

Local Fourier analysis for multigrid with overlapping smoothers applied to systems of PDEs

Scott P. MacLachlan^{1,*},[†],[‡] and C. W. Oosterlee^{2,3}

¹Department of Mathematics, Tufts University, 503 Boston Avenue, Medford, MA 02155, U.S.A.

²Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands

³CWI, National Research Institute for Mathematics and Computer Science, Amsterdam, The Netherlands

SUMMARY

Since their popularization in the late 1970s and early 1980s, multigrid methods have been a central tool in the numerical solution of the linear and nonlinear systems that arise from the discretization of many PDEs. In this paper, we present a local Fourier analysis (LFA, or local mode analysis) framework for analyzing the complementarity between relaxation and coarse-grid correction within multigrid solvers for systems of PDEs. Important features of this analysis framework include the treatment of arbitrary finite-element approximation subspaces, leading to discretizations with staggered grids, and overlapping multiplicative Schwarz smoothers. The resulting tools are demonstrated for the Stokes, curl–curl, and grad–div equations. Copyright © 2011 John Wiley & Sons, Ltd.

Received 21 April 2010; Revised 27 September 2010; Accepted 17 October 2010

KEY WORDS: multigrid; finite-element discretizations; local Fourier analysis

1. INTRODUCTION

First envisioned as a technique for solving Poisson-type problems with optimal complexity, multigrid and other multilevel algorithms have become the methods of choice for solving the matrix equations that arise in a wide variety of applications. In this paper, we are concerned with the design and analysis of multigrid algorithms for the solution of discretized systems of partial differential equations. While general principles exist that can aid in developing multigrid techniques for an application area, the optimal choices for its components are often difficult to determine beforehand. Many approaches to multigrid theory have been investigated in the last 30 years (see, for example, [1–4]); among these, the technique of local Fourier analysis (LFA, or local mode analysis), first introduced in [5], has remained very successful, providing accurate predictions of performance for a variety of problems, including systems of PDEs.

The primary advantage of LFA is that it allows quantitative prediction of multigrid convergence factors under reasonable assumptions. The word, *local*, in LFA indicates a focus on the character of an operator in the interior of its domain, where it is assumed to be represented by a constant discretization stencil. The Fourier symbols of such operators can easily be computed. A further insight in [6] was that all of the components in a multigrid method can be analyzed in this fashion, leading to a block-diagonal representation in a Fourier basis. LFA helped, for example, in

*Correspondence to: Scott P. MacLachlan, Department of Mathematics, Tufts University, 503 Boston Avenue, Medford, MA 02155, U.S.A.

[†]E-mail: scott.maclachlan@tufts.edu

[‡]Formerly at TU-Delft and CWI.

the understanding of SOR as a smoother for moderately anisotropic and high-dimensional problems [7] and the solution of the biharmonic equation with efficiency similar to that of Poisson [8]. Recent advances in LFA include LFA for high-dimensional problems [9], for multigrid as a preconditioner [10], for triangular and hexagonal meshes [11, 12], optimal control problems [13], and discontinuous Galerkin discretizations [14]. The LFA monograph and related software of Wienands and Joppich [8] focus on LFA for collocated discretizations, providing an excellent tool for experimenting with Fourier analysis.

In this paper, we present a framework for performing the LFA for state-of-the-art finite-element (FE) discretizations of systems of PDEs. In particular, we show that the LFA ansatz is still valid when using overlapping multiplicative smoothers, such as the one proposed in [15], for the grad-div and curl-curl equations, and in [16, 17] for the Stokes equations. Analysis of the additive versions of these smoothers was conducted in [18, 19]; however, this form of analysis does not extend to cover the multiplicative case. LFA for overlapping multiplicative smoothers has been, to our knowledge, performed in only two cases, for the staggered finite-difference discretization of the Stokes and Navier-Stokes equations [20], and for a mixed FE discretization of Poisson's equation [21]. We also apply this analysis to the well-known multigrid solvers for the grad-div, curl-curl, and Stokes equations, providing quantitative predictions of the performance of multigrid methods based on these smoothers, in contrast to the non-predictive proofs of convergence offered in [15, 22]. In the case of the Stokes equations, in particular, quantitative estimates have been notably missing from the literature [23].

The remainder of this paper is organized as follows. First, in Section 2, we provide some background on the motivating PDE systems for this work. LFA smoothing analysis is discussed in Section 3, with a focus on the treatment of overlapping multiplicative smoothers. A detailed example is presented in Section 4. Section 5 presents two-grid LFA, focusing on the issue of multigrid grid transfers for staggered discretizations. Finally, Section 6 presents the application of these techniques to appropriate discretizations of the Stokes, grad-div, and curl-curl equations. There, we focus on the impact of the choice of transfer operators and on the choice of smoother and under-relaxation parameters on the two-grid LFA convergence.

2. MULTIGRID AND FES FOR PDE SYSTEMS

The discretization of systems of PDEs must be done with care, to avoid the introduction of unstable modes in the resulting discrete system. For FE discretizations, this typically results in choosing different FE subspaces for different components of the system, to satisfy known inf-sup conditions, leading to the use of Raviart-Thomas, Nédélec, or Taylor-Hood elements, for example. For a thorough treatment of these issues in the FE context, see [24, 25].

We shall refer to the FE discretizations that we treat here collectively as 'staggered discretizations', indicating that the nodes associated with the discrete degrees of freedom are not aligned on the same grid for each component of the PDE system. The techniques developed here are applicable to arbitrary staggered discretizations of systems of PDEs, including the 'trivial' case of a collocated discretization.

2.1. Multigrid for systems of equations

Many standard discretizations of systems of PDEs (including those described below) do not guarantee that the resulting matrices are diagonally dominant (or even that they are definite) either because of the properties of the continuum operators themselves or because of necessary constraints on the discretizations. In these cases, expensive relaxation techniques may be used to reestablish effective multigrid convergence. Unfortunately, these relaxation techniques are no longer algebraic black boxes, such as the Jacobi and Gauss-Seidel iterations. Instead, the details of these techniques are determined by those of the underlying PDEs.

A first indication for the appropriate choice of relaxation method for a system of equations can be derived from the system's determinant. Interestingly, the determinant of the discrete operator

may also give us valuable information about the stability of the discretizations used for a system. The direct relation between effectiveness of smoothing and the determinant of the discrete system is by means of the h -ellipticity concept [26, 27]. For unstable discretizations, which give rise to unphysical oscillations in the numerical solutions, there is no chance that we can set up efficient local, i.e. pointwise, smoothing methods.

An obvious choice in the case of strong off-diagonal operators in the differential system (also indicated by the determinant) is collective smoothing: all unknowns in the system at a certain grid point or grid cell are updated simultaneously. While the use of these smoothers leads to efficient multigrid approaches for systems of PDEs, collective relaxation is not the only possible approach. The main alternative is the use of distributive smoothers [26, 28, 29], which take their name from a distribution operation; the discrete (or continuum) equations are transformed by right matrix multiplication into a block triangular matrix that is amenable to pointwise relaxation. Simple pointwise relaxation is performed on this block triangular system and, then, the resulting update is distributed (based on the transformation matrix) back to the original matrix problem.

While distributed smoothers are often found to be more efficient than overlapping smoothers [29], their applicability is limited by the need to find an effective distribution matrix; this is often difficult to do for problems with unstructured grids or variable coefficients. Thus, distributed relaxation may be difficult to implement in a general and purely algebraic fashion, as would be necessary for use within an algebraic multigrid iteration. Furthermore, the proper treatment of boundary conditions in distributive relaxation may not be trivial, as typically the operator of the preconditioned system is of higher order than the original operator, thus requiring additional (possible non-physical) boundary conditions within smoothing. Substantial effort has however been put into successfully extending the distributed smoother of Hiptmair [28] to algebraic multigrid algorithms for the curl–curl equation [30–32]. In this paper, we focus exclusively on collective relaxation approaches.

2.2. Discretizing systems of PDEs

As a first pair of examples, we consider the gradient–divergence (grad–div) and curl–curl equations,

$$-\nabla(a\nabla\cdot\mathcal{U})+\mathcal{U}=\mathcal{F} \quad \text{in } \Omega, \quad (1)$$

and

$$\nabla\times(a\nabla\times\mathcal{U})+\mathcal{U}=\mathcal{F} \quad \text{in } \Omega, \quad (2)$$

with parameter $a>0$, where Ω is an open domain in \mathbb{R}^d . These operators appear frequently in the formulation of mathematical models in physics and engineering, particularly for problems related to electro-magnetics or fluid and solid mechanics (see, for example, [15, 29, 33] for more details).

In the FE framework, face elements, such as the Raviart–Thomas elements, have been proposed for accurate discretization of (1) [34], whereas edge elements, such as the Nédélec elements [35], are commonly used for (2), see Figure 1. The difficulty in achieving efficient multigrid treatment of the resulting discrete linear systems comes from the fact that the eigenspace associated with the minimal eigenvalue of the discrete operator contains many eigenvectors (for large enough parameter a). For Equation (1), this arises because any divergence-free vector is an eigenvector corresponding to this minimal eigenvalue, while a similar difficulty occurs with curl-free vectors in (2). In both cases, these components can be arbitrarily oscillatory and can neither be reduced by standard (pointwise) smoothing procedures, nor be well represented on coarse grids [15].

A remedy for (1) proposed in [36] builds upon local div-free functions and their orthogonal complements in the FE space. In [37], a multigrid preconditioner was presented for a discretization with the lowest-order Raviart–Thomas FE spaces on triangles. A distributive smoothing technique in multigrid to handle the troublesome div-free components was proposed in [38]. These techniques were extended for the curl–curl equations in [15, 33]. Here, we will quantitatively analyze the multiplicative collective smoother introduced in [15], known as the AFW smoother. This smoother can be motivated by thinking about the different treatment given by the grad–div or curl–curl operator to components of \mathcal{U} that look like gradients and those that look like curls. As the dominant

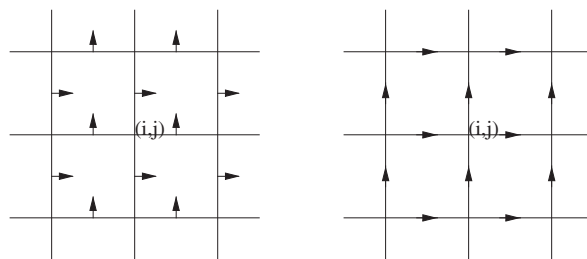


Figure 1. Placement of unknowns with Raviart–Thomas FE for grad–div operator (left), and Nédélec edge elements for curl–curl (right).

part of these operators does not act on one component of the solution, it is important that the relaxation technique accurately resolves these components on all scales.

The Stokes equations are central to the simulation of certain viscous fluid-flow problems. They are represented by a saddle point problem,

$$-\Delta \mathcal{U} + \nabla \mathcal{P} = \mathcal{F}, \quad (3)$$

$$\nabla \cdot \mathcal{U} = 0, \quad (4)$$

for velocity vector, \mathcal{U} , and scalar pressure, \mathcal{P} , of a viscous fluid. The weak form of the Stokes equations is found by multiplying by test functions, \mathcal{V} and \mathcal{Q} , and integrating by parts. Writing this system in terms of the bilinear forms, $a_{11}(\mathcal{U}, \mathcal{V}) = \int_{\Omega} \sum_{i=1}^d (\nabla \mathcal{U}_i) \cdot (\nabla \mathcal{V}_i) d\Omega$ and $a_{12}(\mathcal{V}, \mathcal{P}) = -\int_{\Omega} (\nabla \cdot \mathcal{V}) \mathcal{P} d\Omega$, we have

$$a_{11}(\mathcal{U}, \mathcal{V}) + a_{12}(\mathcal{V}, \mathcal{P}) = \int_{\Omega} \mathcal{F} \cdot \mathcal{V} d\Omega \quad (5)$$

$$a_{21}(\mathcal{U}, \mathcal{Q}) = 0, \quad (6)$$

with $a_{12}(\cdot, \cdot) = a_{21}(\cdot, \cdot)$, and vector-valued functions $\mathcal{U}, \mathcal{V}: \mathbb{R}^d \rightarrow \mathbb{R}^d$. Notice that \mathcal{U} and \mathcal{P} lie in different spaces; typically, $\mathcal{U} \in V \subset (H^1(\Omega))^d$, whereas $\mathcal{P} \in W \subset L^2(\Omega)$. In proving uniqueness of the pressure component of the solution, \mathcal{P} , a natural condition [24, 25, 39] arises,

$$\inf_{\mathcal{P} \in W} \sup_{\mathcal{V} \in V} \frac{a_{12}(\mathcal{V}, \mathcal{P})}{\|\mathcal{P}\| \|\mathcal{V}\|} = \beta > 0. \quad (7)$$

This condition is known by many names, including the Ladyzhenskaya–Babuška–Brezzi (or LBB) condition and the inf–sup condition.

Similar considerations apply to the discrete problem attained by restricting the functions to finite-dimensional subspaces $\mathcal{U}_h, \mathcal{V}_h \in V_h$ and $\mathcal{P}_h, \mathcal{Q}_h \in W_h$, leading to a discrete version of the inf–sup condition. A natural discretization, representing both \mathcal{U}_h and \mathcal{P}_h with bilinear basis functions, does not satisfy the necessary inf–sup condition [40] and, hence, we are forced to consider higher-order basis functions for Equations (3) and (4), such as the Taylor–Hood elements [24, 41] where \mathcal{U}_h is represented by biquadratic basis functions and \mathcal{P}_h is represented by bilinears.

The development of efficient smoothers for the Stokes equations was originally performed in the staggered finite-differences setting. There, the concepts of collective and distributive relaxation were developed by Vanka [16] and Brandt and Dinar [26, 42], respectively. These smoothers were later accompanied by quantitative analysis, based on LFA. For FE discretizations, work on efficient smoothers for the Stokes and Navier–Stokes equations includes that by Braess and Sarazin [43], which is based on an approximate factorization of (3) and (4), as well as that by John and others [17, 23, 44, 45], which focuses on a variety of smoothers including those of collective (Vanka) type.

It was the FE setting especially that drove the rapid development of the algebraic multigrid method in the nineties (of the last century), with the recognition of its impressive efficiency, often for completely unstructured meshes. Quantitative theory for methods on these meshes is not

available, unfortunately. To bridge the gap between the fully understood case of finite differences on structured grids and the case of FEs on completely unstructured grids, we develop here quantitative analysis for FE discretizations on structured quadrilateral meshes in 2D, still leading to stencil-based discretizations.

3. ANALYSIS OF RELAXATION WITH LFA

A quantitative, predictive theoretical framework, such as LFA, allows significant algorithmic development independent of an implementation. Here and in Section 5, we review the ideas behind two-grid LFA; first, we focus on the analysis of the smoothing step in Fourier space. We consider the solution of a linear system of equations, $A_h \mathbf{u}_h = \mathbf{f}_h$, where the subscript, h , serves to remind us that the origins of matrix A_h are in the discretization of a PDE on a uniform quadrilateral grid with mesh size h (or, possibly, with mesh sizes $\mathbf{h} = (h_x, h_y, \dots)^T$ that are not uniform across dimensions).

Given an approximation, \mathbf{v}_h , to the solution of $A_h \mathbf{u}_h = \mathbf{f}_h$, the residual equation relates the error, $\mathbf{e}_h = \mathbf{u}_h - \mathbf{v}_h$, in that approximation to the residual, $\mathbf{r}_h = \mathbf{f}_h - A_h \mathbf{v}_h$, as $A_h \mathbf{e}_h = \mathbf{r}_h$. Thus, for a given approximation, \mathbf{v}_h , we can express the true solution as $\mathbf{u}_h = \mathbf{v}_h + A_h^{-1} \mathbf{r}_h$. Choosing M_h to be an approximation to A_h that is easily inverted leads to an update iteration that can be analyzed in terms of its error-propagation operator

$$\mathbf{e}_h \leftarrow (I - M_h^{-1} A_h) \mathbf{e}_h.$$

A complete analysis of the convergence properties of the error-propagation operator arises in terms of its eigenvectors, $\{\phi^{(j)}\}$, and eigenvalues, $\{\lambda_j\}$. Any initial error, $\mathbf{e}_h^{(0)}$, can then be expanded into the basis given by the eigenvectors of $I - M_h^{-1} A_h$, and the error after k iterations of relaxation is given by $\mathbf{e}_h^{(k)} = \sum_j \sigma_j \lambda_j^k \phi^{(j)}$, where the coefficients, $\{\sigma_j\}$, are defined so that the expansion is valid for the initial error, $\mathbf{e}_h^{(0)}$. The effectiveness of the relaxation on the component of the error in the direction of a given eigenvector, $\phi^{(j)}$, is then given simply by the eigenvalue, λ_j . If λ_j is small (e.g. $|\lambda_j| \leq 0.5$), errors in the direction of $\phi^{(j)}$ are quickly attenuated by the iteration. For large λ_j , such that $|\lambda_j| \approx 1$, the errors in the direction of $\phi^{(j)}$ are slow to be reduced and, after a few steps of the iteration, these errors will dominate the remaining difference between \mathbf{u}_h and \mathbf{v}_h .

Finding the eigenvectors and eigenvalues of $I - M_h^{-1} A_h$ for this analysis can be quite difficult, depending on the matrices, A_h and M_h . For general matrices there may be little relation between the eigenvectors and eigenvalues of A_h and those of relaxation, unless A_h and M_h are assumed to have more structure than is typically expected, such as being circulant. Such structure is strongly affected by boundary conditions on the PDE, while the rows of the matrix corresponding to degrees of freedom in the interior of the PDE domain may have a natural Toeplitz or multilevel Toeplitz structure (representing a discrete PDE on a structured grid); imposition of boundary conditions usually results in a set of rows that have quite different values. The key idea behind LFA is to ignore the effect of these boundary conditions, by extending the operator and relaxation stencils from the interior of the domain to uniform infinite grids. On these grids, the discretized scalar differential operators and the operators that define the relaxation stencils become *infinite-grid (multilevel) Toeplitz matrices*, constant-stencil operators that map from one infinite sequence to another, where these sequences represent functions discretized on the infinite grid. Any infinite-grid (multilevel) Toeplitz matrix is diagonalized by the matrix of Fourier modes, Φ_h , where we index the columns of Φ_h by a continuous index, $\theta \in (-\pi/2, 3\pi/2]^d$, and the rows by their spatial location, \mathbf{x} , and write $\phi_h(\mathbf{x}, \theta) = e^{i\theta \cdot \mathbf{x}/h}$ for $i = \sqrt{-1}$; when $d = 1$, this change of basis is known as the *discrete-time Fourier transform* [46]. In this setting, LFA has provided effective predictions on the performance of multigrid cycles based on many common smoothers, including Gauss–Seidel [27], SOR [7], and ILU [47].

LFA for systems of PDEs is based on a simple extension of the assumptions of LFA for scalar PDEs. In the systems case, we assume that the matrix, A_h , is now a block matrix, where each block

is an infinite-grid (multilevel) Toeplitz matrix. Under this assumption, each block in A_h may be diagonalized by left and right transformations with Fourier matrices, Φ_h , although possibly using different nodal coordinates on the left and right for the off-diagonal blocks.

3.1. LFA for overlapping smoothers

Here, we focus on the LFA of *overlapping coupled multiplicative smoothers*. Overlapping smoothers require only knowledge of the element structure, which may be easily retained on coarse scale through element agglomeration or AMGe techniques [48–51].

We identify a collective relaxation scheme as the one that partitions the degrees of freedom of A_h into regular subsets, $S_{i,j}$, whose union provides a cover for the set of degrees of freedom. By saying that these subsets are regular, we mean that there is a one-to-one correspondence between the degrees of freedom in any two subsets, $S_{i,j}$ and $S_{k,l}$; each subset has the same size, and the same number of each ‘type’ of degree of freedom that comes from discretizing different unknowns of the continuum system, in the same (relative) geometric location. The partitioning need not be disjoint; an overlapping coupled smoother occurs when some collection of the degrees of freedom appears in multiple subsets, typically associated with some adjacent indices. In collective relaxation, updates are computed (sequentially or in parallel) by solving the local system (or another non-singular auxiliary system) associated with each subset, $S_{i,j}$, with the most recent residual restricted to $S_{i,j}$ as a right-hand side.

If the subsets $S_{i,j}$ are mutually disjoint, then a collective relaxation scheme is simply a block-wise Jacobi or Gauss–Seidel scheme and can be analyzed as such. If the blocks overlap, however, so that certain degrees of freedom are updated multiple times over the course of a single sweep of relaxation, classical LFA techniques fail. In a relatively unknown paper [20], Sivaloganathan analyzed the Vanka smoother [16], a multiplicative form of overlapping collective relaxation for the staggered finite-difference discretization of the Stokes equation; unfortunately, this paper includes several misprints, which make the results difficult to appreciate. Independently, Molenaar analyzed a similar collective smoother for a mixed FE discretization of Poisson’s equation [21]. Two important questions are, however, left unanswered in [20, 21]: whether the Fourier ansatz is justified for coupled overlapping smoothers and whether these techniques can be generalized for other PDE problems and discretizations.

LFA for non-overlapping relaxation succeeds because, in the infinite-grid (multilevel) Toeplitz setting, the matrix, A_h , is split into two (multilevel) Toeplitz pieces, $A_h = M_h - N_h$, where M_h and N_h are also both (multilevel) Toeplitz. Thus, all three matrices (and, in particular, the error-propagation operator, $M_h^{-1}N_h$) are diagonalized by a similarity transformation with the Fourier matrix, Φ_h (or, in the case of systems, a block-matrix consisting of disjoint Fourier matrices). It is not apparent that the same is true for overlapping relaxations, because the error-propagation operator is not easily written in terms of a matrix splitting.

To illustrate, we consider the (most common) case of cell-wise relaxation; for each node, (i, j) , associated with the grid- h mesh, we define a cell of size $h \times h$ adjacent to, or, including node (i, j) , and simultaneously solve for updates to all degrees of freedom that fall within or on the boundary of this cell, see Figure 2. Relaxation is then defined in a lexicographical Gauss–Seidel manner, sequentially solving for the unknowns associated with cell (i, j) , going first across the mesh from left to right, then up the mesh. Note that, using this definition of the collections, $S_{i,j}$, each degree of freedom located at the corner of cell (i, j) is included in four subsets, whereas those on the edges of cell (i, j) are included in two subsets, and degrees of freedom in the interior of a cell are included in only the subset corresponding to the cell. By a similar count, if there are k degrees of freedom at each cell corner, l_x and l_y degrees of freedom along the x and y edges of a cell, respectively, and m interior degrees of freedom, then $4k + 2(l_x + l_y) + m$ degrees of freedom are included in the subset, $S_{i,j}$.

Considering the (hypothetical) elements in Figure 2, two possible definitions of the subsets, $S_{i,j}$, are highlighted. One possibility, using the element boundaries (solid lines) to define the cells yields subsets that overlap both at the corners of the elements and along one of the element boundaries, while there is a unique interior node in each subset that belongs only to $S_{i,j}$. Another possibility,

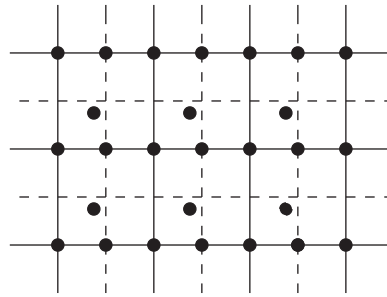


Figure 2. Partitioning of degrees of freedom into overlapping subsets based on cells.

using the dual-element boundaries (marked by the dashed lines), has overlap only at the element edges, with two interior nodes for each subset, $S_{i,j}$.

In order to use the Fourier ansatz, that the error-propagation operator for coupled (overlapping) relaxation is block-diagonalized by the Fourier matrix, Φ_h , we need to know that this is true, regardless of the distribution of degrees of freedom within $S_{i,j}$. The following result is the key step in proving this, showing the inductive step that if the errors before relaxation on collection $S_{i,j}$ satisfy a generalized LFA ansatz, then the errors after relaxation on $S_{i,j}$ satisfy the same ansatz advanced by one cell.

Theorem 1

Assume that A_h is a block-matrix with infinite-grid multilevel Toeplitz blocks, corresponding to the discretization of a two-dimensional PDE on a regular grid with mesh size, h , and let k index the variables within the collections, $S_{i,j}$, of variables to be updated simultaneously. Let the initial error (before the beginning of the relaxation sweep) for each unknown, $U^{(k)}$, be given by a single Fourier frequency θ , which is independent of k , $\mathcal{E}_{i,j}^{(k)} = \alpha^{(k)} e^{i\theta \cdot \mathbf{x}_{i,j}^{(k)}/h}$, where $\mathbf{x}_{i,j}^{(k)}$ is the location of the discrete node corresponding to unknown $U^{(k)}$ associated with relaxation subset $S_{i,j}$. Let the update for the degrees of freedom in each subset $S_{i,j}$ be calculated as

$$U_{i,j}^{\text{new}} = U_{i,j}^{\text{old}} + B^{-1} \mathcal{R}_{i,j}^{\text{old}},$$

where $\mathcal{R}_{i,j}^{\text{old}}$ is the residual at the nodes in $S_{i,j}$ evaluated before these unknowns are updated by the relaxation for cell $S_{i,j}$, $U_{i,j}^{\text{old}}$ and $U_{i,j}^{\text{new}}$ are the approximations to $U_{i,j}$ before and after the relaxation sweep, and B is some non-singular approximation of $A_{i,j}$, the diagonal block of A_h corresponding to the subset $S_{i,j}$.

Consider a partial lexicographical relaxation sweep, at the stage where the correction to cell $S_{i,j}$ is to be computed. Suppose that, for all degrees of freedom, k_n , located at nodes of the cells, $S_{\ell,m}$ (so that $\mathbf{x}_{\ell,m}^{(k_n)}$ is the lower-left corner of the cell associated with $S_{\ell,m}$), the once-corrected, twice-corrected, three-times-corrected, and four-times-corrected errors satisfy

$$\mathcal{E}_{\ell,m}^{(k_n,1)} = \alpha^{(k_n,1)} e^{i\theta \cdot \mathbf{x}_{\ell,m}^{(k_n)}/h} \quad \text{for } m \leq j \quad \text{or } m = j+1 \quad \text{and } \ell \leq i,$$

$$\mathcal{E}_{\ell,m}^{(k_n,2)} = \alpha^{(k_n,2)} e^{i\theta \cdot \mathbf{x}_{\ell,m}^{(k_n)}/h} \quad \text{for } m \leq j \quad \text{or } m = j+1 \quad \text{and } \ell < i,$$

$$\mathcal{E}_{\ell,m}^{(k_n,3)} = \alpha^{(k_n,3)} e^{i\theta \cdot \mathbf{x}_{\ell,m}^{(k_n)}/h} \quad \text{for } m < j \quad \text{or } m = j \quad \text{and } \ell \leq i,$$

$$\mathcal{E}_{\ell,m}^{(k_n,4)} = \alpha^{(k_n,4)} e^{i\theta \cdot \mathbf{x}_{\ell,m}^{(k_n)}/h} \quad \text{for } m < j \quad \text{or } m = j \quad \text{and } \ell < i.$$

Further, suppose that for all degrees of freedom, k_h , located on the horizontal edges of the cells, $S_{\ell,m}$ (so that $\mathbf{x}_{\ell,m}^{(k_h)}$ lies on the bottom edge of the cell associated with $S_{\ell,m}$), the once-corrected

and twice-corrected errors satisfy

$$\begin{aligned} \mathcal{E}_{\ell,m}^{(k_h,1)} &= \alpha^{(k_h,1)} e^{i\theta \cdot \mathbf{x}_{\ell,m}^{(k_h)}/h} \quad \text{for } m \leq j \quad \text{or } m = j+1 \quad \text{and } \ell < i, \\ \mathcal{E}_{\ell,m}^{(k_h,2)} &= \alpha^{(k_h,2)} e^{i\theta \cdot \mathbf{x}_{\ell,m}^{(k_h)}/h} \quad \text{for } m < j \quad \text{or } m = j \quad \text{and } \ell < i. \end{aligned}$$

Similarly, suppose that for all degrees of freedom, k_v , located on the vertical edges of the cells, $S_{\ell,m}$ (so that $\mathbf{x}_{\ell,m}^{(k_v)}$ lies on the left edge of the cell associated with $S_{\ell,m}$), the once-corrected and twice-corrected errors satisfy

$$\begin{aligned} \mathcal{E}_{\ell,m}^{(k_v,1)} &= \alpha^{(k_v,1)} e^{i\theta \cdot \mathbf{x}_{\ell,m}^{(k_v)}/h} \quad \text{for } m < j \quad \text{or } m = j \quad \text{and } \ell \leq i, \\ \mathcal{E}_{\ell,m}^{(k_v,2)} &= \alpha^{(k_v,2)} e^{i\theta \cdot \mathbf{x}_{\ell,m}^{(k_v)}/h} \quad \text{for } m < j \quad \text{or } m = j \quad \text{and } \ell < i. \end{aligned}$$

Finally, suppose that for all degrees of freedom, k_i , located strictly in the interiors of the cells, $S_{\ell,m}$, the once-corrected errors satisfy

$$\mathcal{E}_{\ell,m}^{(k_i,1)} = \alpha^{(k_i,1)} e^{i\theta \cdot \mathbf{x}_{\ell,m}^{(k_i)}/h} \quad \text{for } m < j \quad \text{or } m = j \quad \text{and } \ell < i.$$

Then, after the corrections have been computed for the degrees of freedom in $S_{i,j}$,

$$\begin{aligned} \mathcal{E}_{i+1,j+1}^{(k_n,1)} &= \alpha^{(k_n,1)} e^{i\theta \cdot \mathbf{x}_{i+1,j+1}^{(k_n)}/h}, & \mathcal{E}_{i,j+1}^{(k_n,2)} &= \alpha^{(k_n,2)} e^{i\theta \cdot \mathbf{x}_{i,j+1}^{(k_n)}/h}, \\ \mathcal{E}_{i+1,j}^{(k_n,3)} &= \alpha^{(k_n,3)} e^{i\theta \cdot \mathbf{x}_{i+1,j}^{(k_n)}/h}, & \mathcal{E}_{i,j}^{(k_n,4)} &= \alpha^{(k_n,4)} e^{i\theta \cdot \mathbf{x}_{i,j}^{(k_n)}/h}, \\ \mathcal{E}_{i,j+1}^{(k_h,1)} &= \alpha^{(k_h,1)} e^{i\theta \cdot \mathbf{x}_{i,j+1}^{(k_h)}/h}, & \mathcal{E}_{i,j}^{(k_h,2)} &= \alpha^{(k_h,2)} e^{i\theta \cdot \mathbf{x}_{i,j}^{(k_h)}/h}, \\ \mathcal{E}_{i+1,j}^{(k_v,1)} &= \alpha^{(k_v,1)} e^{i\theta \cdot \mathbf{x}_{i+1,j}^{(k_v)}/h}, & \mathcal{E}_{i,j}^{(k_v,2)} &= \alpha^{(k_v,2)} e^{i\theta \cdot \mathbf{x}_{i,j}^{(k_v)}/h}, \\ \mathcal{E}_{i,j}^{(k_i,1)} &= \alpha^{(k_i,1)} e^{i\theta \cdot \mathbf{x}_{i,j}^{(k_i)}/h}. \end{aligned}$$

Proof

Consider, first, the update equation

$$B(\mathcal{U}_{i,j}^{\text{new}} - \mathcal{U}_{i,j}^{\text{old}}) = \mathcal{R}_{i,j}^{\text{old}}. \tag{8}$$

Since $\mathcal{E}_{i,j}^{\text{new}} = \mathcal{U}_{i,j} - \mathcal{U}_{i,j}^{\text{new}}$ and $\mathcal{E}_{i,j}^{\text{old}} = \mathcal{U}_{i,j} - \mathcal{U}_{i,j}^{\text{old}}$, we can easily rewrite (8) in terms of $\mathcal{E}_{i,j}^{\text{old}}$ and $\mathcal{E}_{i,j}^{\text{new}}$ as

$$B(\mathcal{E}_{i,j}^{\text{old}} - \mathcal{E}_{i,j}^{\text{new}}) = \mathcal{R}_{i,j}^{\text{old}}. \tag{9}$$

From the assumptions of the theorem, we know that we can express each component of $\mathcal{E}_{i,j}^{\text{old}}$ in terms of appropriate Fourier coefficients. For example, for a nodal degree of freedom, k_n , $\mathcal{E}_{i,j}^{(k_n,\text{old})} = \mathcal{E}_{i,j}^{(k_n,3)} = \alpha^{(k_n,3)} e^{i\theta \cdot \mathbf{x}_{i,j}^{(k_n)}/h}$. The values for the components of $\mathcal{E}_{i,j}^{\text{new}}$ are, of course, not known *a priori*, but we can express them in terms of generic Fourier coefficients, $\beta^{(k)} e^{i\theta \cdot \mathbf{x}_{i,j}^{(k)}/h}$ for component k . Thus, we can express the k th component of the update as

$$\mathcal{U}_{i,j}^{(k,\text{new})} - \mathcal{U}_{i,j}^{(k,\text{old})} = \mathcal{E}_{i,j}^{(k,\text{old})} - \mathcal{E}_{i,j}^{(k,\text{new})} = (\alpha^{(k,\text{old})} - \beta^{(k)}) e^{i\theta \cdot \mathbf{x}_{i,j}^{(k)}/h},$$

where $\alpha^{(k,\text{old})}$ is the appropriate Fourier coefficient for component k before the correction of $S_{i,j}$.

With this notation, we can restate the result to be proven as, simply, $\beta^{(k)} = \alpha^{(k,\text{new})}$ for all k , where, for each k , the value of $\alpha^{(k,\text{new})}$ is as given in the final equations of the statement of the theorem.

Notice, however, that solution to the update equation at any other point already processed already has this form; in particular, the solution to

$$B(\mathcal{U}_{i-1,j}^{\text{new}} - \mathcal{U}_{i-1,j}^{\text{old}}) = \mathcal{R}_{i-1,j}^{\text{old}}$$

has components

$$\mathcal{U}_{i-1,j}^{(k,\text{new})} - \mathcal{U}_{i-1,j}^{(k,\text{old})} = \mathcal{E}_{i-1,j}^{(k,\text{old})} - \mathcal{E}_{i-1,j}^{(k,\text{new})} = (\alpha^{(k,\text{old})} - \alpha^{(k,\text{new})}) e^{i\theta \cdot \mathbf{x}_{i-1,j}^{(k)} / h},$$

following the assumptions in the theorem. Thus, for each k , $\beta^{(k)} = \alpha^{(k,\text{new})}$ if and only if

$$\mathcal{U}_{i,j}^{(k,\text{new})} - \mathcal{U}_{i,j}^{(k,\text{old})} = (\alpha^{(k,\text{old})} - \alpha^{(k,\text{new})}) e^{i\theta \cdot \mathbf{x}_{i,j}^{(k)} / h} = e^{i\theta_1} (\mathcal{U}_{i-1,j}^{(k,\text{new})} - \mathcal{U}_{i-1,j}^{(k,\text{old})}).$$

It follows that $\beta^{(k)} = \alpha^{(k,\text{new})}$ for all k if and only if

$$\mathcal{R}_{i,j}^{\text{old}} = B(\mathcal{U}_{i,j}^{\text{new}} - \mathcal{U}_{i,j}^{\text{old}}) = e^{i\theta_1} B(\mathcal{U}_{i-1,j}^{\text{new}} - \mathcal{U}_{i-1,j}^{\text{old}}) = e^{i\theta_1} \mathcal{R}_{i-1,j}^{\text{old}},$$

since B is non-singular. Thus, to prove the theorem, we only need to demonstrate that $\mathcal{R}_{i,j}^{\text{old}} = e^{i\theta_1} \mathcal{R}_{i-1,j}^{\text{old}}$.

Consider, then, a single degree of freedom, k , in $S_{i,j}$. The residual, $r_{i,j}^{(k)}$, associated with k before relaxation on $S_{i,j}$ can be expressed as a function of the Fourier coefficients corresponding to the errors at degrees of freedom connected to k , since $r_{i,j}^{(k)} = (A_h \mathcal{E})_{i,j}^{(k)}$. Note, however, that for any k , for any i, j , at the step before relaxation on $S_{i,j}$, we can write $r_{i,j}^{(k)} = f^{(k)}(\{\alpha\}) e^{i\theta \cdot \mathbf{x}_{i,j}^{(k)} / h}$, where $\{\alpha\}$ denotes the set of all Fourier indices, as described in the statement of the theorem. In particular, for any k , the function $f^{(k)}(\{\alpha\})$ depends only on k and $\{\alpha\}$ and is independent of i and j , since the states of the variables in each update set, $S_{\ell,m}$, are always the same before relaxation treats $S_{\ell,m}$. This is precisely the LFA ansatz assumed in the theorem. Thus, for each k ,

$$r_{i,j}^{(k)} = f^{(k)}(\{\alpha\}) e^{i\theta \cdot \mathbf{x}_{i,j}^{(k)} / h} = e^{i\theta_1} r_{i-1,j}^{(k)},$$

or $\mathcal{R}_{i,j}^{\text{old}} = e^{i\theta_1} \mathcal{R}_{i-1,j}^{\text{old}}$, from which the theorem follows. \square

Theorem 1 states, in essence, that the set of Fourier modes of fixed frequency, θ , for each different unknown function in the PDE system, which are the eigenfunctions of any pointwise relaxation that updates all nodes in the same pattern when applied to a scalar PDE, also form invariant subspaces for any coupled relaxation (overlapping or not) that partitions the degrees of freedom into self-similar collections of degrees of freedom that are treated consistently. In other words, the error-propagation matrix for any coupled relaxation is an infinite-grid block-multilevel-Toeplitz matrix. This, in turn, means that we can attempt to analyze these techniques using classical multigrid smoothing and two-grid Fourier analysis tools to measure the effectiveness of the resulting multigrid cycles. The generalization of this result to 3D is straightforward.

Analysis of the error-propagation operators in this context was done by Sivaloganathan [20] for Vanka relaxation [16] for the standard, staggered finite-difference discretization of the Stokes Equations in two dimensions and by Molenaar for the mixed FE discretization of Poisson's equation using Raviart–Thomas elements [21]. This technique can be generalized to apply to any overlapping relaxation that satisfies conditions such as those in Theorem 1: that A_h is a block-diagonal matrix with infinite-grid multilevel Toeplitz blocks (corresponding to the discretization of a two-dimensional PDE on a regular grid with mesh size, h), that the relaxation subsets, $S_{i,j}$, are determined also by an infinite grid with mesh size h , and that the update matrix, B , is non-singular. Under these conditions, Equation (8) can be rewritten to give the transformation of the Fourier coefficients through the relaxation sweep.

Each equation in (8) can be rewritten to give an equation relating the set of updated Fourier coefficients to the Fourier coefficients before the sweep (by moving the appropriate terms from the residual to the left-hand side, and those from the update equations to the right). The resulting system of equations can be written as $L\alpha^{\text{new}} = M\alpha^{\text{old}}$, where L is a $(4k + 2(l_x + l_y) + m) \times (4k + 2(l_x + l_y) + m)$ matrix, whereas M is a $(4k + 2(l_x + l_y) + m) \times (k + l_x + l_y + m)$ matrix. Computing $L^{-1}M$ gives the error-propagation operator that maps from the error before the sweep to each of the partially updated Fourier coefficients, as well as to the fully updated coefficients, $\alpha^{(k_n, 4)}$, $\alpha^{(k_n, 2)}$, $\alpha^{(k_v, 2)}$, and $\alpha^{(k_i, 1)}$. Taking the $(k + l_x + l_y + m) \times (k + l_x + l_y + m)$ submatrix that corresponds to the rows of $L^{-1}M$ associated with the fully updated Fourier coefficients gives the error-propagation operator for relaxation as a whole. The next section gives a detailed example of this approach.

4. OVERLAPPING SCHWARZ RELAXATION FOR THE POISSON EQUATION

We explain the LFA for multiplicative smoothers in detail for Poisson's equation with a bilinear FE discretization, using an element-wise overlapping Schwarz relaxation. For this discrete operator, this (somewhat involved) smoother is not really necessary, as basic pointwise relaxation is sufficient. This smoother could, however, be useful for the Poisson operator in a discontinuous Galerkin context. For the bilinear ($Q1$) discretization, a typical equation of the linear systems is

$$\frac{9}{3}u_{i,j} - \frac{1}{3} \sum_{\alpha=-1}^1 \sum_{\beta=-1}^1 u_{i+\alpha, j+\beta} = f_{i,j}.$$

By element-wise overlapping, we mean that the relaxation traverses the grid element by element, updating the four nodes at the corners of the element at each step. Subset $S_{i,j}$ is taken to be the four nodes to the North and East of (i, j) : $S_{i,j} = \{(i, j), (i+1, j), (i, j+1), (i+1, j+1)\}$. Thus, before we relax on $S_{i,j}$, the variables that appear in the equations for $S_{i,j}$ are in the following states, gathered by the number of times they have been updated prior to considering $S_{i,j}$:

Four times: $(i-1, j-1), (i, j-1), (i+1, j-1), (i+2, j-1), (i-1, j)$

Three times: (i, j)

Twice: $(i+1, j), (i+2, j), (i-1, j+1)$

Once: $(i, j+1)$

Not updated: $(i+1, j+1), (i+2, j+1), (i-1, j+2), (i, j+2), (i+1, j+2), (i+2, j+2)$

At this stage, we introduce the Fourier expansions for each mode, in terms of the number of updates: $e_{k,l} = \alpha'''' e^{i\theta \cdot \mathbf{x}_{k,l}/h}$, $e_{k,l} = \alpha'''' e^{i\theta \cdot \mathbf{x}_{k,l}/h}$, $e_{k,l} = \alpha''' e^{i\theta \cdot \mathbf{x}_{k,l}/h}$, $e_{k,l} = \alpha'' e^{i\theta \cdot \mathbf{x}_{k,l}/h}$, $e_{k,l} = \alpha e^{i\theta \cdot \mathbf{x}_{k,l}/h}$, for four, three, two, one, and no updates, respectively.

We substitute these expansions into the residual equations associated with the four nodes before the relaxation on $S_{i,j}$, where $r_{k,l}$ and $e_{k,l}$ are the residual and error, respectively, in the approximation to $u_{k,l}$ for each node (k, l) ,

$$\begin{aligned} r_{i,j} &= \left(\frac{8}{3}\alpha'' - \frac{1}{3}(\alpha'''' e^{-i(\theta_1+\theta_2)} + \alpha'''' e^{-i\theta_2} + \alpha'''' e^{i(\theta_1-\theta_2)} + \alpha'''' e^{-i\theta_1} + \alpha'' e^{i\theta_1} \right. \\ &\quad \left. + \alpha'' e^{i(-\theta_1+\theta_2)} + \alpha' e^{i\theta_2} + \alpha e^{i(\theta_1+\theta_2)}) \right) e^{i\theta \cdot \mathbf{x}_{i,j}/h} \\ r_{i+1,j} &= \left(\frac{8}{3}\alpha'' - \frac{1}{3}(\alpha'''' e^{-i(\theta_1+\theta_2)} + \alpha'''' e^{-i\theta_2} + \alpha'''' e^{i(\theta_1-\theta_2)} + \alpha'''' e^{-i\theta_1} + \alpha'' e^{i\theta_1} \right. \\ &\quad \left. + \alpha' e^{i(-\theta_1+\theta_2)} + \alpha e^{i\theta_2} + \alpha e^{i(\theta_1+\theta_2)}) \right) e^{i\theta \cdot \mathbf{x}_{i+1,j}/h} \\ r_{i,j+1} &= \left(\frac{8}{3}\alpha' - \frac{1}{3}(\alpha'''' e^{-i(\theta_1+\theta_2)} + \alpha'''' e^{-i\theta_2} + \alpha'' e^{i(\theta_1-\theta_2)} + \alpha'' e^{-i\theta_1} + \alpha e^{i\theta_1} \right. \\ &\quad \left. + \alpha e^{i(-\theta_1+\theta_2)} + \alpha e^{i\theta_2} + \alpha e^{i(\theta_1+\theta_2)}) \right) e^{i\theta \cdot \mathbf{x}_{i,j+1}/h} \\ r_{i+1,j+1} &= \left(\frac{8}{3}\alpha - \frac{1}{3}(\alpha'''' e^{-i(\theta_1+\theta_2)} + \alpha'''' e^{-i\theta_2} + \alpha'' e^{i(\theta_1-\theta_2)} + \alpha' e^{-i\theta_1} + \alpha e^{i\theta_1} \right. \\ &\quad \left. + \alpha e^{i(-\theta_1+\theta_2)} + \alpha e^{i\theta_2} + \alpha e^{i(\theta_1+\theta_2)}) \right) e^{i\theta \cdot \mathbf{x}_{i+1,j+1}/h} \end{aligned}$$

A weighted overlapping multiplicative Schwarz relaxation sweep can be written in terms of its update equation, $\mathcal{U}_{i,j}^{\text{new}} = \mathcal{U}_{i,j}^{\text{old}} + \omega A_{i,j}^{-1} \mathcal{R}_{i,j}^{\text{old}}$, where

$$A_{i,j} = \frac{1}{3} \begin{bmatrix} 8 & -1 & -1 & -1 \\ -1 & 8 & -1 & -1 \\ -1 & -1 & 8 & -1 \\ -1 & -1 & -1 & 8 \end{bmatrix}$$

is the piece of the system matrix connecting the unknowns in $S_{i,j}$, ω is an under-relaxation parameter, and $\mathcal{U}_{i,j}^{\text{old}}$, $\mathcal{U}_{i,j}^{\text{new}}$, and $\mathcal{R}_{i,j}^{\text{old}}$ are, respectively, the approximations before and after relaxation, and the residual before relaxation at the nodes in $S_{i,j}$. Rewriting this as $(1/\omega)A_{i,j}(\mathcal{E}_{i,j}^{\text{old}} - \mathcal{E}_{i,j}^{\text{new}}) = \mathcal{R}_{i,j}^{\text{new}}$ and substituting the appropriate Fourier expansions for the errors before and after relaxation at the nodes in $S_{i,j}$ gives

$$\frac{1}{3} \begin{bmatrix} 8 & -1 & -1 & -1 \\ -1 & 8 & -1 & -1 \\ -1 & -1 & 8 & -1 \\ -1 & -1 & -1 & 8 \end{bmatrix} \begin{pmatrix} \frac{1}{\omega}(\alpha''' - \alpha''''')e^{i\theta \cdot \mathbf{x}_{i,j}/h} \\ \frac{1}{\omega}(\alpha'' - \alpha''')e^{i\theta \cdot \mathbf{x}_{i+1,j}/h} \\ \frac{1}{\omega}(\alpha' - \alpha'')e^{i\theta \cdot \mathbf{x}_{i,j+1}/h} \\ \frac{1}{\omega}(\alpha - \alpha')e^{i\theta \cdot \mathbf{x}_{i+1,j+1}/h} \end{pmatrix} = \begin{pmatrix} r_{i,j} \\ r_{i+1,j} \\ r_{i,j+1} \\ r_{i+1,j+1} \end{pmatrix}. \quad (10)$$

Now, this system of four equations can be rearranged into a system of equations directly for the four updated Fourier coefficients, α'''' , α''' , α'' , α' . This is simply accomplished by expanding each equation in terms of the Fourier expansions (using the expressions for $r_{i,j}$, $r_{i+1,j}$, $r_{i,j+1}$, $r_{i+1,j+1}$ derived above), then collecting terms that multiply each of the Fourier coefficients. The common factors of $\frac{1}{3}$ and $e^{i\theta \cdot \mathbf{x}_{k,l}/h}$ can be directly canceled to simplify the calculation. For this example, the first equation may be rewritten as

$$\begin{aligned} & \left(-\frac{8}{\omega} + e^{-i(\theta_1+\theta_2)} + e^{-i\theta_1} + e^{-i\theta_2} + e^{i(\theta_1-\theta_2)} \right) \alpha'''' + \left(\frac{8}{\omega} - 8 + \frac{1}{\omega} e^{i\theta_1} \right) \alpha''' \\ & + \left(\left(1 - \frac{1}{\omega} \right) e^{i\theta_1} + \frac{1}{\omega} e^{i\theta_2} + e^{i(-\theta_1+\theta_2)} \right) \alpha'' + \left(\left(1 - \frac{1}{\omega} \right) e^{i\theta_2} + e^{i(\theta_1+\theta_2)} \right) \alpha' \\ & = \left(\frac{1}{\omega} - 1 \right) e^{i(\theta_1+\theta_2)} \alpha, \end{aligned}$$

with similar expressions resulting from the other three equations. These equations may then be solved collectively, expressing $(\alpha'''' , \alpha''' , \alpha'' , \alpha')^T = L^{-1}M\alpha$, where L is a four-by-four matrix and M is a four-by-one matrix. The first entry in (the vector) $L^{-1}M$ is the amplification factor for the complete sweep, mapping the initial error coefficient of the Fourier mode given by θ into that after a sweep of the element-wise overlapping multiplicative Schwarz relaxation.

Based on these amplification factors, we can then perform classical MG smoothing analysis, as in [5, 6], for the overlapping smoothers. Figure 3 shows the amplification factors as a function of the Fourier frequencies, θ , for both pointwise Gauss–Seidel (left) and element-wise overlapping multiplicative Schwarz relaxation (right). Computing the smoothing factors,

$$\mu = \max_{\theta \in [-\pi/2, 3\pi/2]^2 \setminus [-\pi/2, \pi/2]^2} \mu(\theta),$$

where $\mu(\theta)$ is the amplification factor for relaxation for a given Fourier mode, θ ; for these two approaches, we see that, for pointwise Gauss–Seidel, $\mu = 0.43$, whereas for the overlapping relaxation, $\mu = 0.24$, or that one sweep of the overlapping relaxation reduces high-frequency errors

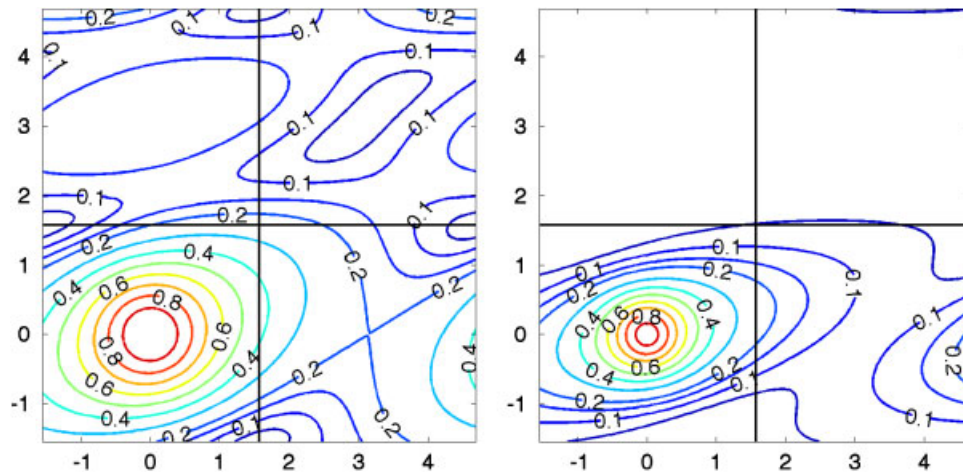


Figure 3. Amplification factors for pointwise Gauss–Seidel relaxation (at left) and element-wise overlapping multiplicative Schwarz (at right) for a Q_1 discretization of the Poisson equation on a mesh with $h = \frac{1}{128}$, as a function of the Fourier mode, θ .

Table I. Average convergence factor over 50 iterations for multigrid cycles based on pointwise and overlapping relaxation schemes.

Grid	Pointwise Gauss–Seidel		Overlapping Schwarz	
	$V(1, 1)$	$W(1, 1)$	$V(1, 1)$	$W(1, 1)$
128×128	0.101	0.068	0.032	0.022
256×256	0.103	0.068	0.034	0.023
512×512	0.103	0.068	0.034	0.023

about the same amount as 1.7 sweeps of pointwise relaxation. (Note that the factor $\mu=0.43$ for the pointwise Gauss–Seidel iteration applied to the FE discretization considered here is different from the often-cited factor of 0.5 for the corresponding finite-difference discretization.)

Furthermore, we can combine this smoothing analysis with the well-known LFA two-grid analysis for scalar PDEs [6, 27] of the coarse-grid correction for this system, using bilinear interpolation and full-weighting restriction, coupled with a Galerkin coarse-grid operator. (Details of two-grid analysis for the case of discretized systems of PDEs will be discussed shortly in Section 5.) The largest-magnitude eigenvalue predicted by the two-grid LFA for pointwise relaxation is 0.073, whereas it is 0.024 for the overlapping Schwarz relaxation in a (1,1)-cycle. One cycle of multigrid with the overlapping relaxation brings about the same total reduction in error as 1.4 cycles using pointwise relaxation. Thus, the overlapping relaxation yields a better solver, but the extra cost of the overlapping relaxation likely does not pay off (unless it can be implemented very efficiently). As a comparison, we consider the true performance of multigrid $V(1, 1)$ and $W(1, 1)$ cycles using both pointwise Gauss–Seidel and element-wise overlapping multiplicative Schwarz smoothers, shown in Table I. Here, we see that the two-grid LFA accurately predicts the W -cycle multigrid convergence rates for both smoothers, but is a noticeable underestimate for the V -cycle convergence rates. This is typical of LFA, because the two-grid analysis is based on exact solution of the first coarse-grid problem; a multigrid W -cycle, where this level is visited twice per iteration, is a much better approximation of this than a V -cycle, which uses much less relaxation.

5. TWO-GRID LFA

We discuss here the basics of two-grid LFA in order to deal with systems of PDEs on staggered FE grids. We focus on two-grid analysis, but multilevel analysis is also possible using inductive arguments.

In general, two-grid methods can be represented by error-propagation operators with form

$$E_h^{(\text{TG})} = (I - M_h^{-1} A_h)^{v_2} (I - P_h^H B_H^{-1} R_H^h A_h) (I - M_h^{-1} A_h)^{v_1}, \quad (11)$$

where H denotes the mesh size of the coarse scale, R_H^h is the restriction operator from grid h to grid H , P_h^H is the interpolation operator from grid H to grid h , and B_H represents some discretization on the coarse scale.

Writing the eigenvector matrix for the error-propagation operator associated with relaxation as $\Phi_h = [\phi^{(1)}, \phi^{(2)}, \dots, \phi^{(N)}]$, we know that $I - M_h^{-1} A_h$ is diagonalized by a similarity transformation with Φ_h ,

$$(\Phi_h)^{-1} (I - M_h^{-1} A_h) \Phi_h = \Lambda,$$

where Λ is the diagonal matrix of eigenvalues of relaxation, $\Lambda_{ii} = \lambda_i$. We can block-diagonalize $E_h^{(\text{TG})}$, also using Φ_h in a similarity transformation. Taking Φ_H to be the matrix of eigenvectors of B_H , we can then write

$$\Phi_h^{-1} E_h^{(\text{TG})} \Phi_h = \Lambda^{v_2} (I - (\Phi_h^{-1} P_h^H \Phi_H) \Gamma^{-1} (\Phi_H^{-1} R_H^h \Phi_h) (\Phi_h^{-1} A_h \Phi_h)) \Lambda^{v_1}, \quad (12)$$

where $\Gamma = \Phi_H^{-1} B_H \Phi_H$ is also a diagonal operator.

Under the LFA assumptions for smoothing that A_h and M_h are infinite-grid (multilevel) Toeplitz matrices, $\Phi_h^{-1} A_h \Phi_h$ is also a diagonal matrix. If we additionally take B_H to correspond to the discretization of a PDE on an infinite grid with fixed mesh size, H , with a stencil that does not vary with position, then the difficulty in analyzing (12) comes from the intergrid-transfer operators, $\Phi_H^{-1} R_H^h \Phi_h$ and $\Phi_h^{-1} P_h^H \Phi_H$. The transformation of R_H^h and P_h^H in terms of the coarse-grid and fine-grid Fourier matrices, Φ_H and Φ_h , depends on the relationship between the two mesh sizes, H and h . Taking $H = 2h$, as is commonly the case in geometric multigrid, then a constant-stencil restriction operator, R_H^h , for a two-dimensional mesh maps four fine-grid frequencies into one coarse-grid function. These four functions, known as the Fourier harmonics, are associated with some base index, $\theta_{0,0} \in (-\frac{\pi}{2}, \frac{\pi}{2}]^2$, and three more-oscillatory modes, associated with frequencies

$$\theta_{1,0} = \theta_{0,0} + \begin{pmatrix} \pi \\ 0 \end{pmatrix}, \quad \theta_{0,1} = \theta_{0,0} + \begin{pmatrix} 0 \\ \pi \end{pmatrix} \quad \text{and} \quad \theta_{1,1} = \theta_{0,0} + \begin{pmatrix} \pi \\ \pi \end{pmatrix}.$$

The action of a constant-coefficient restriction operator, R_{2h}^h , on a fine-grid residual, \mathbf{r}_h , can be written in terms of a set of restriction weights, $\{w_{k,l}\}$, as

$$(R_{2h}^h \mathbf{r}_h)_{I,J} = \sum_{k,l} w_{k,l} r_{i+k, j+l}, \quad (13)$$

where (i, j) is the fine-grid index corresponding to coarse-grid index (I, J) , hence $(i, j) = (2I, 2J)$. Note that the weights, $w_{k,l}$ for restriction are independent of the coarse-grid degree of freedom, (I, J) , being considered. This means that the rows of R_{2h}^h all take the same form, with only a simple shift in location of the non-zero entries, depending on (I, J) .

Similarly, we can write the action of a constant-coefficient interpolation operator, P_h^{2h} , on a coarse-grid correction, \mathbf{e}_{2h} , in terms of several sets of interpolation weights, $\{w_{K,L}^{(m)}\}$,

$$(P_h^{2h} \mathbf{e}_{2h})_{i,j} = \sum_{K,L} w_{K,L}^{(m)} e_{I+K, J+L}, \quad (14)$$

where the additional superscript, m , is used to denote the position of node (i, j) relative to the coarse node, (I, J) , see Figure 4. Notice that these weights are independent of the absolute location of the underlying nodes, but do depend on the staggering of the nodes relative to the coarse grid, $(i - 2I, j - 2J)$. In other words, while the columns of P_h^{2h} take the same form, up to shifts, there

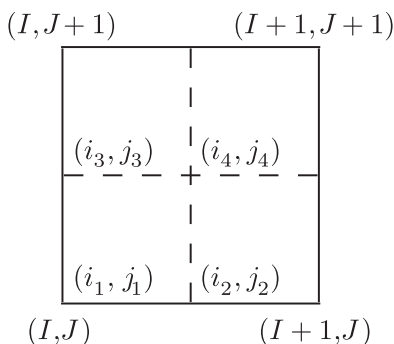


Figure 4. Nesting of fine-grid nodes relative to the coarse grid for nested (collocated) grids.

are m distinct forms for the rows of P_h^{2h} . In Figure 4, there are four distinct cases: a fine-grid node (i_1, j_1) that is co-located with the coarse-grid nodes, a fine-grid node (i_2, j_2) that lies on an x -axis-oriented line between two coarse-grid nodes, a fine-grid node (i_3, j_3) that lies on a y -axis-oriented line between two coarse-grid nodes, and a fine-grid node (i_4, j_4) that lies at the center of a coarse-grid cell. The weights for interpolation to each of these types of points will be different, hence the notation $w_{K,L}^{(m)}$.

The set of modes, $\phi_{2h}(\mathbf{x}, 2\theta_{0,0})$ for $\theta_{0,0} \in (-\pi/2, \pi/2]^2$, is a complete set of Fourier modes on the coarse grid and thus, by assumption, diagonalize the multilevel Toeplitz operator, B_{2h} . As such, the spaces of harmonic frequencies become invariant subspaces for the coarse-grid correction process and for the two-grid cycle as a whole. From the similarity transformation representation given in Equation (12), we can then compute the eigenvalues of the two-grid error-propagation operator by computing the eigenvalues of $\Phi_h^{-1} E_h^{(\text{TG})} \Phi_h$. This matrix is easily permuted into block-diagonal form with, at most, 4×4 blocks corresponding to the spaces of harmonic modes. Therefore, to compute the LFA eigenvalues, we only need to compute the spectra of 4×4 matrices.

5.1. LFA for systems

For systems of PDEs, the LFA analysis does not diagonalize A_h through the Fourier-mode similarity transformation but, rather, transforms the $N \times N$ block-matrix, A_h , corresponding to the discretization of a system of N PDEs with N unknown functions, into a matrix that can be permuted into a block-diagonal form with dense $N \times N$ blocks, by diagonalizing each block within A_h . The block coupling of the operator resulting from the full similarity transformation will have larger dense blocks than 4×4 , as we must also account for the $N \times N$ coupling within A_h (and relaxation) that arises because of the systems form of A_h . Thus, LFA for systems in 2D results in coupled $4N \times 4N$ blocks of harmonics (with four harmonics for each of the N scalar unknowns of the PDE) for each base frequency, $\theta_{0,0} \in (-\pi/2, \pi/2]^2$. Aside from this difference, the analysis below proceeds in the same manner as in the scalar case.

5.2. Grid transfers for staggered systems

An added challenge in the multigrid treatment of a system of PDEs is that each different variable type may be staggered in its own way.

If the different variable types do not interact in interpolation and restriction (so that each variable type only restricts to and interpolates from a coarse-grid variable with the same staggering pattern), then the LFA for interpolation for the whole system can be done treating each variable type, in turn, as a scalar problem. If, on the other hand, there is a need for inter-variable interpolation or restriction in the treatment of the system, we must modify the method for the scalar case to account for the different staggering on the two grid levels. As in Figure 5, we now take (δ_1, δ_2) to

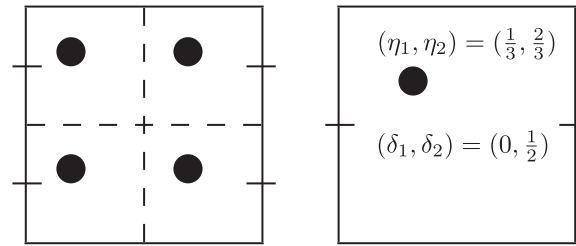


Figure 5. Nesting of fine-grid nodes relative to the coarse grid for non-nested grids, with $(\delta_1, \delta_2) = (0, \frac{1}{2})$ and $(\eta_1, \eta_2) = (\frac{1}{3}, \frac{2}{3})$.

describe the staggering of a fine-level variable, \mathbf{r}_h (either to be restricted from or interpolated to), and (η_1, η_2) to describe the staggering of a coarse-level variable, \mathbf{e}_{2h} .

Lemma 1

Let $\theta_{0,0} \in (-\pi/2, \pi/2]^2$, and let (I, J) be a coarse-grid node index, identified with fine-grid node index $(i, j) = (2I, 2J)$. Then, any constant-coefficient restriction operator, as defined by Equation (13), maps the four Fourier harmonic modes, $\phi_h(\mathbf{x}, \theta_{0,0})$, $\phi_h(\mathbf{x}, \theta_{1,0})$, $\phi_h(\mathbf{x}, \theta_{0,1})$, and $\phi_h(\mathbf{x}, \theta_{1,1})$, on the grid with shift (δ_1, δ_2) into the single coarse-grid mode, $\phi_{2h}(\mathbf{x}, 2\theta_{0,0})$, on the grid with shift (η_1, η_2) .

Proof

Considering a fine-grid residual, \mathbf{r}_h , that is a linear combination of the four harmonic frequencies,

$$\begin{aligned} r_{i,j} &= c_{0,0}\phi_h(((i+\delta_1)h, (j+\delta_2)h), \theta_{0,0}) + c_{1,0}\phi_h(((i+\delta_1)h, (j+\delta_2)h), \theta_{1,0}) \\ &\quad + c_{0,1}\phi_h(((i+\delta_1)h, (j+\delta_2)h), \theta_{0,1}) + c_{1,1}\phi_h(((i+\delta_1)h, (j+\delta_2)h), \theta_{1,1}) \\ &= c_{0,0}e^{i\theta_{0,0}\cdot(i+\delta_1, j+\delta_2)} + c_{1,0}e^{i\theta_{1,0}\cdot(i+\delta_1, j+\delta_2)} + c_{0,1}e^{i\theta_{0,1}\cdot(i+\delta_1, j+\delta_2)} + c_{1,1}e^{i\theta_{1,1}\cdot(i+\delta_1, j+\delta_2)} \\ &= (c_{0,0} + c_{1,0}e^{i\pi i} e^{i\pi\delta_1} + c_{0,1}e^{i\pi j} e^{i\pi\delta_2} + c_{1,1}e^{i\pi(i+j)} e^{i\pi(\delta_1+\delta_2)})e^{i\theta_{0,0}\cdot(\delta_1, \delta_2)} e^{i\theta_{0,0}\cdot(i, j)}. \end{aligned}$$

As the nodes are numbered based on their cell, node (I, J) on grid $2h$ can be identified with node (i, j) on grid h for $i = 2I$ and $j = 2J$. We seek to represent the restricted residual on the coarse grid, $R_{2h}^h \mathbf{r}_h$, as a multiple of the coarse-grid harmonic function, $\phi_{2h}((I + \eta_1)(2h), (J + \eta_2)(2h), 2\theta_{0,0}) = e^{i2\theta_{0,0}\cdot(I+\eta_1, J+\eta_2)}$. Thus, we can write the restriction of \mathbf{r}_h to a differently staggered coarse-grid variable in cell (I, J) , where $(i, j) = (2I, 2J)$, as

$$\begin{aligned} (R_{2h}^h \mathbf{r}_h)_{I,J} &= \sum_{k,l} w_{k,l} r_{i+k, j+l} \\ &= e^{i2\theta_{0,0}\cdot(I+\eta_1, J+\eta_2)} \left[\sum_{k,l} w_{k,l} e^{i\theta_{0,0}\cdot(k,l)} e^{i\theta_{0,0}\cdot(\delta_1-2\eta_1, \delta_2-2\eta_2)} (c_{0,0} + c_{1,0}e^{i\pi k} e^{i\pi\delta_1} \right. \\ &\quad \left. + c_{0,1}e^{i\pi l} e^{i\pi\delta_2} + c_{1,1}e^{i\pi(k+l)} e^{i\pi(\delta_1+\delta_2)}) \right] \end{aligned}$$

□

Lemma 2

Let $\theta_{0,0} \in (-\pi/2, \pi/2]^2$ and (I, J) be a coarse-grid node index. Let (i_1, j_1) , (i_2, j_2) , (i_3, j_3) , and (i_4, j_4) be four fine-grid node indices, as identified in Figure 5. Then, any constant-coefficient interpolation operator on the shifted grid maps coarse-grid mode $\phi_{2h}(\mathbf{x}, 2\theta_{0,0})$ into the four fine-grid harmonics, $\phi_h(\mathbf{x}, \theta_{0,0})$, $\phi_h(\mathbf{x}, \theta_{1,0})$, $\phi_h(\mathbf{x}, \theta_{0,1})$, and $\phi_h(\mathbf{x}, \theta_{1,1})$.

Proof

The Fourier analysis for interpolation can be derived by similarly accounting for the different staggering of a coarse-grid variable, \mathbf{e}_{2h} , staggered on grid $2h$ with (η_1, η_2) , and its fine-grid interpolant, staggered on grid h with (δ_1, δ_2) . Choosing $e_{I,J} = \phi_{2h}((I + \eta_1)(2h), (J + \eta_2)(2h), 2\theta_{0,0})$, we get (taking $(i_1, j_1) = (2I, 2J)$)

$$(P_h^{2h} \mathbf{e}_{2h})_{i_1, j_1} = \left(\sum_{K,L} w_{K,L}^{(1)} e^{i2\theta_{0,0} \cdot (K,L)} e^{i\theta_{0,0} \cdot (2\eta_1 - \delta_1, 2\eta_2 - \delta_2)} \right) e^{i\theta_{0,0} \cdot (i_1 + \delta_1, j_1 + \delta_2)}. \tag{15}$$

Similarly, we can derive the staggered interpolation relations for the other three node points in coarse-grid cell (I, J) as

$$\begin{aligned} (P_h^{2h} \mathbf{e}_{2h})_{i_2, j_2} &= \left(\sum_{K,L} w_{K,L}^{(2)} e^{i\theta_{0,0} \cdot (2K-1, 2L)} e^{i\theta_{0,0} \cdot (2\eta_1 - \delta_1, 2\eta_2 - \delta_2)} \right) e^{i\theta_{0,0} \cdot (i_2 + \delta_1, j_2 + \delta_2)}, \\ (P_h^{2h} \mathbf{e}_{2h})_{i_3, j_3} &= \left(\sum_{K,L} w_{K,L}^{(3)} e^{i\theta_{0,0} \cdot (2K, 2L-1)} e^{i\theta_{0,0} \cdot (2\eta_1 - \delta_1, 2\eta_2 - \delta_2)} \right) e^{i\theta_{0,0} \cdot (i_3 + \delta_1, j_3 + \delta_2)}, \\ (P_h^{2h} \mathbf{e}_{2h})_{i_4, j_4} &= \left(\sum_{K,L} w_{K,L}^{(4)} e^{i\theta_{0,0} \cdot (2K-1, 2L-1)} e^{i\theta_{0,0} \cdot (2\eta_1 - \delta_1, 2\eta_2 - \delta_2)} \right) e^{i\theta_{0,0} \cdot (i_4 + \delta_1, j_4 + \delta_2)}. \end{aligned}$$

Making the ansatz that $P_h^{2h} \mathbf{e}_{2h}$ can be written as a linear combination of the four Fourier harmonics, now on the shifted grid, we have

$$\begin{aligned} (P_h^{2h} \mathbf{e}_{2h})_{i,j} &= c_{0,0} e^{i\theta_{0,0} \cdot (i + \delta_1, j + \delta_2)} + c_{1,0} e^{i\theta_{1,0} \cdot (i + \delta_1, j + \delta_2)} \\ &\quad + c_{0,1} e^{i\theta_{0,1} \cdot (i + \delta_1, j + \delta_2)} + c_{1,1} e^{i\theta_{1,1} \cdot (i + \delta_1, j + \delta_2)} \\ &= (c_{0,0} + c_{1,0} e^{i\pi(i + \delta_1)} + c_{0,1} e^{i\pi(j + \delta_2)} + c_{1,1} e^{i\pi(i + j + \delta_1 + \delta_2)}) e^{i\theta_{0,0} \cdot (i + \delta_1, j + \delta_2)}. \end{aligned}$$

Equating terms for $(i_1, j_1) = (2I, 2J)$ gives

$$c_{0,0} + c_{1,0} e^{i\pi\delta_1} + c_{0,1} e^{i\pi\delta_2} + c_{1,1} e^{i\pi(\delta_1 + \delta_2)} = c_1(\theta_{0,0}),$$

for $c_1(\theta_{0,0})$ given by the expression in Equation (15). With similar equations for the other interpolation nodes, we have

$$\begin{bmatrix} c_{0,0} \\ c_{1,0} e^{i\pi\delta_1} \\ c_{0,1} e^{i\pi\delta_2} \\ c_{1,1} e^{i\pi(\delta_1 + \delta_2)} \end{bmatrix} = \hat{\varepsilon} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} \sum_{K,L} w_{K,L}^{(1)} e^{i2\theta_{0,0} \cdot (K,L)} \\ \sum_{K,L} w_{K,L}^{(2)} e^{i\theta_{0,0} \cdot (2K-1, 2L)} \\ \sum_{K,L} w_{K,L}^{(3)} e^{i\theta_{0,0} \cdot (2K, 2L-1)} \\ \sum_{K,L} w_{K,L}^{(4)} e^{i\theta_{0,0} \cdot (2K-1, 2L-1)} \end{bmatrix},$$

with $\hat{\varepsilon} = e^{i\theta_{0,0} \cdot (2\eta_1 - \delta_1, 2\eta_2 - \delta_2)} / 4$. □

6. NUMERICAL EXAMPLES

In this section, we give several smoothing and two-grid LFA estimates for systems of PDEs within the framework of multiplicative collective smoothers of Vanka-type. These smoothers, which

explicitly deal with the large nullspaces that appear, are the only collective smoothers that give satisfactory performance for the operators of interest. We do not investigate the impact of the ordering of the cells in the present paper and stay with a lexicographical ordering of the relaxation subsets.

To take LFA from its infinite-grid setting and get a predictive analysis tool, we need to introduce a second discretization into the analysis, going from a continuous parameter, $\theta_{0,0}$, to a discrete mesh in $\theta_{0,0}$, upon which a convergence prediction can be made. Thus, the results presented here have two step-size parameters: h , the spatial grid size, which is directly reflected in the coefficients of the discrete operators to which we apply LFA, and h_θ , the mesh size for the discrete lattice of θ used to make a quantitative prediction based on LFA.

A discrete set of the 4×4 Fourier blocks is then analyzed, corresponding to a discrete choice of frequencies, $\theta_{0,0} \in (-\pi/2, \pi/2)^2$, given by a tensor product of an equally spaced mesh over the interval of length π with itself, with mesh-spacing h_θ . Because the infinite-grid fine-grid and coarse-grid operators, A_h and B_{2h} , are often singular, with the constant functions, $\phi_h(\mathbf{x}, (0, 0))$ and $\phi_{2h}(\mathbf{x}, (0, 0))$, in their nullspaces, we choose the mesh in $\theta_{0,0}$ so that $\theta_{0,0} = (0, 0)$ does not appear. A prediction of the performance of the multigrid algorithm is made by measuring the largest eigenvalue of the transformed operators over this discrete space. All of the numbers quoted here result from this process, for $h = \frac{1}{64}$ and $h_\theta = \pi/32$. The impact of finer meshes in either space or Fourier frequency was negligible in the examples considered here.

6.1. The grad-div and the curl-curl operators

We first consider the discretization of the grad-div and the curl-curl equations, (1) and (2), respectively. As the stencils, multigrid methods, and smoothers, as well as their LFA performance estimates are very similar for the two equations, we discuss them in one section.

For the grad-div equation (1), we use first-order Raviart-Thomas (face) elements for the vector field $\mathcal{U} = (u, v)^T$, as already discussed in Section 2.2. The discrete degrees of freedom for this discretization are the values of u at the midpoints of mesh edges that are parallel to the y -axis, and the values of v at the midpoint of mesh edges that are parallel to the x -axis. The resulting stencil, for u , reads

$$\begin{bmatrix} & 1 & & -1 \\ -1 & & 2 & & -1 \\ & -1 & & 1 & \end{bmatrix}_h,$$

where the one-dimensional part of the stencil, $[-1 \ 2 \ -1]$, gives the u - u connection, and the remaining terms give the u - v connection. The contribution of the mass matrix is then added to this stencil, taking the form of a one-dimensional stencil, $[h^2/6 \ 2h^2/3 \ h^2/6]$, connecting u degrees of freedom to u degrees of freedom; thus, the central stencil element for the discretization of $-\nabla(a \nabla \cdot \mathcal{U}) + \mathcal{U}$ (with constant-coefficient a) is $2a + 2h^2/3$. The resulting, rotated, stencil for the v components is similar.

As mentioned earlier, we choose the Nédélec edge elements [35] to discretize the weak curl-curl operator in (2), for unknown $\mathcal{U} = (u, v)^T$. The resulting stencil for the u -component is

$$\begin{bmatrix} & & -1 \\ -1 & & 1 \\ & 2 & \\ 1 & & -1 \\ & -1 & \end{bmatrix}_h,$$

while, for v , we find an identical, but rotated, stencil. It is clear that the resulting stencils for the grad-div and curl-curl operators are identical, but rotated. Hence, we focus on the discussion of the smoother for grad-div, as the one for curl-curl is similar and produces identical LFA results.

The overlapping smoother, in particular for the grad-div operator, was proposed in [15, 37], where the degrees of freedom along the faces (edges) adjacent to a node were chosen to be relaxed simultaneously, see Figure 1. We refer to this as a *node-wise* smoothing procedure.

Thus, we take the collection, $S_{i,j}$, of degrees of freedom to be relaxed simultaneously to be $S_{i,j} = \{u_{i,j-1/2}, u_{i,j+1/2}, v_{i-1/2,j}, v_{i+1/2,j}\}$, and introduce intermediate variables for the Fourier expansions of the errors in u and v at each stage in the smoothing process, as in Section 3: before relaxation, $\alpha_u e^{i\theta \cdot \mathbf{x}/h}$, $\alpha_v e^{i\theta \cdot \mathbf{x}/h}$; after the first correction, $\alpha'_u e^{i\theta \cdot \mathbf{x}/h}$, $\alpha'_v e^{i\theta \cdot \mathbf{x}/h}$; and after the second correction, $\alpha''_u e^{i\theta \cdot \mathbf{x}/h}$, $\alpha''_v e^{i\theta \cdot \mathbf{x}/h}$. We can then write the update equations in terms of the Fourier coefficients:

$$\begin{bmatrix} \frac{2h^2}{3} + 2a & 0 & a & -a \\ 0 & \frac{2h^2}{3} + 2a & -a & a \\ a & -a & \frac{2h^2}{3} + 2a & 0 \\ -a & a & 0 & \frac{2h^2}{3} + 2a \end{bmatrix} \begin{bmatrix} \delta u_{i,j+\frac{1}{2}} \\ \delta u_{i,j-\frac{1}{2}} \\ \delta v_{i+\frac{1}{2},j} \\ \delta v_{i-\frac{1}{2},j} \end{bmatrix} = \begin{bmatrix} r_{i,j+\frac{1}{2}}^u \\ r_{i,j-\frac{1}{2}}^u \\ r_{i+\frac{1}{2},j}^v \\ r_{i-\frac{1}{2},j}^v \end{bmatrix},$$

and convert this system into a system for α''_u , α''_v , α'_u , α'_v in terms of α_u and α_v , as explained for Equation (10).

In the numerical LFA smoothing and two-grid experiments here, we vary the transfer operators in the algorithms. We consider the usual six-point restriction operator, based on the fine-grid residuals at the six nearest fine-grid locations of the corresponding unknown. This operator is dictated by the staggered arrangement of the unknowns. For the prolongation operators, we compare the transpose of the six-point restriction, i.e. the six-point interpolation, with a twelve-point interpolation and a two-point interpolation. We denote these transfer operators by $6r$, $6p$, $12p$ and $2p$, respectively. See [52] or [27, Section 8.7.1] for details on these transfer operators. In the experiments, we fix the smoother to be the lexicographical AFW smoother; the corresponding smoothing factor, based on one smoothing iteration, is $\mu=0.44$. Further, the Galerkin coarse-grid operator is chosen. Table II presents the corresponding two-grid LFA results. Results here are shown for the grad-div (and curl-curl) parameter $a=10^6$. However, for values of a ranging from 10^6 to $O(1)$, we find identical LFA results; that is, no sensitivity to the value of the parameter a is observed. The results in Table II do demonstrate a sensitivity to the choice of transfer operators.

The algorithm with the commonly chosen six-point transfer operators, however, shows an excellent two-grid factor for these problems. Very similar results are obtained with finite-difference multigrid experiments for the grad-div operator in [29]. These results can be labeled as *text-book multigrid efficiency*, and an increasing number of smoothing steps further decreases the two-grid factors. In contrast, LFA smoothing analysis, as well as actual numerical multigrid experiments, indicate that an element-wise multiplicative smoothing method, updating unknowns

Table II. Two-grid LFA factors for different sets of transfer operators. Grad-div and curl-curl results with a multiplicative AFW smoother.

Restriction	Prolongation	$\rho_{2g}, (1, 1)\text{-cycle}$
$6r$	$6p$	0.134
$6r$	$12p$	0.134
$6r$	$2p$	0.53

Table III. Two-grid LFA convergence factors, ρ_{2g} , and smoothing factors, μ , for the full Vanka smoother for the $Q2-Q1$ discretization of Stokes with pressure-wise relaxation groups.

ω	Velocity under-relaxation		Full under-relaxation	
	ρ_{2g}	μ	ρ_{2g}	μ
1.0	0.77	0.76	0.77	0.76
0.9	0.67	0.74	0.58	0.67
0.8	0.58	0.71	0.43	0.61
0.7	0.48	0.68	0.37	0.60
0.6	0.38	0.65	0.45	0.67
0.5	0.36	0.60	0.54	0.74

In the left two columns, under-relaxation is used only on the velocity variables, whereas it is used on all variables (with the same under-relaxation parameter) in the right two columns.

around the element center simultaneously, does *not* provide any smoothing for this type of problem.

6.2. Stokes equations

For the Taylor–Hood ($Q2-Q1$) elements on quadrilateral grids, there are many possible collections of unknowns that can be used to define the relaxation subsets, $S_{i,j}$. Vanka’s original choice for the staggered finite-difference discretization of the Stokes equations [16] can be viewed both as being an element-wise choice, as each relaxation subset consists of the four velocity unknowns and single pressure unknown on each grid cell, and also as being a pressure-wise choice, as these relaxation subsets also correspond to the complete set of velocities that appear in the divergence equation at a given pressure node plus that pressure node itself. For this FE discretization, however, the element-wise and pressure-wise relaxation subsets are distinct. Further choices are also possible, such as the larger collections analyzed in [22]. As first noted (without explanation) in [17], difficulties arise in using element-wise smoothing for the Taylor–Hood discretization. LFA confirms this, with substantial under-relaxation necessary to achieve a smoothing factor that is less than one. For this reason, we focus here on the pressure-wise smoothing algorithm.

Table III displays both two-grid LFA convergence factors, ρ_{2g} , and LFA smoothing factors, μ , for the full Vanka smoother using pressure-wise relaxation. Two types of under-relaxation are considered: under-relaxation on only the velocities (as first suggested by Vanka [16]) and full under-relaxation, where the corrections to velocities and pressure are weighted equally. Both types of under-relaxation are shown to yield significant improvement in both the smoothing and two-grid convergence factors. Figures 6 and 7 depict the amplification factors for relaxation as a function of the Fourier mode and the spectrum of the two-grid operator (sampled on a 64×64 mesh in the Fourier domain $[-\pi/2, 3\pi/2]^2$) for no under-relaxation (in Figure 6), and under-relaxation in velocity only with a factor of $\omega=0.5$ (in Figure 7). Note that in the under-relaxed case, the amplification factor near (π, π) is greater than 0.5; however, in this case, the reduction of error in the Fourier modes with frequencies near (π, π) by two steps of relaxation is in balance with the reduction in the smooth modes by the coarse-grid correction process, leading to a much more efficient cycle than is achieved with no weighting of the relaxation process.

Because of the expense of inverting the full velocity submatrix associated with the relaxation subsets, a cheaper alternative to the full Vanka smoother using only the diagonal of this matrix is often considered [17, 23]. Table IV gives two-grid LFA convergence factors, ρ_{2g} , and smoothing factors, μ , for this variant, known as the diagonal Vanka smoother. Here, we see that without under-relaxation, the smoother is mildly divergent; however, under-relaxation quickly resolves this and leads to a smoother that is not only much less expensive than, but also less effective

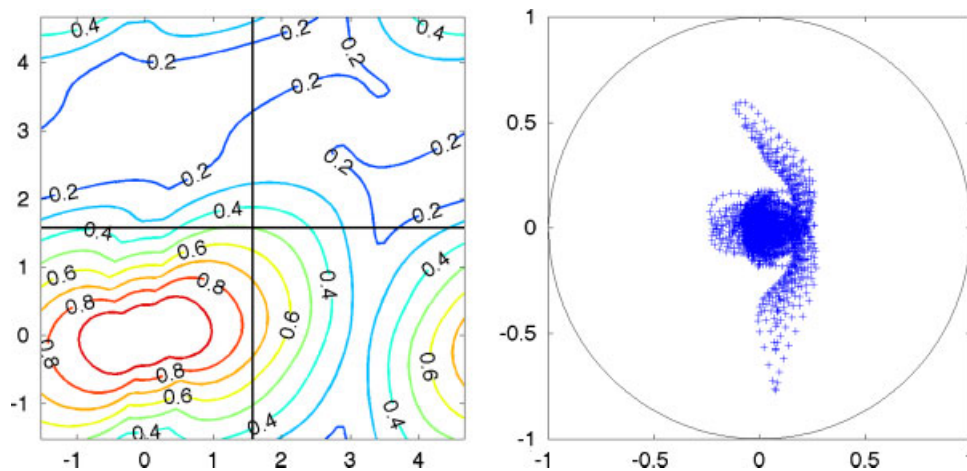


Figure 6. At left, the amplification factor for the full pressure-wise Vanka smoother with no under-relaxation for the $Q2-Q1$ discretization of the Stokes equations on a mesh with $h = \frac{1}{64}$, as a function of the Fourier mode, θ . At right, the spectrum of the two-grid error-propagation operator using this smoother with biquadratic interpolation for the velocities and bilinear interpolation for the pressure.

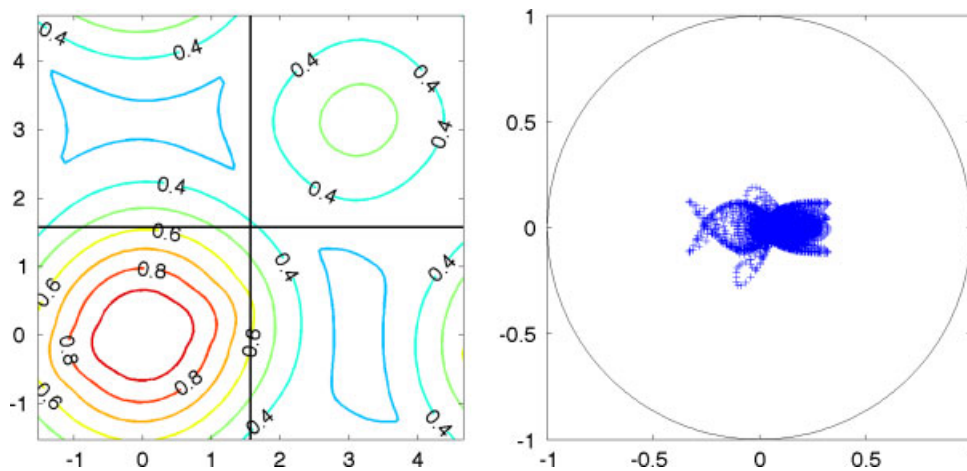


Figure 7. At left, the amplification factor for the full pressure-wise Vanka smoother with under-relaxation with parameter $\omega = 0.5$ used for the velocities only with the $Q2-Q1$ discretization of the Stokes equations on a mesh with $h = \frac{1}{64}$, as a function of the Fourier mode, θ . At right, the spectrum of the two-grid error-propagation operator using this smoother with biquadratic interpolation for the velocities and bilinear interpolation for pressure.

than the full Vanka smoother. The amplification factor for this smoother and the spectrum of the two-grid iteration matrix using under-relaxation on all variables with parameter $\omega = 0.8$ are shown in Figure 8.

If we ignore the inf-sup condition (7) when discretizing Equations (5) and (6), one appealing discretization would be to represent the velocities by bilinear basis functions and pressure as a piece-wise constant function on each element. While it is well known that such an approach (without the addition of a stabilization term) sacrifices significant accuracy in the resulting solution [40, 53], it is equally important to note the difficulty in designing an efficient solver for this discretization. Figure 9 shows the amplification factor for element-wise Vanka smoothing for this discretization (which is, in this case, the same as the pressure-wise smoother) at left, and the spectrum of the two-grid error-propagation operator for a (1,1)-cycle at right. Adding under-relaxation to the velocities

Table IV. Two-grid LFA convergence factors, ρ_{2g} , and smoothing factors, μ , for the diagonal Vanka smoother for the $Q2-Q1$ discretization of Stokes with pressure-wise relaxation groups.

ω	Velocity under-relaxation		Full under-relaxation	
	ρ_{2g}	μ	ρ_{2g}	μ
1.0	1.41	1.22	1.41	1.22
0.9	1.10	1.08	0.71	0.89
0.8	0.83	0.95	0.52	0.75
0.7	0.59	0.84	0.56	0.70
0.6	0.56	0.77	0.61	0.74
0.5	0.61	0.78	0.64	0.78

In the left two columns, under-relaxation is used only on the velocity variables, whereas it is used on all variables (with the same under-relaxation parameter) in the right two columns.

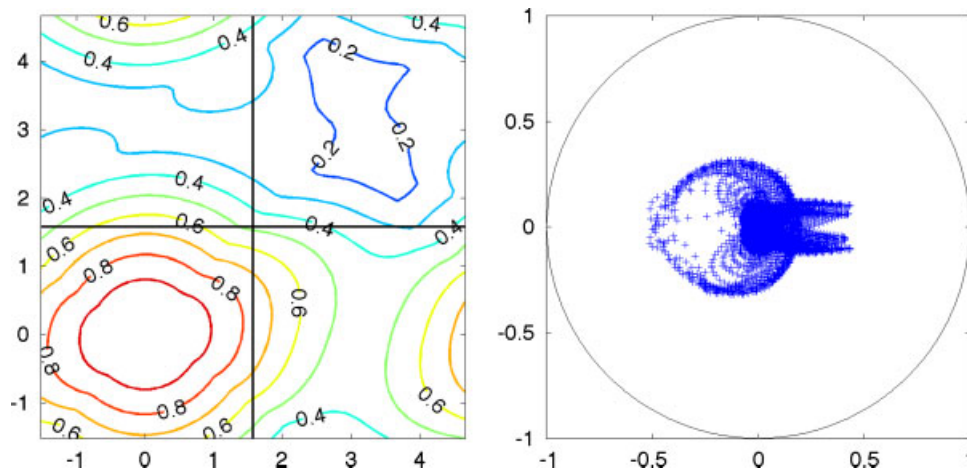


Figure 8. At left, the amplification factor for the diagonal pressure-wise Vanka smoother with under-relaxation on all variables with parameter $\omega=0.8$ for the $Q2-Q1$ discretization of the Stokes equations on a mesh with $h=\frac{1}{64}$, as a function of the Fourier mode, θ . At right, the spectrum of the two-grid error-propagation operator using this smoother with biquadratic interpolation for the velocities and bilinear interpolation for pressure.

has little effect on these results, with an LFA smoothing factor of 0.995 for $\omega=0.6, 0.7, 0.8, 0.9$, and 1.0. Of particular interest is the large amplification factor associated with the Fourier mode at (π, π) , which is also the same mode that leads to the instability in the discretization.

7. CONCLUSION

In this paper, we present a theoretical analysis on the validity of LFA for multigrid methods in which staggered grid transfers and multiplicative overlapping smoothers are included. The LFA for these smoothers had been performed before in two specific instances, but fundamental insight into the validity of the exponential Fourier functions as eigenfunctions of these smoothers has not been found in the literature. We focus on systems of PDEs discretized by state-of-the-art FEs: the grad-div equation with Raviart-Thomas elements, the curl-curl equation with Nédélec edge elements, and the Stokes equation with Taylor-Hood elements. We show that the analysis for overlapping smoothers, as developed for fluid mechanics problems, can easily be applied to the model equations in electromagnetics. The corresponding multiplicative smoothers give rise to

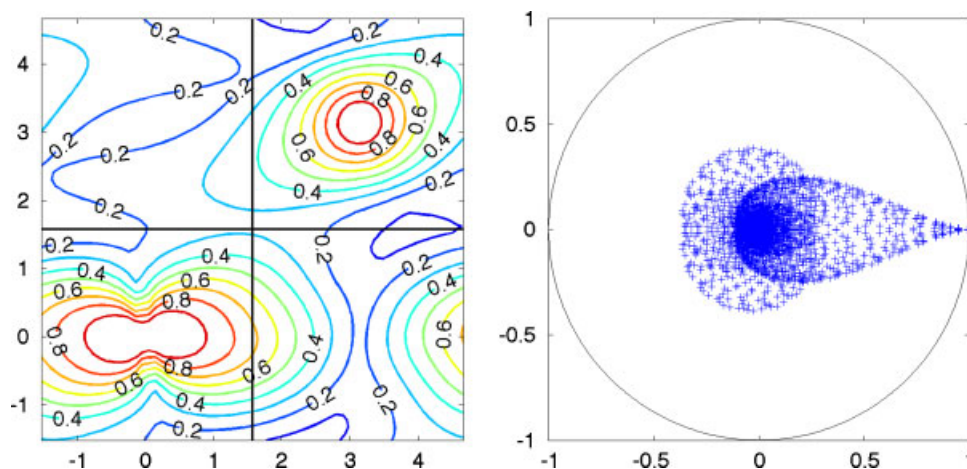


Figure 9. At left, the amplification factor for the element-wise (overlapping) Vanka smoother for the $Q_1 - P_0$ discretization of the Stokes equations on a mesh with $h = \frac{1}{64}$, as a function of the Fourier mode, θ . At right, the spectrum of the two-grid error-propagation operator using this smoother with bilinear interpolation for the velocities and piece-wise constant interpolation for pressure.

solvers with textbook multigrid efficiency. In this way, we extend the LFA and related software to arbitrary FE discretizations.

ACKNOWLEDGEMENTS

This research was supported by the European Community's Sixth Framework Programme, through a Marie Curie International Incoming Fellowship, MIF1-CT-2006-021927. The work of S. P. M. was partially supported by the National Science Foundation, under grant DMS-0811022.

REFERENCES

1. Bramble JH, Pasciak JE, Wang J, Xu J. Convergence estimates for multigrid algorithms without regularity assumptions. *Mathematics of Computation* 1991; **57**:23–45.
2. Hackbusch W. Convergence of multi-grid iterations applied to difference equations. *Mathematics of Computation* 1980; **34**:425–440.
3. McCormick SF. An algebraic interpretation of multigrid methods. *SIAM Journal on Numerical Analysis* 1982; **19**:548–560.
4. Falgout RD, Vassilevski PS, Zikatanov LT. On two-grid convergence estimates. *Numerical Linear Algebra with Applications* 2005; **12**:471–494. Also available as *LLNL Technical Report UCRL-JC-150807*.
5. Brandt A. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation* 1977; **31**:333–390.
6. Stüben K, Trottenberg U. Multigrid methods: fundamental algorithms, model problem analysis and applications. In *Multigrid Methods*, Hackbusch W, Trottenberg U (eds). Lecture Notes in Mathematics, vol. 960. Springer: Berlin, 1982; 1–176.
7. Yavneh I. On red–black SOR smoothing in multigrid. *SIAM Journal on Scientific Computing* 1996; **17**(1):180–192.
8. Wienands R, Joppich W. Practical Fourier analysis for multigrid methods. *Numerical Insights*, vol. 4. Chapman & Hall/CRC: Boca Raton, FL, 2005. With 1 CD-ROM (Windows and UNIX).
9. bin Zubair H, Oosterlee C, Wienands R. Multigrid for high-dimensional elliptic partial differential equations on non-equidistant grids. *SIAM Journal on Scientific Computing* 2007; **29**(4):1613–1636.
10. Wienands R, Oosterlee CW, Washio T. Fourier analysis of GMRES(m) preconditioned by multigrid. *SIAM Journal on Scientific Computing* 2000; **22**(2):582–603.
11. Gaspar F, Gracia J, Lisbona F. Fourier analysis for multigrid methods on triangular grids. *SIAM Journal on Scientific Computing* 2009; **31**(3):2081–2102.
12. Zhou G, Fulton SR. Fourier analysis of multigrid methods on hexagonal grids. *SIAM Journal on Scientific Computing* 2009; **31**(2):1518–1538.
13. Borzi A. High-order discretization and multigrid solution of elliptic nonlinear constrained optimal control problems. *Journal of Computational and Applied Mathematics* 2007; **200**(1):67–85.

14. Hemker PW, Hoffmann W, van Raalte MH. Fourier two-level analysis for discontinuous Galerkin discretization with linear elements. *Numerical Linear Algebra with Applications* 2004; **11**(5–6):473–491.
15. Arnold D, Falk R, Winther R. Multigrid in $H(\text{div})$ and $H(\text{curl})$. *Numerische Mathematik* 2000; **85**(2):197–217.
16. Vanka SP. Block-implicit multigrid solution of Navier–Stokes equations in primitive variables. *Journal of Computational Physics* 1986; **65**:138–158.
17. John V, Matthies G. Higher-order finite element discretizations in a benchmark problem for incompressible flows. *International Journal for Numerical Methods in Fluids* 2001; **37**(8):885–903.
18. Boonen T, van Lent J, Vandewalle S. Local Fourier analysis of multigrid for the curl–curl equation. *SIAM Journal on Scientific Computing* 2008; **30**(4):1730–1755.
19. Schöberl J, Zulehner W. On Schwarz-type smoothers for saddle point problems. *Numerische Mathematik* 2003; **95**(2):377–399.
20. Sivaloganathan S. The use of local mode analysis in the design and comparison of multigrid methods. *Computer Physics Communications* 1991; **65**:246–252.
21. Molenaar J. A two-grid analysis of the combination of mixed finite elements and Vanka-type relaxation. In *Multigrid Methods III*, Hackbusch W, Trottenberg U (eds). International Series of Numerical Mathematics, vol. 98. Birkhäuser: Basel, 1991; 313–324.
22. Manservigi S. Numerical analysis of Vanka-type solvers for steady Stokes and Navier–Stokes flows. *SIAM Journal on Numerical Analysis* 2006; **44**(5):2025–2056.
23. Larin M, Reusken A. A comparative study of efficient iterative solvers for generalized Stokes equations. *Numerical Linear Algebra with Applications* 2008; **15**(1):13–34.
24. Braess D. *Finite Elements* (2nd edn). Cambridge University Press: Cambridge, 2001.
25. Brenner S, Scott L. *The Mathematical Theory of Finite Element Methods*. Texts in Applied Mathematics, vol. 15. Springer: New York, 1994.
26. Brandt A. *Multigrid Techniques: 1984 Guide with Applications to Fluid Dynamics*. GMD–Studien, Nr. 85. Gesellschaft für Mathematik und Datenverarbeitung: St. Augustin, 1984.
27. Trottenberg U, Oosterlee CW, Schüller A. *Multigrid*. Academic Press: London, 2001.
28. Hiptmair R. Multigrid method for $H(\text{div})$ in three dimensions. *Electronic Transactions on Numerical Analysis* 1997; **6**:133–152.
29. Gaspar FJ, Gracia JL, Lisbona FJ, Oosterlee CW. Distributive smoothers in multigrid for problems with dominating grad–div operators. *Numerical Linear Algebra with Applications* 2008; **15**(8):661–683.
30. Bochev PB, Garasi CJ, Hu JJ, Robinson AC, Tuminaro RS. An improved algebraic multigrid method for solving Maxwell’s equations. *SIAM Journal on Scientific Computing* 2003; **25**(2):623–642.
31. Hu J, Tuminaro R, Bochev P, Garasi C, Robinson A. Toward an h -independent algebraic multigrid method for Maxwell’s equations. *SIAM Journal on Scientific Computing* 2006; **27**(5):1669–1688.
32. Reitzinger S, Schöberl J. An algebraic multigrid method for finite element discretizations with edge elements. *Numerical Linear Algebra with Applications* 2002; **9**(3):223–238.
33. Hiptmair R. Multigrid method for Maxwell’s equations. *SIAM Journal on Numerical Analysis* 1999; **36**(1):204–225.
34. Raviart PA, Thomas JM. A mixed finite element method for 2nd order elliptic problems. *Mathematical Aspects of Finite Element Methods*. Lecture Notes in Mathematics, vol. 606. Springer: Berlin, 1977; 292–315. *Proceedings of the Conference on Consiglio Nazionale delle Ricerche (C.N.R.)*, Rome, 1975.
35. Nédélec JC. Mixed finite elements in \mathbf{R}^3 . *Numerische Mathematik* 1980; **35**(3):315–341.
36. Vassilevski PS, Wang JP. Multilevel iterative methods for mixed finite element discretizations of elliptic problems. *Numerische Mathematik* 1992; **63**(4):503–520.
37. Arnold D, Falk R, Winther R. Preconditioning in $H(\text{div})$ and applications. *Mathematics of Computation* 1997; **66**(219):957–984.
38. Hiptmair R, Hoppe RHW. Multilevel methods for mixed finite elements in three dimensions. *Numerische Mathematik* 1999; **82**(2):253–279.
39. Brezzi F, Fortin M. *Mixed and Hybrid Finite Element Methods*. Springer Series in Computational Mathematics, vol. 15. Springer: New York, 1991.
40. Fortin M. Old and new finite elements for incompressible flows. *International Journal for Numerical Methods in Fluids* 1981; **1**(4):347–364.
41. Girault V, Raviart PA. *Finite Element Approximation of the Navier–Stokes Equations*. Lecture Notes in Mathematics, vol. 749. Springer: Berlin, 1979.
42. Brandt A, Dinar N. Multigrid solutions to elliptic flow problems. In *Numerical Methods for Partial Differential Equations*, Parter S (ed.). Academic Press: New York, 1979; 53–147.
43. Braess D, Sarazin R. An efficient smoother for the Stokes problem. *Applied Numerical Mathematics* 1997; **23**:3–20.
44. John V, Tobiska L. Numerical performance of smoothers in coupled multigrid methods for the parallel solution of the incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Fluids* 2000; **33**(4):453–473.
45. Wobker H, Turek S. Numerical studies of Vanka-type smoothers in computational solid mechanics. *Advances in Applied Mathematics and Mechanics* 2009; **1**:29–55.
46. Oppenheim A, Schaffer R. *Discrete-time Signal Processing*. Prentice-Hall Signal Processing Series. Prentice-Hall: Upper Saddle River, NJ, 1999.

47. Wittum G. On the robustness of ILU-smoothing. *SIAM Journal on Scientific Computing* 1989; **10**:699–717.
48. Brezina M, Cleary A, Falgout R, Henson V, Jones J, Manteuffel T, McCormick S, Ruge J. Algebraic multigrid based on element interpolation (AMGe). *SIAM Journal on Scientific Computing* 2000; **22**:1570–1592.
49. Vassilevski P. Sparse matrix element topology with application to AMG(e) and preconditioning. *Numerical Linear Algebra with Applications* 2002; **9**(6–7):429–444. Preconditioned robust iterative solution methods, *PRISM '01*, Nijmegen.
50. Chartier T, Falgout R, Henson V, Jones J, Manteuffel T, McCormick S, Ruge J, Vassilevski P. Spectral element agglomerate AMGe. *Domain Decomposition Methods in Science and Engineering XVI*. Lecture Notes in Computational Science and Engineering, vol. 55. Springer: Berlin, 2007; 513–521.
51. Kolev TV, Vassilevski PS. AMG by element agglomeration and constrained energy minimization interpolation. *Numerical Linear Algebra with Applications* 2006; **13**(9):771–788.
52. Niestegge A, Witsch K. Analysis of a multigrid Stokes solver. *Applied Mathematics and Computation* 1990; **35**(3):291–303.
53. Elman H, Silvester D, Wathen A. Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics. *Numerical Mathematics and Scientific Computation*. Oxford University Press: New York, 2005.