

ON THE SPECTRAL CLUSTERING OF A CLASS OF MULTIGRID PRECONDITIONERS

JOSE PABLO LUCERO LORCA

Abstract. This paper studies a common two-level multigrid construction for block-structured linear systems and identifies a simple way to describe how its smoothing and coarse-grid components interact. By examining the method through a collection of small coupling modes, we show that its behavior can be captured by a single scalar quantity for each mode. The main result is an explicit choice of smoothing parameters that makes all modes respond in the same way, causing the nontrivial eigenvalues of the preconditioned operator to collapse to a single value. This gives a clear and self-contained description of the ideal version of the method and provides a concrete target for designing related schemes. Although the exact spectral collapse requires ideal components, we also show that the same construction naturally produces operators that resemble those used in practical discretizations. Examples from finite-difference and discontinuous Galerkin settings illustrate how the ideal parameters can be used in practice.

1. Introduction. Large linear systems arising from the discretization of partial differential equations are often solved by exploiting a hierarchy of approximations. Multilevel methods construct such hierarchies by combining local relaxation with coarse representations of the problem, so that errors of different spatial scales are treated at different levels. Their success in practice depends on the interaction between these components: the smoother, the operators that transfer information between grids, and the coarse problems that represent the global behaviour of the solution. Despite their wide use, the precise algebraic mechanisms that make a multilevel iteration effective are often hidden behind problem-dependent analysis or numerical experimentation.

This paper develops a framework in which those mechanisms can be described and controlled explicitly. We consider a general nonsingular matrix written in two-by-two block form and examine the two-level iteration built from the restriction, prolongation, and inherited coarse operator defined by this partitioning. The partition may arise from a physical decomposition, a coloring or ordering of the unknowns, or simply from algebraic convenience; what matters is that the diagonal blocks represent subproblems that can be inverted or approximated efficiently. The central question is whether there exist relaxation parameters that collapse the spectrum of the preconditioned matrix to a fixed, low-dimensional set of eigenvalues, independent of the problem size or discretization. The results show that such parameters not only exist but can be written in closed form: for a sequence indexed by the number of pre- and post-relaxations, all nontrivial eigenvalues of the preconditioned matrix coincide. In this regime a Krylov method converges in a fixed number of steps—effectively turning the preconditioner into a direct solver.

The derivation that leads to this result is purely algebraic. By exploiting the natural invariance of the two-block partition, the two-level iteration can be expressed as a combination of identical low-dimensional problems, each acting independently on its own subspace. Within each of them, the interaction between relaxation and coarse correction reduces to a single scalar response that determines the overall convergence. This formulation reveals how specific parameter choices cause all these subproblems to behave identically, collapsing the spectrum of the full preconditioned matrix.

1.1. Related work. Multigrid methods have a long history, beginning with early analyses of relaxation for elliptic difference equations [12, 13, 1, 2, 16, 27, 7]. Algebraic multigrid (AMG) developed from a distinct viewpoint based on the CF-

splitting framework introduced in [4, 3, 25]. Ordering the unknowns into F- and C-points and writing the system in block form, exact block elimination shows that performing F-relaxation on the A_{ff} block together with coarse-grid correction using the Schur complement yields an exact two-level method. As noted in [17], this algebraic structure can be viewed as a nonsymmetric two-level V-cycle with no post-smoothing and a presmoothing acting only on the fine variables.

Ideal restriction and interpolation arise naturally in this block context. The classical ideal operators

$$P_{\text{ideal}} = \begin{pmatrix} -A_{ff}^{-1}A_{fc} \\ I \end{pmatrix}, \quad R_{\text{ideal}} = \begin{pmatrix} -A_{cf}A_{ff}^{-1} & I \end{pmatrix},$$

follow directly from the block LDU decomposition [14, 24, 9]. These operators define the reduction-based notion of ideality and form the foundation for many AMG algorithms. They support adaptive strategies [20, 5] and modern nonsymmetric methods based on approximate ideal restriction [21]. In the SPD setting, one typically enforces $R = P^*$ and forms $A_c = P^*AP$, an approach central to symmetric two-grid theory and energy-minimizing interpolation [11, 29], though this symmetry-based viewpoint does not naturally extend to nonsymmetric matrices.

Several nonsymmetric AMG approaches approximate R_{ideal} or P_{ideal} under fixed sparsity patterns. Constrained minimization techniques design interpolation or restriction operators [23, 28], while Petrov–Galerkin smoothed aggregation introduces transfer operators adapted to nonsymmetric systems [26, 6]. The AIR and nAIR algorithms [22] further develop reduction-based AMG, showing that local approximations of ideal restriction combined with F-relaxation can yield robust two-level convergence for general nonsymmetric problems.

Local Fourier Analysis (LFA) provides a complementary tool for predicting multigrid convergence factors and smoothing behavior on structured grids. Classical LFA results [8, 2, 7] analyze the Fourier symbol of the operator and the relaxation scheme to understand smoothing efficiency and the asymptotic two-grid contraction factor. Variants of this approach have been applied to discontinuous Galerkin methods and related discretizations, including settings where exact coarse-grid elimination leads to direct two-level solvers [19]. LFA-like block-structure analyses also appear in multilevel direct solvers for variable-coefficient problems [18], though their goals differ: these approaches pursue accurate hierarchical factorizations rather than iterative two-level convergence.

In contrast to these frameworks, the present work develops an algebraic two-level formulation based on invariant 2×2 subspaces. This reduction isolates the interaction of smoothing and coarse-grid correction and leads to closed-form relaxation parameters that enforce two-point spectral clustering for a broad class of block-partitioned nonsymmetric systems. The resulting perspective is independent of geometric assumptions, CF-ordering, or symmetry, and complements both reduction-based AMG theory and symbol-based Fourier analyses.

2. Problem setting. We consider a complex, invertible matrix $\tilde{A} \in \mathbb{C}^{N \times N}$. The goal of this section is to construct a two-level preconditioner M^{-1} based on a smoothing operator and an inherited coarse operator.

2.1. Two-level preconditioner. Let $R : \mathbb{C}^N \rightarrow \mathbb{C}^n$ and $P : \mathbb{C}^n \rightarrow \mathbb{C}^N$, with $n < N$, be linear *restriction* and *prolongation* operators, respectively, that map vectors between a *fine* space and a lower-dimensional *coarse* space, and vice versa. We define

the corresponding *coarse* operator as

$$M_0 := R\tilde{A}P,$$

We expect, in some sense, that M_0 captures the action of $\tilde{A}$ on elements of $\mathbb{C}^n$.

Let $S^{-1} : \mathbb{C}^N \rightarrow \mathbb{C}^N$ be a *smoother* linear operator that, while R filters out the components of vectors in $\mathbb{C}^N$ that cannot be represented in $\mathbb{C}^n$, S^{-1} *reduces*, intuitively, the components that R filters out in some norm.

We construct a two-level preconditioner M^{-1} with these operators, acting on a right-hand side f with m pre- and post-relaxations shown in Algorithm 2.1.

Algorithm 2.1 Two-level V-cycle preconditioner M^{-1} : action $u \leftarrow M^{-1}f$

```

1:  $x \leftarrow 0$ 
2: for  $i = 1, \dots, m$  do
3:    $x \leftarrow x + \alpha_i S^{-1}(f - \tilde{A}x)$  ▷ pre-smoothing
4: end for
5:  $x \leftarrow x + PM_0^{-1}R(f - \tilde{A}x)$  ▷ coarse correction,  $M_0 = R\tilde{A}P$ 
6: for  $i = 1, \dots, m$  do
7:    $x \leftarrow x + \alpha_i S^{-1}(f - \tilde{A}x)$  ▷ post-smoothing
8: end for
9: return  $u \leftarrow x$ 

```

Algorithm 2.1 defines M^{-1} as a linear operator that is expected to approximate $\tilde{A}^{-1}$ in the sense that it iteratively drives the residual $\|f - \tilde{A}u_i\|$ to zero for each iteration i , for some norm $\|\cdot\|$. We call this a preconditioner, where the preconditioned system to be solved is

$$M^{-1}\tilde{A}u = M^{-1}f,$$

and we expect $M^{-1}\tilde{A}$ to be easier to invert than $\tilde{A}$ alone, while providing the same solution u .

The relaxations $\alpha_i \in \mathbb{R}, \alpha_i > 0, i = 1 \dots m$ will be the center of attention of this manuscript, as we will find choices for these parameters that reduce the total number of applications of Algorithm 2.1 required to retrieve the solution u to a minimum.

2.2. Error operator. With the definitions above, we can express the operator providing the error after a single pre- or post-smoothing as

$$E_s(\alpha) = I - \alpha S^{-1}\tilde{A},$$

and likewise the error-propagation operator for the coarse correction

$$E_c = I - PM_0^{-1}R\tilde{A}.$$

The full two-level error operator corresponding to Algorithm 2.1 is then

$$(2.1) \quad E_m = \left(\prod_{i=m}^1 E_s(\alpha_i) \right) E_c \left(\prod_{i=1}^m E_s(\alpha_i) \right).$$

With these definitions, the preconditioned operator can be written as

$$M^{-1}\tilde{A} = I - E_m.$$

We can measure how good our choice of operators and relaxations is, by looking at the spectrum of E_m . A smaller spectrum will reduce the error faster and in turn, require fewer iterations to achieve a desired tolerance. The next section motivates our operator choices.

2.2.1. Algebraic two-block inverse identity and its link to the two-level scheme. A classical block-LDU identity expresses the inverse of a 2×2 block matrix in terms of its Schur complement (see, e.g., [14, 24, 9] and standard references on block factorizations). The formulation used here can be written in the two-level form employed throughout AMG theory. As in [17], we interpret the matrix $\tilde{A}$ in its 2×2 block form

$$\tilde{A} = \begin{bmatrix} A & B \\ C & D \end{bmatrix},$$

and select

$$\tilde{M}_0 = D - CA^{-1}B, \quad \tilde{S}^{-1} = \begin{bmatrix} A^{-1} & 0 \\ 0 & 0 \end{bmatrix}, \quad P = \begin{bmatrix} -A^{-1}B \\ I \end{bmatrix}, \quad \tilde{R} = \begin{bmatrix} 0 & I \end{bmatrix}.$$

The standard block identity then reads

$$(2.2) \quad \tilde{A}^{-1} = \tilde{S}^{-1} + P\tilde{M}_0^{-1}\tilde{R}(I - \tilde{A}\tilde{S}^{-1}).$$

and corresponds to a single application of the two-level Richardson relaxation scheme shown in Algorithm 2.2. In [17], this classical identity was simply expressed in that

Algorithm 2.2 Two-level realization of $\tilde{A}^{-1}$: action $u \leftarrow \tilde{A}^{-1}f$

- 1: $x \leftarrow 0$
 - 2: $x \leftarrow x + \tilde{S}^{-1}(f - \tilde{A}x)$ $\triangleright$ pre-smoothing
 - 3: $x \leftarrow x + P\tilde{M}_0^{-1}\tilde{R}(f - \tilde{A}x)$ $\triangleright$ coarse correction, $\tilde{M}_0 = \tilde{R}\tilde{A}P$
 - 4: **return** $u \leftarrow x$
-

two-level multigrid form (with one pre-smoothing step and no post-smoothing) to make the connection with reduction-based AMG explicit.

However, this choice of algorithm is not representative of common practices in multigrid preconditioning since

1. $\tilde{S}^{-1}$ has a nontrivial kernel.
2. The algorithm is not symmetric in that the operators after the coarse correction are not the operations before the coarse correction in reverse order.

On the other hand, the V-cycle algorithm in Algorithm 2.1 with the choices

$$(2.3) \quad P = \begin{bmatrix} -A^{-1}B \\ I \end{bmatrix}, \quad R = \begin{bmatrix} -CA^{-1} & I \end{bmatrix}, \quad M_0 = R\tilde{A}P, \quad S^{-1} = \begin{bmatrix} A^{-1} & 0 \\ 0 & D^{-1} \end{bmatrix},$$

displays an invertible smoother operator and what can be identified as a block-Jacobi smoother, but loses the property of the method being *direct*, the complete algorithm does not converge generally in a single iteration.

In this setting, we investigate the existence of a choice of α_i that retrieves the direct solver nature of Algorithm 2.1 with the operator choices (2.3). This is the subject of the next section.

3. Spectral optimization. This section analyzes the error operator (2.1) and selects relaxation parameters so that the spectrum of the preconditioned operator is shrunk to the largest extent possible, ideally independent of N . We isolate a 2×2 invariant subspace associated with each eigenpair of an auxiliary operator T , reduce the global mapping to a scalar response $r_m(\lambda)$ on that subspace, and then choose $\{\alpha_i\}_{i=1}^m$ to make $r_m(\lambda)$ independent of λ . Throughout, $\tilde{A}$ is invertible (possibly complex) and admits a two-by-two block partition with invertible diagonal blocks.

3.1. Preliminaries. We establish the algebraic framework for the 2×2 reduction that underlies the spectral analysis of the error operator. The goal is to obtain expressions for the error operators that are more amenable to the analysis we perform afterwards.

DEFINITION 3.1 (Coupling operator). *We define the coupling operator*

$$T = A^{-1}BD^{-1}C.$$

The matrix T encodes the effective interaction between the two blocks through the off-diagonal coupling B . We'll observe later that minimizing the spectrum of E_m relies on isolating the spectrum of T . We assume throughout that T is diagonalizable over $\mathbb{C}$ and admits a complete set of eigenvectors. The spectrum $\sigma(T)$ may in general be complex.

We use the operators from Algorithm 2.1 defined in (2.3). The smoother error operator keeps the form

$$(3.1) \quad \begin{aligned} E_s(\alpha) &= I - \alpha \begin{bmatrix} A^{-1} & 0 \\ 0 & D^{-1} \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \\ &= \begin{bmatrix} (1 - \alpha)I & -\alpha A^{-1}B \\ -\alpha D^{-1}C & (1 - \alpha)I \end{bmatrix}. \end{aligned}$$

For the coarse correction, we start from

$$E_c = I - PM_0^{-1}R\tilde{A}.$$

First compute

$$\begin{aligned} R\tilde{A} &= \begin{bmatrix} -CA^{-1} & I \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} -CA^{-1}A + C & -CA^{-1}B + D \end{bmatrix} \\ &= \begin{bmatrix} 0 & D - CA^{-1}B \end{bmatrix} = \begin{bmatrix} 0 & M_0 \end{bmatrix}. \end{aligned}$$

Hence

$$M_0^{-1}R\tilde{A} = M_0^{-1} \begin{bmatrix} 0 & M_0 \end{bmatrix} = \begin{bmatrix} 0 & I \end{bmatrix}.$$

Multiplying on the left by P gives

$$PM_0^{-1}R\tilde{A} = \begin{bmatrix} -A^{-1}B \\ I \end{bmatrix} \begin{bmatrix} 0 & I \end{bmatrix} = \begin{bmatrix} 0 & -A^{-1}B \\ 0 & I \end{bmatrix}.$$

Therefore

$$(3.2) \quad E_c = I - PM_0^{-1}R\tilde{A} = \begin{bmatrix} I & A^{-1}B \\ 0 & 0 \end{bmatrix}.$$

Equations (3.1) and (3.2) will be used in the optimizations below.

3.2. Invariant subspaces and scalar reduction. The operators $E_s(\alpha)$ and E_c act on the full two-block error vector and therefore have a block structure. To analyze their combined effect, we decompose the space into smaller subspaces on which their action is completely determined by a single scalar parameter. This decomposition follows from the eigenstructure of the coupling operator $T = A^{-1}BD^{-1}C$. Each eigenpair (w, λ) of T generates a two-dimensional invariant subspace on which both operators act independently. The following definition introduces this family of subspaces.

DEFINITION 3.2 (Invariant subspaces). *For each eigenpair (w, λ) of T , with $Tw = \lambda w$, define*

$$v_1 = \begin{bmatrix} w \\ 0 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 0 \\ D^{-1}Cw \end{bmatrix}, \quad \mathcal{V}_\lambda = \text{span}\{v_1, v_2\}.$$

Each $\mathcal{V}_\lambda$ captures the interaction between the two diagonal blocks of $\tilde{A}$ mediated by B . Because T is assumed diagonalizable with a complete set of eigenvectors, these subspaces span the portion of the system affected by the coupling, and they have the invariant properties verified below.

We now verify that the smoother preserves every subspace $\mathcal{V}_\lambda$ in this family; that is, for each eigenpair (w, λ) of T , one smoothing step maps $\mathcal{V}_\lambda$ into itself.

PROPOSITION 3.3 (Invariance under the smoother). *For any $\alpha \in \mathbb{R}$ and any eigenpair (w, λ) of T , possibly complex, the smoother $E_s(\alpha)$ defined in (3.1) satisfies*

$$E_s(\alpha)\mathcal{V}_\lambda \subseteq \mathcal{V}_\lambda.$$

Proof. Applying $E_s(\alpha)$ to v_1 and v_2 gives

$$\begin{aligned} E_s(\alpha)v_1 &= \begin{bmatrix} (1-\alpha)I & -\alpha A^{-1}B \\ -\alpha D^{-1}C & (1-\alpha)I \end{bmatrix} \begin{bmatrix} w \\ 0 \end{bmatrix} = \begin{bmatrix} (1-\alpha)w \\ -\alpha D^{-1}Cw \end{bmatrix} = (1-\alpha)v_1 - \alpha v_2, \\ E_s(\alpha)v_2 &= \begin{bmatrix} (1-\alpha)I & -\alpha A^{-1}B \\ -\alpha D^{-1}C & (1-\alpha)I \end{bmatrix} \begin{bmatrix} 0 \\ D^{-1}Cw \end{bmatrix} = \begin{bmatrix} -\alpha A^{-1}BD^{-1}Cw \\ (1-\alpha)D^{-1}Cw \end{bmatrix} \\ &= -\alpha\lambda \begin{bmatrix} w \\ 0 \end{bmatrix} + (1-\alpha) \begin{bmatrix} 0 \\ D^{-1}Cw \end{bmatrix} = -\alpha\lambda v_1 + (1-\alpha)v_2. \end{aligned}$$

Both images are linear combinations of v_1 and v_2 , hence $E_s(\alpha)$ maps $\mathcal{V}_\lambda$ into itself. $\square$

We next establish the same invariance property for the coarse correction.

PROPOSITION 3.4 (Invariance under the coarse correction). *Let E_c be given by (3.2). For any (possibly complex) eigenpair (w, λ) of T ,*

$$E_c\mathcal{V}_\lambda \subseteq \mathcal{V}_\lambda.$$

Proof. Using the block form of E_c ,

$$\begin{aligned} E_cv_1 &= \begin{bmatrix} I & A^{-1}B \\ 0 & 0 \end{bmatrix} \begin{bmatrix} w \\ 0 \end{bmatrix} = v_1, \\ E_cv_2 &= \begin{bmatrix} I & A^{-1}B \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ D^{-1}Cw \end{bmatrix} = \begin{bmatrix} A^{-1}BD^{-1}Cw \\ 0 \end{bmatrix} = \begin{bmatrix} Tw \\ 0 \end{bmatrix} = \lambda \begin{bmatrix} w \\ 0 \end{bmatrix} = \lambda v_1. \end{aligned}$$

Hence E_c also maps $\mathcal{V}_\lambda$ into itself. $\square$

The previous two propositions imply that any product of $E_s(\alpha)$ and E_c —in particular the symmetric composition E_m —preserves each $\mathcal{V}_\lambda$. Therefore, on the subspace spanned by the $\mathcal{V}_\lambda$, the operator decomposes into a direct sum of independent two-dimensional problems, one for each eigenvalue λ of T . On each $\mathcal{V}_\lambda$, every vector can be written as $av_1 + bv_2$, where $(a, b)^\top$ are scalar coefficients. It is thus natural to describe the action of $E_s(\alpha)$ and E_c on $\mathcal{V}_\lambda$ using 2×2 matrices acting on $(a, b)^\top$.

LEMMA 3.5 (Matrix form in the basis $\{v_1, v_2\}$). *Relative to the basis $\{v_1, v_2\}$, the restrictions of $E_s(\alpha)$ and E_c to $\mathcal{V}_\lambda$ are*

$$\tilde{E}_s(\alpha) := [E_s(\alpha)]_{\{v_1, v_2\}} = \begin{bmatrix} 1-\alpha & -\alpha\lambda \\ -\alpha & 1-\alpha \end{bmatrix}, \quad \tilde{E}_c := [E_c]_{\{v_1, v_2\}} = \begin{bmatrix} 1 & \lambda \\ 0 & 0 \end{bmatrix}.$$

Proof. Recall that for an eigenpair (w, λ) of $T = A^{-1}BD^{-1}C$, we defined

$$v_1 = \begin{bmatrix} w \\ 0 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 0 \\ D^{-1}Cw \end{bmatrix}, \quad \mathcal{V}_\lambda = \text{span}\{v_1, v_2\}.$$

The matrix representation of a linear operator in the basis $\{v_1, v_2\}$ is obtained by writing the images of v_1 and v_2 as linear combinations of v_1 and v_2 ; the coefficients are the columns of the matrix.

First consider the smoother error operator $E_s(\alpha)$, which we wrote as

$$E_s(\alpha) = \begin{bmatrix} (1-\alpha)I & -\alpha A^{-1}B \\ -\alpha D^{-1}C & (1-\alpha)I \end{bmatrix}.$$

Acting on v_1 and v_2 gives

$$\begin{aligned} E_s(\alpha)v_1 &= \begin{bmatrix} (1-\alpha)I & -\alpha A^{-1}B \\ -\alpha D^{-1}C & (1-\alpha)I \end{bmatrix} \begin{bmatrix} w \\ 0 \end{bmatrix} = \begin{bmatrix} (1-\alpha)w \\ -\alpha D^{-1}Cw \end{bmatrix} = (1-\alpha)v_1 - \alpha v_2, \\ E_s(\alpha)v_2 &= \begin{bmatrix} (1-\alpha)I & -\alpha A^{-1}B \\ -\alpha D^{-1}C & (1-\alpha)I \end{bmatrix} \begin{bmatrix} 0 \\ D^{-1}Cw \end{bmatrix} = \begin{bmatrix} -\alpha A^{-1}BD^{-1}Cw \\ (1-\alpha)D^{-1}Cw \end{bmatrix} \\ &= \begin{bmatrix} -\alpha Tw \\ (1-\alpha)D^{-1}Cw \end{bmatrix} = -\alpha \lambda v_1 + (1-\alpha)v_2, \end{aligned}$$

since $Tw = \lambda w$. Thus, in the basis $\{v_1, v_2\}$,

$$[E_s(\alpha)]_{\{v_1, v_2\}} = \begin{bmatrix} 1-\alpha & -\alpha\lambda \\ -\alpha & 1-\alpha \end{bmatrix} =: \tilde{\mathcal{E}}_s(\alpha).$$

Now consider the coarse-correction error operator

$$E_c = \begin{bmatrix} I & A^{-1}B \\ 0 & 0 \end{bmatrix}.$$

Acting on v_1 and v_2 ,

$$\begin{aligned} E_cv_1 &= \begin{bmatrix} I & A^{-1}B \\ 0 & 0 \end{bmatrix} \begin{bmatrix} w \\ 0 \end{bmatrix} = \begin{bmatrix} w \\ 0 \end{bmatrix} = v_1, \\ E_cv_2 &= \begin{bmatrix} I & A^{-1}B \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ D^{-1}Cw \end{bmatrix} = \begin{bmatrix} A^{-1}BD^{-1}Cw \\ 0 \end{bmatrix} = \begin{bmatrix} Tw \\ 0 \end{bmatrix} = \lambda v_1. \end{aligned}$$

Therefore, relative to the basis $\{v_1, v_2\}$,

$$[E_c]_{\{v_1, v_2\}} = \begin{bmatrix} 1 & \lambda \\ 0 & 0 \end{bmatrix} =: \tilde{\mathcal{E}}_c.$$

This proves the stated matrix forms. $\square$

The matrices above depend only on the eigenvalue λ of T and the relaxation parameter α ; here λ may be complex, and $\tau := \sqrt{\lambda}$ is defined according to the branch choice introduced below. Although the analysis can proceed directly from this form, a symmetric scaling of the basis reveals additional structure and simplifies subsequent expressions.

DEFINITION 3.6 (Symmetric scaling). *To obtain a balanced representation, set*

$$u_1 = v_1, \quad u_2 = \frac{1}{\sqrt{\lambda}} v_2.$$

For $\lambda \neq 0$, this scaling yields

$$\mathcal{E}_s(\alpha, \lambda) := \begin{bmatrix} 1 - \alpha & -\alpha\sqrt{\lambda} \\ -\alpha\sqrt{\lambda} & 1 - \alpha \end{bmatrix}, \quad \mathcal{E}_c(\lambda) := \begin{bmatrix} 1 & \sqrt{\lambda} \\ 0 & 0 \end{bmatrix}.$$

For $\lambda = 0$, these formulas are obtained by continuity in $\tau = \sqrt{\lambda}$. This scaling equalizes the off-diagonal entries of $\tilde{\mathcal{E}}_s(\alpha, \lambda)$, making the underlying symmetry explicit.

REMARK 1. *Branch choice and the case $\lambda = 0$. Throughout, we write $\tau = \sqrt{\lambda}$ using the principal branch of the square root, i.e.*

$$\sqrt{\lambda} := \exp\left(\frac{1}{2} \log \lambda\right),$$

where $\log$ denotes the principal complex logarithm, with branch cut on $(-\infty, 0]$ and $\arg \lambda \in (-\pi, \pi]$. The scaling $u_2 = \frac{1}{\sqrt{\lambda}} v_2$ is used only for $\lambda \neq 0$; when $\lambda = 0$ we retain the unscaled basis $\{v_1, v_2\}$ and interpret all formulas by continuity, that is, by taking the limit $\tau \rightarrow 0$. None of the conclusions depend on the sign of $\sqrt{\lambda}$, since the expressions for $r_m(\lambda)$ are even in τ or involve only the symmetric combination of τ and $-\tau$. If the spectrum of T intersects the branch cut, one may equivalently select any fixed branch that is analytic on a neighborhood of $\sigma(T)$; the analysis proceeds componentwise on each connected subset of $\sigma(T)$ disjoint from the cut.

We next show that all the matrices $\mathcal{E}_s(\alpha_i, \lambda)$ commute with one another. For any α_i, α_j ,

$$\begin{aligned} \mathcal{E}_s(\alpha_i, \lambda) \mathcal{E}_s(\alpha_j, \lambda) = \\ \begin{bmatrix} (1 - \alpha_i)(1 - \alpha_j) + \alpha_i \alpha_j \lambda & -\sqrt{\lambda}(\alpha_i(1 - \alpha_j) + \alpha_j(1 - \alpha_i)) \\ -\sqrt{\lambda}(\alpha_i(1 - \alpha_j) + \alpha_j(1 - \alpha_i)) & (1 - \alpha_i)(1 - \alpha_j) + \alpha_i \alpha_j \lambda \end{bmatrix}, \end{aligned}$$

which is symmetric in (i, j) ; hence $\mathcal{E}_s(\alpha_i, \lambda)$ and $\mathcal{E}_s(\alpha_j, \lambda)$ commute. It is useful to define

$$\mathcal{E}_{s,m}(\lambda) := \prod_{i=1}^m \mathcal{E}_s(\alpha_i, \lambda) = \prod_{i=m}^1 \mathcal{E}_s(\alpha_i, \lambda).$$

PROPOSITION 3.7 (Rank-one factorization). *In the scaled basis, the coarse operator $\mathcal{E}_c(\lambda)$ factors as*

$$\mathcal{E}_c(\lambda) := u v(\lambda)^\top, \quad u = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad v(\lambda)^\top = \begin{bmatrix} 1 & \sqrt{\lambda} \end{bmatrix}.$$

Proof. A direct multiplication verifies $u v(\lambda)^\top = \begin{bmatrix} 1 & \sqrt{\lambda} \\ 0 & 0 \end{bmatrix} = \mathcal{E}_c(\lambda)$. $\square$

The restriction of the global two-level error operator E_m to the invariant subspace $\mathcal{V}_\lambda$ is obtained by replacing $E_s(\alpha)$ and E_c with their 2×2 scalar representations $\mathcal{E}_{s,m}(\lambda)$ and $\mathcal{E}_c(\lambda)$:

$$\mathcal{E}_m(\lambda) = \left(\prod_{i=m}^1 \mathcal{E}_s(\alpha_i, \lambda) \right) \mathcal{E}_c(\lambda) \left(\prod_{i=1}^m \mathcal{E}_s(\alpha_i, \lambda) \right) = \mathcal{E}_{s,m}(\lambda) \mathcal{E}_c(\lambda) \mathcal{E}_{s,m}(\lambda),$$

LEMMA 3.8 (Scalar reduction). *Let $\mathcal{E}_c(\lambda) = uv(\lambda)^\top$ as in Proposition 3.7, with $u = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $v(\lambda)^\top = [1 \quad \sqrt{\lambda}]$. Then the nonzero eigenvalue of $\mathcal{E}_m(\lambda)$ is*

$$r_m(\lambda) = v(\lambda)^\top \mathcal{E}_{s,m}(\lambda)^2 u.$$

Proof. Since $\mathcal{E}_c(\lambda) = uv(\lambda)^\top$, we have

$$\mathcal{E}_m(\lambda) = \mathcal{E}_{s,m}(\lambda)uv(\lambda)^\top \mathcal{E}_{s,m}(\lambda) = (\mathcal{E}_{s,m}(\lambda)u) (v(\lambda)^\top \mathcal{E}_{s,m}(\lambda)).$$

Applying it to $w := \mathcal{E}_{s,m}(\lambda)u$ gives

$$\begin{aligned} \mathcal{E}_m(\lambda)\mathcal{E}_{s,m}(\lambda)u &= (\mathcal{E}_{s,m}(\lambda)u) (v(\lambda)^\top \mathcal{E}_{s,m}(\lambda)) \mathcal{E}_{s,m}(\lambda)u \\ \mathcal{E}_m(\lambda)\mathcal{E}_{s,m}(\lambda)u &= (\mathcal{E}_{s,m}(\lambda)u) (v(\lambda)^\top \mathcal{E}_{s,m}(\lambda)^2 u) \\ \mathcal{E}_m(\lambda)w &= w \underbrace{(v(\lambda)^\top \mathcal{E}_{s,m}(\lambda)^2 u)}_{=: r_m(\lambda)} \end{aligned}$$

so we obtain an expression for the corresponding eigenvalue $r_m(\lambda) = v(\lambda)^\top \mathcal{E}_{s,m}(\lambda)^2 u$. $\square$

This lemma provides the explicit expression for the nonzero eigenvalue of the error operator restricted to each invariant subspace $\mathcal{V}_\lambda$, which fully characterizes the iteration's convergence on that subspace.

3.3. Spectral clustering. This subsection brings together all previous results to establish the spectral properties of the two-level error operator. We combine the invariant-subspace structure, the explicit two-by-two representations, and the scalar response derived earlier to prove that the method achieves perfect clustering of the nonzero eigenvalues of the two-level error operator. The flow proceeds from the spectral decomposition of the global operator, through the diagonalization of the local action, to the identification of the relaxation parameters that flatten the spectrum.

3.3.1. Diagonalization. To expose the dependence of $r_m(\lambda)$ on the relaxation parameters, we compute explicitly the square of $\mathcal{E}_s(\alpha, \lambda)$, which acts on each $\mathcal{V}_\lambda$. Its symmetry allows a common orthogonal diagonalization, but first we expand its algebraic form to reveal the pattern of coefficients.

With

$$\mathcal{E}_s(\alpha, \lambda) = \begin{bmatrix} 1 - \alpha & -\alpha\sqrt{\lambda} \\ -\alpha\sqrt{\lambda} & 1 - \alpha \end{bmatrix},$$

a direct multiplication gives

$$\begin{aligned} \mathcal{E}_s(\alpha, \lambda)^2 &= \begin{bmatrix} (1 - \alpha)^2 + \alpha^2\lambda & -2\alpha(1 - \alpha)\sqrt{\lambda} \\ -2\alpha(1 - \alpha)\sqrt{\lambda} & (1 - \alpha)^2 + \alpha^2\lambda \end{bmatrix}, \\ r_1(\lambda) &= v(\lambda)^\top \mathcal{E}_s(\alpha, \lambda)^2 u = (1 - \alpha)^2 + \alpha^2\lambda - 2\alpha(1 - \alpha)\lambda. \end{aligned}$$

We realize that choosing $\alpha = \frac{2}{3}$ eliminates the dependence of $r_1(\lambda)$ on λ , then $\mathcal{E}_1(\lambda)$ does not depend on λ , and its eigenvalues are repeated instances of $\{0, \frac{1}{9}\}$ which implies that Algorithm 2.1 as a preconditioner of a Krylov method converges in 2 iterations. It is a direct method!

The identities in this section will be the building blocks to prove further that $r_m(\lambda)$ can be made independent of λ for any $m > 0$.

3.3.2. Diagonalization and explicit $r_m(\lambda)$. Each $\mathcal{E}_s(\alpha_i, \lambda)$ can now be diagonalized by the fixed orthogonal matrix

$$Q = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

yielding

$$Q^\top \mathcal{E}_s(\alpha_i, \lambda) Q = \begin{bmatrix} (1 - \alpha_i) - \alpha_i \sqrt{\lambda} & 0 \\ 0 & (1 - \alpha_i) + \alpha_i \sqrt{\lambda} \end{bmatrix}$$

Because all $\mathcal{E}_s(\alpha_i, \lambda)$ commute, their products are easily represented as

$$Q^\top \mathcal{E}_{s,m}(\lambda)^2 Q = \begin{bmatrix} \prod_{i=1}^m \left((1 - \alpha_i) - \alpha_i \sqrt{\lambda} \right)^2 & 0 \\ 0 & \prod_{i=1}^m \left((1 - \alpha_i) + \alpha_i \sqrt{\lambda} \right)^2 \end{bmatrix}$$

Projecting onto u and $v(\lambda)$ gives

$$Q^\top u = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad v(\lambda)^\top Q = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 + \sqrt{\lambda} & 1 - \sqrt{\lambda} \end{bmatrix},$$

and therefore

(3.3)

$$\begin{aligned} r_m(\lambda) &= v(\lambda)^\top \mathcal{E}_{s,m}(\lambda)^2 u \\ &= \frac{1}{2} \left(1 + \sqrt{\lambda} \right) \prod_{i=1}^m \left((1 - \alpha_i) - \alpha_i \sqrt{\lambda} \right)^2 + \frac{1}{2} \left(1 - \sqrt{\lambda} \right) \prod_{i=1}^m \left((1 - \alpha_i) + \alpha_i \sqrt{\lambda} \right)^2. \end{aligned}$$

This expression makes the scalar response completely explicit: it is a combination of two degree- $2m$ polynomials in $\sqrt{\lambda}$. Here $\tau = \sqrt{\lambda}$ may be complex; for real $\lambda \geq 0$ this coincides with the standard positive square root. In the algebraic derivations that follow, $\sqrt{\lambda}$ and the auxiliary variable τ are treated as complex analytic quantities to ensure validity for complex spectra.

3.3.3. Perfect clustering for arbitrary m . To achieve perfect clustering, we must determine parameters $\{\alpha_i\}$ such that $r_m(\lambda)$ is constant. The proof relies on a trigonometric product identity, which we recall first.

PROPOSITION 3.9. *Let $m \geq 1$, $\theta_i = \frac{2\pi i}{2m+1}$ for $i = 1, \dots, m$, and let $x \in \mathbb{C} \setminus \{0\}$ with $t = \frac{x + \frac{1}{x}}{2}$. Then*

$$(3.4) \quad \prod_{i=1}^m (\cos \theta_i \pm t) = \frac{x^{2m+1} \pm 1}{2^m x^m (x \pm 1)}.$$

Proof. From $t = \frac{1}{2}(x + x^{-1})$ we have, for each i ,

$$\cos \theta_i \pm t = \cos \theta_i \pm \frac{x + \frac{1}{x}}{2} = \frac{2 \cos \theta_i x \pm (x^2 + 1)}{2x} = \frac{x^2 \pm 2x \cos \theta_i + 1}{2x}.$$

Taking the product over $i = 1, \dots, m$ gives

$$\prod_{i=1}^m (\cos \theta_i \pm t) = \prod_{i=1}^m \frac{x^2 \pm 2x \cos \theta_i + 1}{2x} = \frac{1}{(2x)^m} \prod_{i=1}^m (x^2 \pm 2x \cos \theta_i + 1).$$

By [15, §1.39, 1.396(3) and 1.396(2)]

$$\prod_{i=1}^m (x^2 + 2x \cos \theta_i + 1) = \frac{x^{2m+1} + 1}{x + 1}, \quad \prod_{i=1}^m (x^2 - 2x \cos \theta_i + 1) = \frac{x^{2m+1} - 1}{x - 1}.$$

Substituting each case yields the claim. $\square$

The next lemma identifies the relaxation parameters that make $r_m(\lambda)$ constant, independent of λ .

LEMMA 3.10. *Let $m \geq 1$ and $\alpha_i \neq 0$ for $i = 1, \dots, m$. Define*

$$(3.5) \quad r_m(\tau) = \frac{1}{2}(1 + \tau) \prod_{i=1}^m ((1 - \alpha_i) - \alpha_i \tau)^2 + \frac{1}{2}(1 - \tau) \prod_{i=1}^m ((1 - \alpha_i) + \alpha_i \tau)^2,$$

where $\tau = \sqrt{\lambda}$ may take complex values. Set $c_i := \frac{1 - \alpha_i}{\alpha_i}$ and assume

$$(3.6) \quad c_i = -\cos \theta_i, \quad \theta_i = \frac{2\pi i}{2m+1}, \quad i = 1, \dots, m.$$

Then $r_m(\tau) = \frac{1}{(2m+1)^2}$, i.e. r_m is independent of τ .

Proof. Using $(1 - \alpha_i) \mp \alpha_i \tau = \alpha_i(c_i \mp \tau)$, we write

$$r_m(\tau) = \left(\prod_{i=1}^m \alpha_i^2 \right) \left[\frac{1}{2}(1 + \tau) \prod_{i=1}^m (c_i - \tau)^2 + \frac{1}{2}(1 - \tau) \prod_{i=1}^m (c_i + \tau)^2 \right].$$

With $x \in \mathbb{C} \setminus \{0\}$ such that $t = \frac{x + \frac{1}{x}}{2}$, Proposition 3.9 gives

$$\prod_{i=1}^m (\cos \theta_i + \tau) = \frac{x^{2m+1} + 1}{2^m x^m (x + 1)}, \quad \prod_{i=1}^m (\cos \theta_i - \tau) = \frac{x^{2m+1} - 1}{2^m x^m (x - 1)}.$$

With $c_i = -\cos \theta_i$ we have $(c_i - \tau)^2 = (\cos \theta_i + \tau)^2$ and $(c_i + \tau)^2 = (\cos \theta_i - \tau)^2$. Substituting yields

$$\begin{aligned} r_m(\tau) &= \left(\prod_{i=1}^m \alpha_i^2 \right) \left\{ \frac{1}{2} \left(1 + \frac{x + \frac{1}{x}}{2} \right) \left(\frac{x^{2m+1} + 1}{2^m x^m (x + 1)} \right)^2 \right. \\ &\quad \left. + \frac{1}{2} \left(1 - \frac{x + \frac{1}{x}}{2} \right) \left(\frac{x^{2m+1} - 1}{2^m x^m (x - 1)} \right)^2 \right\} \\ &= \left(\prod_{i=1}^m \alpha_i^2 \right) \left\{ \frac{(x + 1)^2}{4x} \left(\frac{x^{2m+1} + 1}{2^m x^m (x + 1)} \right)^2 - \frac{(x - 1)^2}{4x} \left(\frac{x^{2m+1} - 1}{2^m x^m (x - 1)} \right)^2 \right\} \\ &= \left(\prod_{i=1}^m \alpha_i^2 \right) \left\{ \frac{(x^{2m+1} + 1)^2 - (x^{2m+1} - 1)^2}{2^{2m+2} x^{2m+1}} \right\} \\ &= \left(\prod_{i=1}^m \alpha_i^2 \right) \left\{ \frac{[(x^{2m+1} + 1) - (x^{2m+1} - 1)][(x^{2m+1} + 1) + (x^{2m+1} - 1)]}{2^{2m+2} x^{2m+1}} \right\} \\ &= \left(\prod_{i=1}^m \alpha_i^2 \right) \left\{ \frac{1}{2^{2m+2} x^{2m+1}} \cdot 2 \cdot 2x^{2m+1} \right\} \\ &= \frac{\prod_{i=1}^m \alpha_i^2}{2^{2m}} \end{aligned}$$

Evaluating Proposition 3.9 for $\tau \rightarrow 1$ gives

$$\prod_{i=1}^m (1 - \cos \theta_i)^2 = \left(\frac{2m+1}{2^m} \right)^2.$$

Thus

$$\prod_{i=1}^m \alpha_i^2 = \left(\frac{2^m}{2m+1} \right)^2$$

which proves $r_m(\tau) = \frac{1}{(2m+1)^2}$. $\square$

3.3.4. Spectral consequence for Algorithm 2.1. The following theorem states that, for the specific ingredients used in Algorithm 2.1, the symmetric two-level preconditioner produces a system whose spectrum is perfectly clustered and independent of the problem size.

THEOREM 3.11 (Clustered spectrum of the symmetric two-level preconditioner).

Let

$$\tilde{A} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

be complex with A, D invertible. Define $T := A^{-1}BD^{-1}C$ and assume that it is diagonalizable over $\mathbb{C}$ and admits a complete set of eigenvectors. The two-level method of Algorithm 2.1 with the following components:

$$S^{-1} = \begin{bmatrix} A^{-1} & 0 \\ 0 & D^{-1} \end{bmatrix}, \quad P = \begin{bmatrix} -A^{-1}B \\ I \end{bmatrix}, \quad R = \begin{bmatrix} -CA^{-1} & I \end{bmatrix},$$

and

$$\alpha_j = \frac{1}{1 - \cos\left(\frac{2\pi j}{2m+1}\right)}, \quad j = 1, \dots, m.$$

defines a preconditioned system $M^{-1}\tilde{A}$ whose spectrum consists of only two distinct values,

$$(3.7) \quad \sigma(M^{-1}\tilde{A}) = \left\{ 1, 1 - \frac{1}{(2m+1)^2} \right\},$$

independent of the number of unknowns.

Proof. By Propositions 3.3 and 3.4, the error operator of the smoother $E_s(\alpha)$ and the coarse correction E_c preserve, for every (possibly complex) eigenpair (w, λ) of $T = A^{-1}BD^{-1}C$, the two-dimensional subspace

$$\mathcal{V}_\lambda = \text{span} \left\{ \begin{bmatrix} w \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ D^{-1}Cw \end{bmatrix} \right\}.$$

Hence the error operator of the full two-level method

$$E_m = \left(\prod_{j=m}^1 E_s(\alpha_j) \right) E_c \left(\prod_{j=1}^m E_s(\alpha_j) \right)$$

is block diagonal on the sum of the subspaces $\mathcal{V}_\lambda$, with one 2×2 block for each eigenvalue λ of T , while on vectors of the form $[0; y]$ with $By = 0$ it acts as the zero operator.

In the scaled basis of Section 3.2, the restriction of $E_s(\alpha)$ to $\mathcal{V}_\lambda$ equals $\mathcal{E}_s(\alpha, \lambda)$ and the restriction of E_c equals a rank-one map $uv(\lambda)^\top$. Therefore, on $\mathcal{V}_\lambda$,

$$\mathcal{E}_m(\lambda) = \mathcal{E}_{s,m}(\lambda)uv(\lambda)^\top \mathcal{E}_{s,m}(\lambda), \quad \mathcal{E}_{s,m}(\lambda) = \mathcal{E}_s(\alpha_m, \lambda) \cdots \mathcal{E}_s(\alpha_1, \lambda),$$

has rank at most one. By Lemma 3.8, its unique nonzero eigenvalue is

$$r_m(\lambda) = v(\lambda)^\top \mathcal{E}_{s,m}(\lambda)^2 u.$$

Using the commutativity and diagonalization from Section 3.3.2 yields the explicit form (3.3). With the choice of relaxation parameters specified in the theorem, Lemma 3.10 gives

$$r_m(\lambda) = (2m+1)^{-2} \quad \text{for all } \lambda.$$

Because $r_m(\lambda)$ is an even analytic function of $t = \sqrt{\lambda}$, the result is independent of the branch chosen for $\sqrt{\lambda}$. Hence every restricted block of $\mathcal{E}_m$ has spectrum $\{0, (2m+1)^{-2}\}$, so the spectrum of the preconditioned matrix

$$M^{-1}\tilde{A} = I - E_m$$

is

$$(3.8) \quad \sigma(M^{-1}\tilde{A}) = \left\{ 1, 1 - \frac{1}{(2m+1)^2} \right\}.$$

The multiplicities follow from the decomposition into the $\mathcal{V}_\lambda$ blocks and are independent of the total number of unknowns. This proves the claim. $\square$

This theorem provides the operational result: when Algorithm 2.1 is implemented with these definitions of restriction, prolongation, smoother, and relaxation parameters, the preconditioned system has perfectly clustered eigenvalues. As a consequence, a Krylov solver applied to it converges in two iterations.

4. Clustering in practice. This section highlights several practical instances in which the clustering phenomenon appears, showcasing its behavior beyond the abstract setting. A number of these examples come from active research efforts and will be presented in forthcoming papers.

4.1. Finite differences 1D and 2D. We now illustrate the previous theory on a simple model problem: the two-dimensional Poisson equation with homogeneous Dirichlet boundary conditions on the unit square. We discretize $-\Delta$ on a uniform finite-difference mesh and build a two-level method whose algebraic ingredients follow Theorem 3.11 as closely as possible under a sparsity constraint.

Let $A_1 \in \mathbb{R}^{N \times N}$ be the one-dimensional second-order finite-difference Laplacian with stencil $\frac{1}{h^2}[-1 \ 2 \ -1]$ on a uniform grid with N interior points and spacing h . The two-dimensional discrete operator is assembled as

$$(4.1) \quad A_2 = A_1 \otimes I_N + I_N \otimes A_1,$$

where I_N denotes the $N \times N$ identity matrix. In all experiments we fix a 16×16 interior grid of unknowns, so that $A_2 \in \mathbb{R}^{256 \times 256}$. The right-hand side is taken as

$f \equiv 1$ so that the underlying PDE solution is smooth and the discrete operator is well conditioned for the purposes of this test.

If we were to apply Theorem 3.11 *directly* to the two-dimensional matrix A_2 , using its 2×2 block partition and the corresponding operators

$$S^{-1}, \quad P, \quad R, \quad M_0 = RA_2P,$$

then the theorem would again guarantee a perfectly clustered spectrum and therefore a direct method when used as a preconditioner for GMRES: the preconditioned operator would have only two distinct eigenvalues. However, in this setting the block $-A^{-1}B$ is a dense matrix for the 2D Laplacian, so both P and R become full. This would destroy the sparsity structure that makes multilevel methods efficient, and is therefore not acceptable in practice.

To preserve sparsity, we instead apply the algebraic construction of Theorem 3.11 to the *one-dimensional* matrix A_1 and then lift the resulting operators to two dimensions by tensor products. The first step is a red/black reordering of the one-dimensional grid: we permute the unknown vector so that all “red” points (say, odd indices) come first, followed by all “black” points (even indices). In this reordered basis, A_1 takes the 2×2 block form

$$A_1 \sim \begin{bmatrix} A & B \\ B^\top & D \end{bmatrix},$$

where A and D correspond to the red-red and black-black interactions and B couples the two colors. From this block structure we build the one-dimensional algebraic operators P_1 and $R_1 = P_1^\top$ according to Theorem 3.11, and then undo the red/black reordering so that P_1 and R_1 act on the original ordering of unknowns. The corresponding two-dimensional operators are then defined by tensor products

$$(4.2) \quad R = R_1 \otimes R_1, \quad P = P_1 \otimes P_1,$$

and the inherited coarse operator is $M_0 = RA_2P$.

It is straightforward to verify that this tensor-product construction reproduces, in the two-dimensional finite-difference setting, the *classical* restriction and prolongation pair: P acts as the standard bilinear interpolation operator that reconstructs fine-grid values from coarse-grid ones, while $R = P^\top$ corresponds to the full-weighting restriction. Thus, the algebraic formulation obtained from the one-dimensional red/black analysis naturally collapses to the standard multigrid transfer operators used in practice, but it arises here from an exact algebraic derivation rather than from geometric considerations.

As smoother we use a Jacobi iteration based on the diagonal of A_2 , i.e. $S^{-1} = \text{diag}(A_2)^{-1}$, applied symmetrically with m pre- and m post-smoothing steps as in Algorithm 2.1. For each choice of relaxation vector $\{\alpha_i\}_{i=1}^m$ we explicitly assemble the corresponding two-level error operator E_m defined in (2.1) and compute its spectral radius $\rho(E_m)$.

Figure 1 reports $\rho(E_m)$ as a function of m for three families of relaxation parameters. The gray curves correspond to constant choices $\alpha_i \equiv \alpha$ with $\alpha \in \{0.1, 0.2, \dots, 1.0\}$. The dashed black curve shows the classical choice $\alpha_i \equiv 2/3$ obtained by optimizing the Jacobi smoother alone (see [10]). The thick curve with filled markers uses the relaxation parameters given by Theorem 3.11. Although the theorem itself is stated for a general symmetric 2×2 block partition and does not rely on a purely one-dimensional PDE model, here we *apply* its relaxation parameters in the tensor-product setting described above. The figure shows that this choice produces relaxations which, in the

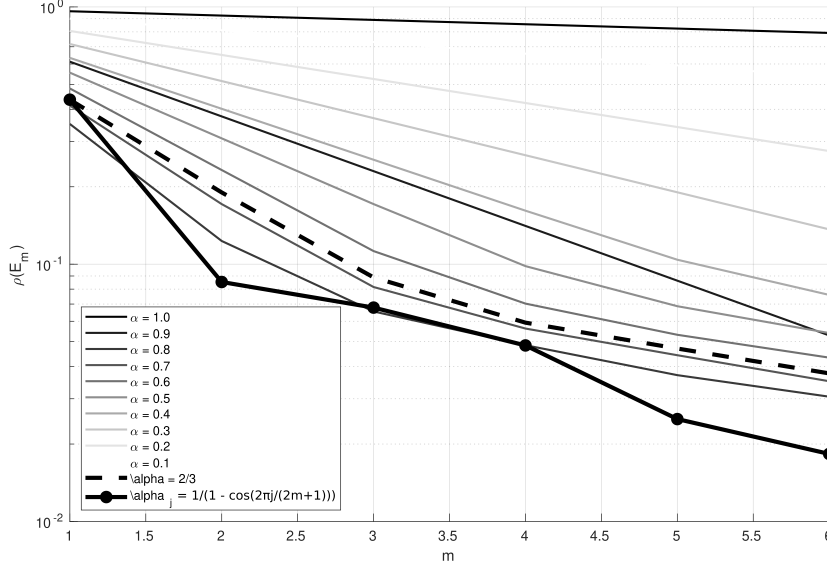


Fig. 1: Spectral radius $\rho(E_m)$ versus the number of smoothing steps m for the two-dimensional Laplacian preconditioned system on a 16×16 interior finite-difference grid. Gray curves: constant $\alpha \in \{0.1, \dots, 1.0\}$; dashed: constant $\alpha = 2/3$; thick curve with filled markers: relaxation parameters given by Theorem 3.11.

two-dimensional finite-difference setting, are very close to the optimal ones in minimizing $\rho(E_m)$ among all constant relaxations. This confirms that the algebraic spectral optimization from Theorem 3.11 extends approximately to higher-dimensional tensor-product discretizations while preserving sparse restriction and prolongation operators.

4.2. Discontinuous Galerkin 1D. To illustrate the generality of the clustering mechanism established in Section 3, we now construct, within the same algebraic framework, a prolongation operator that also yields a two-point clustered spectrum. This experiment is inspired by—but *not identical to*—the discontinuous interpolation introduced in [19].

In [19], the prolongation operator was designed under strict locality and symmetry constraints: it acted *within each DG element* and its sparsity pattern ensured that each fine-grid degree of freedom depended on a *single* coarse element. The interpolation was therefore purely local, and the only adjustable parameter controlling its behavior was the discontinuity parameter c , combined with the penalty parameter δ_0 and the relaxation parameter α . Under these constraints, there existed a *unique* combination of (α, c, δ_0) that produced a perfectly clustered spectrum.

Reproducing those constraints within our present algebraic framework would require restarting the derivation from the invariance relations of Section 2, and would likely yield a different scalar response. Rather than pursuing this restricted case, we find it more illustrative to generalize that result, allowing for a broader class of prolongation operators and thereby uncovering a richer family of clustered spectra within the same theoretical structure.

In the present setting, these geometric constraints are lifted. We no longer prescribe the interpolation geometry but instead derive the prolongation operator directly from the algebraic block structure of the reordered fine-grid matrix. As in Section 4.1, we perform an element-wise red/black permutation so that the operator takes the form

$$(4.3) \quad M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \quad P_{\text{RB}} = \begin{bmatrix} -A^{-1}B \\ I \end{bmatrix}, \quad R_{\text{RB}} = P_{\text{RB}}^\top, \quad M_0 = R_{\text{RB}} M P_{\text{RB}}.$$

Two admissible 1D red/black permutations. Both an *element-wise* permutation (grouping DoFs by alternating elements) and an *interface-wise* permutation (alternating traces across interfaces) are valid. They lead to different *local* algebraic stencils for P and for the Jacobi smoother S^{-1} . In all cases, the smoothing relaxations are the $\{\alpha_i\}_{i=1}^m$ from Theorem 3.11.

Element-wise (local stencils). On each two-element patch, the local prolongation block (fine 8 DoFs from 4 coarse DoFs) and the Jacobi smoother uses 2×2 diagonal blocks are

$$(4.4) \quad P_{\text{loc,el}}(\delta) = \begin{pmatrix} \frac{\delta}{4\delta-2} & \frac{\delta-1}{4\delta-2} & 0 & 0 \\ \frac{\delta-1}{4\delta-2} & \frac{\delta}{4\delta-2} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{\delta}{4\delta-2} & \frac{\delta-1}{4\delta-2} & \frac{\delta}{4\delta-2} & \frac{\delta-1}{4\delta-2} \\ \frac{\delta-1}{4\delta-2} & \frac{\delta}{4\delta-2} & \frac{\delta-1}{4\delta-2} & \frac{\delta}{4\delta-2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad S_{\text{loc,el}}^{-1}(\delta) = \frac{1}{-1+2\delta} \begin{pmatrix} \delta & \delta-1 \\ \delta-1 & \delta \end{pmatrix}.$$

Interface-wise (local stencils). On each interface, the local prolongation block and the Jacobi smoother are

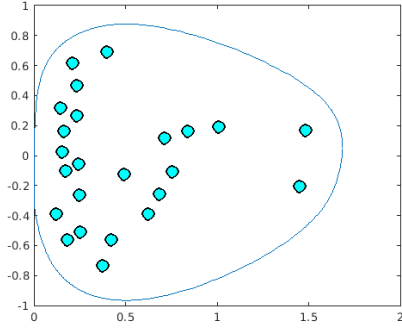
$$(4.5) \quad P_{\text{loc,int}}(\delta) = \begin{pmatrix} \frac{\delta-1}{\delta} & \frac{1}{2\delta} & 0 & 0 \\ \frac{1}{\delta} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{2\delta} & \frac{\delta-1}{\delta} & \frac{1}{2\delta} & 0 \\ 0 & \frac{1}{2\delta} & \frac{\delta-1}{\delta} & \frac{1}{2\delta} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{1}{2\delta} & \frac{\delta-1}{\delta} \end{pmatrix}, \quad S_{\text{loc,int}}^{-1}(\delta) = \frac{1}{\delta} I_2,$$

where $\delta > 0$ denotes the DG penalty parameter (analogous to δ_0 in [19]) and assembly uses the usual overlap; the resulting global P are *not* block diagonal.

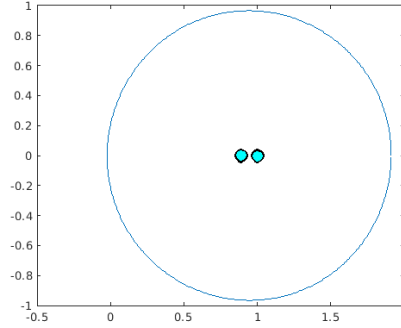
The definition of R_{RB} and M_0 follows directly from the symmetric two-level formulation introduced in Section 2; it is not an additional assumption but a necessary consequence of the formulation. In particular, the theory developed in Section 3 applies for an *arbitrary number of smoothing steps* m , with the relaxation parameters $\{\alpha_j\}_{j=1}^m$ chosen as in Theorem 3.11. Obtaining such a closed-form description of the spectrum for general m would be extremely cumbersome with a classical Local Fourier Analysis approach, which requires a mode-by-mode treatment and becomes rapidly intractable as the number of smoothing steps increases.

Compared with the interpolation from [19], this algebraic prolongation has a richer coupling pattern: some fine-grid degrees of freedom are interpolated from *two neighboring coarse elements* instead of one. This stronger coupling increases the effective range of the operator and removes the need for parameter tuning. The two-point spectral clustering, which in [19] appeared only for a specific triple (α, c, δ_0) , now emerges automatically for *any* choice of δ .

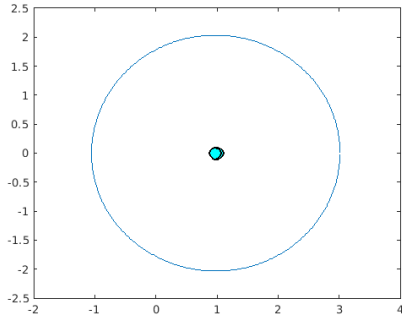
From the DG perspective, this prolongation is again discontinuous: even if the coarse solution is continuous across an interface (that is, the two coarse traces coincide), the mixed rows of $P_{\text{loc}}(\delta)$ generally produce different fine values on the two



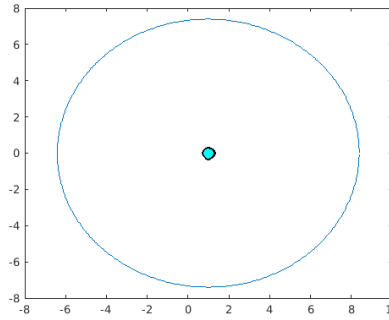
(a) Field of values and spectrum of B .



(b) Preconditioned operator, $m = 1$ smoothing.



(c) Preconditioned operator, $m = 2$ smoothings.



(d) Preconditioned operator, $m = 3$ smoothings.

Fig. 2: Field of values and spectra for the original matrix B and the three preconditioned operators used in the non-normality experiment.

sides of that interface, except for the constant mode. Hence a continuous coarse function is typically mapped to a *discontinuous* fine one. Together with the results of [19], this reinforces the same qualitative conclusion: when constructing efficient multilevel preconditioners for DG discretizations, the answer to the question “*Should multilevel methods for DG discretizations use discontinuous interpolation operators?*” remains **yes**—even when the interpolation arises purely from the algebraic two-level factorization rather than from an explicit geometric design.

This shows that the clustering phenomenon does not depend on the geometric notion of continuity or discontinuity of the interpolation, but follows directly from the algebraic structure of the symmetric two-level formulation itself.

4.3. Nonsymmetric matrices and effect on the field of values. To probe the behaviour of the clustering result in a fully non-symmetric setting, we construct a complex test matrix $B \in \mathbb{C}^{N \times N}$ by combining a Hermitian positive-definite part and a skew-Hermitian part. We draw W and X with independent standard normal

real and imaginary entries, and set $H = W^*W + \eta I$ and $K = \frac{1}{2}(X - X^*)$, so that H is Hermitian positive definite and K is skew-Hermitian. With parameters η and γ chosen to place the field of values of M near the origin while keeping the matrix visibly non-normal, we define

$$(4.6) \quad B = \frac{1}{1000}H + \gamma K.$$

On this matrix we apply GMRES with a single application of Algorithm 2.1 as a preconditioner. Inside Algorithm 2.1, the coarse operator $M_0 = RBP$ is again solved by GMRES using the same two-level construction applied recursively to the 2×2 block structure of M_0 , until the coarsest system is small enough to be inverted directly. Since two coarse iterations are performed at every level, the resulting Krylov scheme has the overall structure of a W-cycle.

For this non-normal test problem, the preconditioned operator retains the spectral collapse predicted by Theorem 3.11, but its field of values does not necessarily shrink. Figures 2a–2d illustrate one such case: although each preconditioned operator has only two eigenvalues, the associated fields of values widen as the number of smoothing steps m increases.

This effect is particularly pronounced when the field of values extends into both the left and right half-planes. In such situations, the preconditioned operator loses the half-plane separation that underlies standard GMRES convergence estimates based on field-of-values bounds. Consequently, problems whose discretizations naturally place the field of values across both half-planes—such as time-harmonic wave equations—must be treated with care when using this recursive preconditioning strategy.

REFERENCES

- [1] N. S. Bakhvalov. On the convergence of a relaxation method with natural constraints on the elliptic operator. *USSR Computational Mathematics and Mathematical Physics*, 6(5):101–135, 1966.
- [2] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, 31(138):333–390, 1977.
- [3] A. Brandt, S. F. McCormick, and J. W. Ruge. Algebraic multigrid (amg) for sparse matrix equations. In D. J. Evans, editor, *Sparsity and its Applications*, pages 257–284. Cambridge University Press, Cambridge, UK, 1985.
- [4] Achi Brandt. Algebraic multigrid theory: The symmetric case. *Applied Mathematics and Computation*, 19(1–4):23–56, 1986.
- [5] J. Brannick, A. Frommer, K. Kahl, S. Maclachlan, and L. Zikatanov. Adaptive reduction-based multigrid for nearly singular and highly disordered physical systems. *ETNA. Electronic Transactions on Numerical Analysis [electronic only]*, 37:276–295, 2010.
- [6] M. Brezina, T. Manteuffel, S. McCormick, J. Ruge, and G. Sanders. Towards adaptive smoothed aggregation (α sa) for nonsymmetric problems. *SIAM Journal on Scientific Computing*, 32(1):14–39, 2010.
- [7] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial*. SIAM, Philadelphia, 2nd edition, 2000.
- [8] B. L. Buzbee, G. H. Golub, and C. W. Nielson. On direct methods for solving poisson’s equations. *SIAM Journal on Numerical Analysis*, 7(4):627–656, 1970.
- [9] Tony F. Chan and W.L. Wan. Robust multigrid methods for nonsmooth coefficient elliptic linear systems. *Journal of Computational and Applied Mathematics*, 123(1):323–352, 2000. Numerical Analysis 2000. Vol. III: Linear Algebra.
- [10] Oliver G. Ernst and Martin J. Gander. Multigrid methods for helmholtz problems: A convergent scheme in 1d using standard components. In Ivan Graham, Ulrich Langer, Jens M. Melenk, and Mourad Sini, editors, *Direct and Inverse Problems in Wave Propagation and Applications*, volume 14 of *Radon Series on Computational and Applied Mathematics*, pages 135–186. De Gruyter, Berlin, Boston, 2013.
- [11] R. D. Falgout and P. S. Vassilevski. On generalizing the algebraic multigrid framework. *SIAM Journal on Scientific Computing*, 27(4):1233–1259, 2004.

- [12] R. P. Fedorenko. A relaxation method for solving elliptic difference equations. *USSR Computational Mathematics and Mathematical Physics*, 1:109–128, 1962.
- [13] R. P. Fedorenko. The speed of convergence of one iterative process. *USSR Computational Mathematics and Mathematical Physics*, 4(3):227–235, 1964.
- [14] Hartmut Foerster, Klaus Stüben, and Ulrich Trottenberg. Non-standard multigrid techniques using checkered relaxation and intermediate grids. In Martin H. Schultz, editor, *Elliptic Problem Solvers*, pages 285–300. Academic Press, 1981.
- [15] I. S. Gradshteyn and I. M. Ryzhik. *Table of Integrals, Series, and Products*. Academic Press, Amsterdam, 7 edition, 2007. Edited by Alan Jeffrey and Daniel Zwillinger.
- [16] W. Hackbusch. *Multi-Grid Methods and Applications*, volume 4 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1985.
- [17] José Pablo Lucero Lorca, Duane Rosenberg, Isidora Jankov, Conor McCoid, and Martin Jakob Gander. On an efficient line smoother for the p-multigrid γ -cycle. *arXiv preprint arXiv:2504.10710*, April 2025. preprint.
- [18] José Pablo Lucero Lorca. Towards a multigrid preconditioner interpretation of Hierarchical Poincaré-Steklov solvers. *arXiv*, 2025.
- [19] José Pablo Lucero Lorca and Martin J. Gander. Should multilevel methods for discontinuous galerkin discretizations use discontinuous interpolation operators? In Susanne C. Brenner, Eric Chung, Axel Klawonn, Felix Kwok, Jinchao Xu, and Jun Zou, editors, *Domain Decomposition Methods in Science and Engineering XXVI*, pages 273–280, Cham, 2022. Springer International Publishing.
- [20] Scott MacLachlan, Tom Manteuffel, and Steve McCormick. Adaptive reduction-based amg. *Numerical Linear Algebra with Applications*, 13(8):599–620, 2006.
- [21] Thomas A. Manteuffel, Steffen Münzenmaier, John Ruge, and Ben Southworth. Nonsymmetric reduction-based algebraic multigrid. *SIAM Journal on Scientific Computing*, 41(5):S242–S268, 2019.
- [22] Thomas A. Manteuffel, John Ruge, and Ben S. Southworth. Nonsymmetric algebraic multigrid based on local approximate ideal restriction (ℓ air). *SIAM Journal on Scientific Computing*, 40(6):A4105–A4130, 2018.
- [23] Luke N. Olson, Jacob B. Schroder, and Raymond S. Tuminaro. A general interpolation strategy for algebraic multigrid using energy minimization. *SIAM Journal on Scientific Computing*, 33(2):966–991, 2011.
- [24] Manfred Ries, Ulrich Trottenberg, and Gerd Winter. A note on mgr methods. *Linear Algebra and its Applications*, 49:1–26, 1983.
- [25] J. W. Ruge and K. Stüben. Algebraic multigrid. In S. F. McCormick, editor, *Multigrid Methods, Frontiers in Applied Mathematics, Vol. 3*, pages 73–130. SIAM, Philadelphia, PA, USA, 1987.
- [26] Marzio Sala and Raymond S. Tuminaro. A new petrov–galerkin smoothed aggregation preconditioner for nonsymmetric linear systems. *SIAM Journal on Scientific Computing*, 31(1):143–166, 2008.
- [27] P. Wesseling. *An Introduction to Multigrid Methods*. John Wiley & Sons, Chichester, 1992.
- [28] T. A. Wiesner, R. S. Tuminaro, W. A. Wall, and M. W. Gee. Multigrid transfers for nonsymmetric systems based on schur complements and galerkin projections. *Numerical Linear Algebra with Applications*, 21(3):415–438, 2014.
- [29] Xuefeng Xu and Chen-Song Zhang. On the ideal interpolation operator in algebraic multigrid methods. *SIAM Journal on Numerical Analysis*, 56(3):1693–1710, 2018.