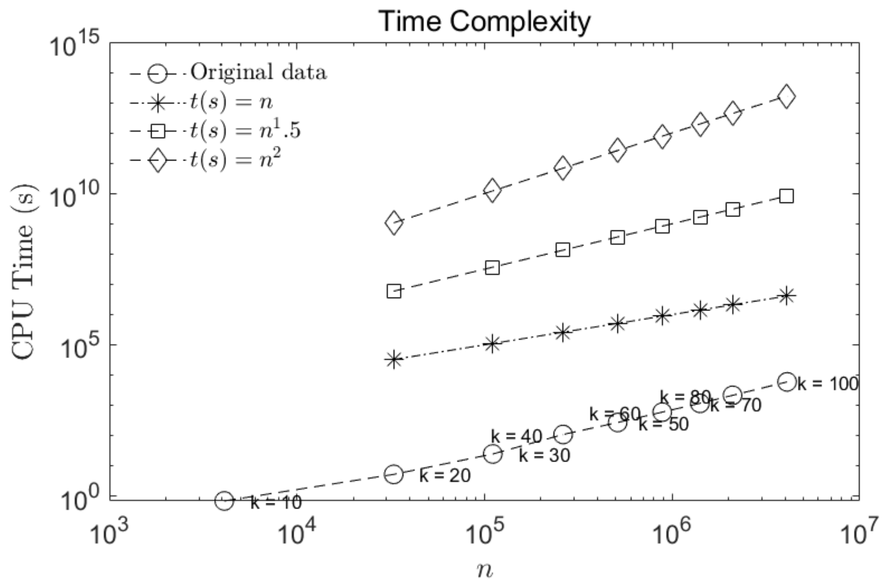
5.4 Three-dimensional Constant Wave Number Model

Here we only report the ADP-ML results as we have seen from the two-dimensional numerical experiments that the number of iterations for the DEF-ML preconditioner already increases for medium-high wavenumbers. Due to our sequential implementation, we are only able to test up to $k = 100$. For the inner GMRES-iterations we use a tolerance of 10^{-1} and a maximum of 7 iterations for all k. Note that we do not require convergence, as the inner GMRES-iterations serve as a proxy for the inverse of the CSLP. The convergence results for MP-3 are presented in table 4. While we observe a slight increase in the number of iterations for MP-3, the convergence appears more or less wave number independent.The time complexity can be analyzed from fig. 11, where timings are recorded using the tic toc command from Matlab.

Table 4: Number of outer FGMRES-iterations for MP-3 using $kh = 0.625$.

k	n	ADP-ML Iterations
10	4096	10
20	32.768	11
40	262.144	11
60	884.736	11
80	2.097.152	12
100	4.096.000	12

Figure 11: CPU time in seconds (s) versus problem size n for MP-3. The wave number corresponding to the problem size has been reported next to the marker. For comparison up to quadratic complexity reference lines are also given.



Unlike the two-level ADP-preconditioner, the current multilevel preconditioner no longer requires the direct solve on the second level. Consequently, the method becomes significantly more efficient both in terms of memory and computational resource usage. In fact, the combination of having a fixed and bounded number of (F)GMRES-iterations and a cheap inner direct solve of size $n = 100$ provides the theoretical potential for an $\mathcal{O}(n)$ solver. Our Matlab implementation is not yet optimized so the slope in the timings plot does not reflect the linear time complexity.

6 Conclusion

In this work we extend the two-level deflation preconditioner using higher-order deflation vectors to a multilevel deflation preconditioner [6]. We provide theoretical and numerical evidence to show that up to a certain level, the coefficient matrices are indefinite. These levels are of paramount importance as the near-zero eigenvalues at these level can effectively be removed by the multilevel deflation preconditioner. If the near-zero eigenvalues are aligned, then the eigenvalues cluster near the point $(1, 0)$ in the complex plane, accelerating the convergence of the underlying Krylov solver.
After this level, the subsequent coarse coefficient matrices become negative definite and its spectrum resembles the spectrum of the CSLP-preconditioned system. Thus, we implement a small number of inner GMRES-iterations to approximate the CSLP using the inverse of the wave number k as the shift ($\beta_2 = k^{-1}$). This circumvents the difficulty of multigrid approximations, where the shift β_2 has to be kept large. The proposed configuration leads to scalable results as we obtain close to wave number independent convergence in terms of a fixed number of iterations. It furthermore, extends the results for both a constant and non-constant wave number model problem, such as the industrial Marmousi model problem. Additionally, sequential implementation of the method leads to scalable timing results for the three-dimensional model problem.
Depending on the implementation, we expect the practical time complexity to be somewhere between $\mathcal{O}(n)$ and $\mathcal{O}(n^{1.5})$.

References

[1] M. Bonazzoli, V. Dolean, I. G. Graham, E. A. Spence, and P.-H. Tournier, *Two-level preconditioners for the helmholtz equation*, in International Conference on Domain Decomposition Methods, Springer, 2017, pp. 139–147.

[2] H. Chen, P. Lu, and X. Xu, *A robust multilevel method for hybridizable discontinuous galerkin method for the helmholtz equation*, Journal of Computational Physics, 264 (2014), pp. 133–151.

[3] H. Chen, H. Wu, and X. Xu, *Multilevel preconditioner with stable coarse grid corrections for the helmholtz equation*, SIAM Journal on Scientific Computing, 37 (2015), pp. A221–A244.

[4] P.-H. Cocquet and M. J. Gander, *How large a shift is needed in the shifted helmholtz preconditioner for its effective inversion by multigrid?*, SIAM Journal on Scientific Computing, 39 (2017), pp. A438–A478.

[5] L. Conen, V. Dolean, R. Krause, and F. Nataf, *A coarse space for heterogeneous helmholtz problems based on the dirichlet-to-neumann operator*, Journal of Computational and Applied Mathematics, 271 (2014), pp. 83–99.

[6] V. Dwarka and C. Vuik, *Scalable convergence using two-level deflation preconditioning for the helmholtz equation*, SIAM Journal on Scientific Computing, 42 (2020), pp. A901–A928.

[7] H. C. Elman, O. G. Ernst, and D. P. O'leary, *A multigrid method enhanced by krylov subspace iteration for discrete helmholtz equations*, SIAM Journal on scientific computing, 23 (2001), pp. 1291–1315.

[8] Y. A. Erlangga and R. Nabben, *On a multilevel krylov method for the helmholtz equation preconditioned by shifted laplacian*, Electronic Transactions on Numerical Analysis, 31 (2008), p. 3.

[9] Y. A. Erlangga, C. W. Oosterlee, and C. Vuik, *A novel multigrid based preconditioner for heterogeneous helmholtz problems*, SIAM Journal on Scientific Computing, 27 (2006), pp. 1471–1492.

[10] Y. A. Erlangga, C. Vuik, and C. W. Oosterlee, *On a class of preconditioners for solving the helmholtz equation*, Applied Numerical Mathematics, 50 (2004), pp. 409–425.

[11] O. G. Ernst and M. J. Gander, *Multigrid methods for helmholtz problems: A convergent scheme in 1d using standard components.*

[12] O. G. Ernst and M. J. Gander, *Why it is difficult to solve helmholtz problems with classical iterative methods*, in Numerical analysis of multiscale problems, Springer, 2012, pp. 325–363.

[13] M. J. Gander, I. G. Graham, and E. A. Spence, *Applying gmres to the helmholtz equation with shifted laplacian preconditioning: what is the largest shift for which wavenumber-independent convergence is guaranteed?*, Numerische Mathematik, 131 (2015), pp. 567–614.

[14] M. J. Gander, F. Magoules, and F. Nataf, *Optimized schwarz methods without overlap for the helmholtz equation*, SIAM Journal on Scientific Computing, 24 (2002), pp. 38–60.

[15] M. J. Gander and H. Zhang, *Restrictions on the use of sweeping type preconditioners for helmholtz problems*, in International Conference on Domain Decomposition Methods, Springer, 2017, pp. 321–332.

[16] M. J. Gander and H. Zhang, *A class of iterative solvers for the helmholtz equation: Factorizations, sweeping preconditioners, source transfer, single layer potentials, polarized traces, and optimized schwarz methods*, Siam Review, 61 (2019), pp. 3–76.

[17] I. Graham, E. Spence, and E. Vainikko, *Domain decomposition preconditioning for high-frequency helmholtz problems with absorption*, Mathematics of Computation, 86 (2017), pp. 2089–2127.

[18] I. Graham, E. Spence, and J. Zou, *Domain decomposition with local impedance conditions for the helmholtz equation*, arXiv preprint arXiv:1806.03731, (2018).

[19] I. G. Graham, E. A. Spence, and E. Vainikko, *Recent results on domain decomposition preconditioning for the high-frequency helmholtz equation using absorption*, in Modern solvers for Helmholtz problems, Springer, 2017, pp. 3–26.

[20] S. Kim and S. Kim, *Multigrid simulation for high-frequency solutions of the helmholtz problem in heterogeneous media*, SIAM Journal on Scientific Computing, 24 (2002), pp. 684–701.

[21] D. Lahaye and C. Vuik, *How to choose the shift in the shifted laplace preconditioner for the helmholtz equation combined with deflation*, in Modern Solvers for Helmholtz Problems, Springer, 2017, pp. 85–112.

[22] I. Livshits and A. Brandt, *Accuracy properties of the wave-ray multigrid algorithm for helmholtz equations*, SIAM Journal on Scientific Computing, 28 (2006), pp. 1228–1251.

[23] R. Nabben and C. Vuik, *A comparison of deflation and the balancing preconditioner*, SIAM Journal on Scientific Computing, 27 (2006), pp. 1742–1759.

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

[25] A. Sheikh, D. Lahaye, L. G. Ramos, R. Nabben, and C. Vuik, *Accelerating the shifted laplace preconditioner for the helmholtz equation by multilevel deflation*, Journal of Computational Physics, 322 (2016), pp. 473–490.

[26] A. Sheikh, D. Lahaye, and C. Vuik, *On the convergence of shifted Laplace preconditioner combined with multilevel deflation*, Numerical Linear Algebra with Applications, 20 (2013), pp. 645–662.