Multigrid with Nonstandard Coarse-level Operators and Coarsening factors

Chang Liu[1,2*] and William Henshaw[2]

[1*]Department of Mathematics, the University of Utah, 201 Presidents Circle, Salt Lake City, 84112, UT, USA.
[2]Department of Mathematical Sciences, Rensselaer Polytechnic Institute, 110 Eighth Street, Troy, 12180, NY, USA.

*Corresponding author(s). E-mail(s): liukamala@math.utah.edu;
Contributing authors: henshw@rpi.edu;

Abstract

We consider the numerical solution of Poisson's equation on structured grids using geometric multigrid with nonstandard coarse grids and coarse-level operators. We are motivated by the problem of developing high-order accurate numerical solvers for elliptic boundary value problems on complex geometry using overset grids. For flexibility in grid generation, we would like to consider lower-order accurate coarse-level approximations, and coarsening factors other than two. We show that *second-order accurate coarse-level approximations* are very effective for fourth- or sixth-order accurate fine-level finite difference discretizations. We study the use of different Galerkin and non-Galerkin coarse-level operators. Using local Fourier analysis (LFA) we choose the smoothing parameter $\boldsymbol{\omega}$ and the coarse-level operators to optimize the overall multigrid convergence rate. We show that the results based on LFA for periodic problems also hold for more general boundary conditions provided these are discretized using *compatibility conditions*. Numerical results for Poisson's equation on a sample overset grid show that our multigrid solver is many times faster, and uses less memory, than selected Krylov solvers and an algebraic multigrid solver. We also study *grid coarsening by a general factor* and show that good convergence rates are retained for a range of coarsening factors near two. We ask the question of which coarsening factor leads to the most efficient multigrid algorithm.

Keywords: multigrid, nonstandard coarsening, overset grids, high-order accuracy

1 Introduction

We are motivated by the solution of elliptic partial differential equations (PDEs) and boundary value problems (BVPs) on complex geometry using composite overlapping grids. As shown in Figure 1, a composite or "overset" grid consists of multiple overlapping *component* grids used to cover a complex geometry [1]. Each component grid is a logically-rectangular structured grid, defined by a mapping from the unit-square or unit-cube to the physical space. Solution values are interpolated at internal *interpolation boundaries* where two component grids overlap. These grids have been shown to be very effective at efficiently solving a wide class of problems including low speed flows [2], high-speed flows [3], fluid structure interactions [4] and electromagnetics [5].The approach is especially useful for problems with moving geometry. Multigrid methods are well-known to provide excellent iterative solvers for elliptic problems, and have been shown to be quite effective for solving problems on overset grids [1, 6], being fast and efficient and having low startup costs as grids change in a moving geometry scenario.

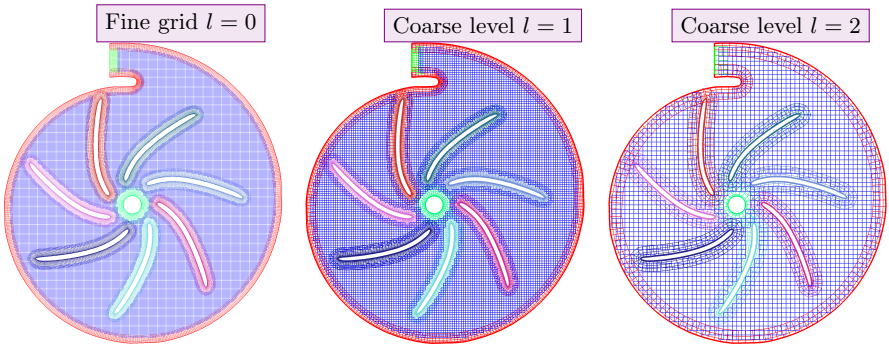

Figure 1 Overset grid for a centrifugal pump, fine grid and two multigrid coarsenings. As the mesh is refined further the majority of grid points will belong to the blue Cartesian background grid. Ideally, multigrid convergence rates for this overset grid should approach the convergence rates and efficiency of a single Cartesian grid.

We are particularly interested in *high-order accurate schemes* on overset grids, and we wish to develop high-order accurate elliptic solvers that are nearly as efficient as second-order accurate ones. A fourth-order accurate scheme, for example, although somewhat more expensive due to a larger stencil, will generally be much more efficient than a second-order scheme in obtaining a solution to a given accuracy. On the other hand, it is possible to obtain multigrid convergence rates for a fourth-order accurate scheme that are very similar to those for a second-order accurate one [7].

In practical problems, the generation of high-quality grids is one of the most important steps in solving a boundary value problem using multigrid. It is easy to construct a single Cartesian grid that can be coarsened many times by a factor of two with standard coarsening. For flexibility in overset grid

generation, however, we do not want to place such constraints on the number of grid points in each individual component grid. Thus we would like to study whether using *more general coarsening factors*, for example in a neighborhood of two, will still retain good convergence rates and efficiency. Furthermore, a primary technical challenge in applying multigrid methods to overset grids is the automatic generation of coarser levels [6]. This is more prominent for high-order accurate discretizations, which require multiple layers of interpolation points to support wide stencils and sufficient overlap between component grids. We would like to relax these grid generation requirements on coarse levels by using *lower-order accurate approximations*, so that more valid coarse levels can be generated.

Our overall goal is to achieve fast multigrid convergence when solving elliptic boundary value problems on overset grids. We approach this goal by first studying a model problem, and optimize the multigrid convergence using *local Fourier analysis* (LFA). The results of this model problem analysis, such as optimal smoothing parameters and choice of coarse-level operators, are then applied to the more general problem on overset grids. One of the important aspects of the multigrid algorithm in service of achieving fast convergence similar to LFA, is the discretization of boundary conditions. For general boundary value problems it is well-known that multigrid convergence rates can be significantly impacted by the treatment of the boundary conditions, especially for higher-order accurate schemes [8]. We show that the results based on LFA also hold for more general boundary conditions provided these are discretized using *compatibility boundary conditions*.

In this paper, we focus on the solution of the *model problem of Poisson's equation on Cartesian grids*, where we can make use of local Fourier analysis to study properties of our multigrid algorithms. The solution to Poisson's equation is important in many applications such as incompressible fluids, incompressible elasticity, and electromagnetics. A typical, efficient overset grid would often have the majority of its grid points in *Cartesian background grids*. A well-designed multigrid algorithm for overset grids might be expected to have convergence rates similar to those on a single Cartesian grid, and thus fast algorithms for Cartesian grids are highly desired. We present some sample results for an overset grid to illustrate the applicability of the LFA results to the more general boundary value problem. Further model problem LFA study of *anisotropic* Poisson's equation, and more detailed results for overset grids, in particular for grids with stretching where line or plane smoothers are utilized, are left for future work.

Three primary topics are considered in this paper. First, we investigate the use of nonstandard coarse-level operators for high-order accurate schemes. We show that second-order accurate coarse-level approximations are very effective when fourth- and sixth-order accurate discretizations are used on the fine level. We study the use of different Galerkin and non-Galerkin coarse-level operators, and assess their influence on the multigrid convergence rate as well as their computational cost, by studying the *effective convergence rate*. Based on these

results we also propose more flexible and simpler Galerkin-like operators for coarse levels. We study the use of an over-relaxed *red-black smoother* with a relaxation parameter ω, which is commonly used and highly effective for the model problem. Using local Fourier analysis we choose the value ω as well as the coarse-level operators to optimize the overall multigrid convergence rate, rather than the more common approach of choosing ω to optimize the smoothing rate in isolation.

Second, we show that a good way to discretize boundary conditions is to use compatibility boundary conditions (CBCs). For a model boundary value problem we show how the use of CBCs leads to discrete eigenfunctions that mimic the continuous ones, which means local Fourier analysis still approximately applies. In contrast, the use of more traditional, standard one-sided approximations leads to a degradation of the multigrid convergence rate. In addition, we show that using CBCs, the LFA results and optimizations, in particular the use of lower-order coarse-level operators, can be generalized to a sample overset grid on general geometry, achieving multigrid convergence rates comparable to LFA. Our multigrid solver is shown to outperform some other iterative solvers, including a standard algebraic multigrid (AMG) solver, on the sample overset grid, both in CPU time and memory usage.

Lastly, we study grid coarsening by general factors other than two, motivated by the need of flexible grid generation for multi-level overset grids. It is desired that good effective convergence rates can be obtained when coarsening by a general factor r over a fairly large range about two. In particular, we ask the question of which coarsening factor leads to the most efficient multigrid algorithm. Local Fourier analysis is used to study the properties of factor-r coarsening.

Multigrid is a well established field with a vast literature, see for example the textbooks [8–12] and the references therein. There has been much active research in algebraic multigrid methods [13–17]. Multigrid for overset grids has been considered in [18–21]. The use of over-relaxed red-black smoothers, in particular choosing the parameter ω to optimize the smoothing factor, has been studied in [22]. Multigrid convergence rates derived from local Fourier analysis are shown to match exactly those for a model boundary value problem in [23]. The development of multigrid algorithms for high-order accurate discretizations is often accomplished by adding correction terms to a second-order accurate scheme, for instance, via defect-correction [24] or τ-extrapolation [25]. We also note that the order of accuracy of the transfer operators has been studied in [26], in terms of how it affects multigrid convergence.

The remainder of this paper is structured as follows. We first specify the model problem and give a brief overview of multigrid and local Fourier analysis in Section 2. In Section 3, we consider the use of lower-order accurate coarse-level operators for higher-order accurate fine-level discretizations. In Section 4, for a model problem, we compare the use of CBCs to one-sided approximations of boundary conditions and their effect on multigrid convergence. Multigrid

convergence results are shown for a sample overset grid, and a performance comparison is made to an AMG solver and two Krylov solvers. Coarsening by factors other than two is studied in Section 5. Finally, Section 6 summarizes and concludes the paper.

2 Problem specification and multigrid overview

In general, we are interested in solving an elliptic boundary value problem on some domain in $d = 1,\ 2,\ 3$ space dimensions with a finite-difference discretization. It is known that the main properties of a well-designed multigrid algorithm should not depend strongly on the particular domain or boundary conditions. Therefore, in the analysis of this paper we mainly consider the discretized domain to be the infinite Cartesian grid

$$\mathcal{G}_h = \left\{ \mathbf{x} = \mathbf{j}\, h : \ \mathbf{j} \in \mathbb{Z}^d \right\} \tag{1}$$

with spacing h. We focus on the important *model problem of Poisson's equation in two dimensions*

$$Lu = f, \quad L = -\Delta = -(\partial_x^2 + \partial_y^2), \tag{2}$$

and its discretization

$$L_h u_h = f_h. \tag{3a}$$

In particular, on (1) ($d = 2$) we consider the standard finite-difference discretization of the negative Laplacian L to second- or fourth-order accuracy (using stencil notation [8])

$$L_h^{(2)} = \frac{1}{h^2} \begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}_h, \quad L_h^{(4)} = \frac{1}{12h^2} \begin{bmatrix} & & 1 & & \\ & & -16 & & \\ 1 & -16 & 60 & -16 & 1 \\ & & -16 & & \\ & & 1 & & \end{bmatrix}_h. \tag{3b}$$

Consider a multi-level algorithm for solving a general discretized boundary value problem represented as $L_h u_h = f_h$. A geometric multigrid algorithm to solve this problem is often based on four key components:

i) a sequence of grids (levels) of increasing coarseness;
ii) an iterative procedure called a *smoother*, that is effective at reducing high-frequency components of the error or residual (for example, common smoothers are based on the Jacobi or Gauss-Seidel iteration);
iii) fine-to-coarse (restriction) and coarse-to-fine (interpolation) operators, that transfer a solution between a fine grid and a coarse grid;
iv) coarse-level operators that approximate the fine-level operator L_h.

We consider the defect-correction multigrid algorithm, where the error equation is solved recursively on the coarser level. Each iteration of a multigrid

solver consists of a multi-level *cycle*. In particular, consider an $(l_{\max}+1)$-level cycle with grids of spacing h_l $(h_0 = h)$, $l = 0,\ 1,\ \cdots,\ l_{\max}$, on each level. The *smoothing operator* on the l-th level is denoted as S_{h_l} (line 3), and restriction and interpolation operators between the l-th and $(l+1)$-st levels are denoted as $I_{h_l}^{h_{l+1}}$ (line 5) and $I_{h_{l+1}}^{h_l}$ (line 11), respectively. The coarse-level error equations are not to be solved until convergence, but only approximately by a few iterations, defined by a cycle parameter γ (line 9). In particular, the multigrid cycle is called a V$[\nu_1,\ \nu_2]$ cycle if $\gamma = 1$, and a W$[\nu_1,\ \nu_2]$ cycle if $\gamma = 2$, where the parameters ν_1 and ν_2 indicate the number of pre- and post-smoothing sweeps per cycle. The multigrid solver with multi-level cycles is presented in pseudo-code in Algorithm 1.

Algorithm 1 $(l+1)$-level cycles for $L_h u_h = f_h$

1: **function** $u_{h_l} = MG(f_{h_l},\ L_{h_l};\ u_{h_l}^{(0)},\ l,\ N_{\text{iter}})$
2: **for** $n = 0,\ 1,\ \cdots,\ N_{\text{iter}} - 1$ **do**
▷ *Peform N_{iter} iterations (until convergence)*
3: $\bar{u}_{h_l}^{(n)} \xleftarrow{S_{h_l}^{\nu_1}} u_{h_l}^{(n)}$ ▷ *Pre-smoothing*
4: $\bar{d}_{h_l}^{(n)} = f_{h_l} - L_{h_l}\bar{u}_{h_l}^{(n)}$
5: $\bar{d}_{h_{l+1}}^{(n)} = I_{h_l}^{h_{l+1}}\bar{d}_{h_l}^{(n)}$ ▷ *Restriction*
6: **if** $l+1 = l_{\max}$ **then**
7: $\bar{v}_{h_{l+1}}^{(n)} = L_{h_{l+1}}^{-1}\bar{d}_{h_{l+1}}^{(n)}$ ▷ *Exact solve, coarsest level*
8: **else**
9: $\bar{v}_{h_{l+1}}^{(n)} = MG(\bar{d}_{h_{l+1}}^{(n)},\ L_{h_{l+1}};\ 0,\ l+1,\ \gamma)$ ▷ *Coarse-level solve*
10: **end if**
11: $\tilde{v}_{h_l}^{(n)} = I_{h_{l+1}}^{h_l}\bar{v}_{h_{l+1}}^{(n)}$ ▷ *Interpolation*
12: $\bar{u}_{h_l}^{(n+1)} = \bar{u}_{h_l}^{(n)} + \tilde{v}_{h_l}^{(n)}$
13: $u_{h_l}^{(n+1)} \xleftarrow{S_{h_l}^{\nu_2}} \bar{u}_{h_l}^{(n+1)}$ ▷ *Post-smoothing*
14: **end for**
15: $u_{h_l} = u_{h_l}^{(n+1)}$
16: **end function**

Given a current solution $u_h^{(n)}$ (indexed by an iteration number n), as an approximation of the solution u_h to $L_h u_h = f_h$, we will consider the action of each component in a multigrid cycle on the error and the residual

$$v_h^{(n)} \overset{\text{def}}{=} u_h - u_h^{(n)}, \qquad d_h^{(n)} \overset{\text{def}}{=} f_h - L_h u_h^{(n)}. \tag{4}$$

For example, the smoother's effect on the current error is depicted by the smoothing operator S_h. In particular, with ν_1 pre-smoothing sweeps (on the fine level) in a multigrid cycle, the error reduction, from $v_h^{(n)}$ to $\bar{v}_h^{(n)} \equiv u_h - \bar{u}_h^{(n)}$,

is given by $\bar{v}_h^{(n)} = S_h^{\nu_1} v_h^{(n)}$; while ν_2 post-smoothing sweeps give the error reduction $v_h^{(n+1)} = S_h^{\nu_2} \bar{v}_h^{(n+1)}$, from $\bar{v}_h^{(n+1)} \equiv u_h - \bar{u}_h^{(n+1)}$ to $v_h^{(n+1)} \equiv u_h - u_h^{(n+1)}$. Most importantly, the iteration operator of a multigrid cycle, denoted by M_h, relates the error between successive iterations as

$$v_h^{(n+1)} = M_h \, v_h^{(n)}. \tag{5}$$

Thus the asymptotic convergence rate (ACR) of the multigrid cycle

$$\text{ACR} \stackrel{\text{def}}{=} \lim_{n\to\infty} \frac{\|v_h^{(n+1)}\|_h}{\|v_h^{(n)}\|_h} \tag{6}$$

is given by the spectral radius of the multigrid iteration operator M_h, denoted as ρ,

$$\text{ACR} = \rho \stackrel{\text{def}}{=} \sigma(M_h). \tag{7}$$

In this paper, we denote $\sigma(\cdot)$ as the spectral radius of an operator on a grid-function space, or a square matrix.

Local Fourier analysis (LFA) is a useful tool for analyzing multigrid algorithms, in particular their convergence properties [8]. When the boundary conditions are properly dealt with numerically, the most important properties of a multigrid cycle are "local", in that its convergence is not affected much by the boundaries, as if on an infinite domain (see Section 4). This is why the infinite grid (1) is introduced; the discrete Fourier modes

$$\phi_h(\mathbf{x},\ \boldsymbol{\theta}) \stackrel{\text{def}}{=} e^{i\boldsymbol{\theta}\cdot\frac{\mathbf{x}}{h}}, \quad \mathbf{x} \in \mathcal{G}_h,\ \boldsymbol{\theta} \in \Theta \stackrel{\text{def}}{=} [-\pi,\ \pi)^d, \tag{8}$$

are thus eigenfunctions for any linear constant-coefficient operator that maps a grid function on $\mathcal{G}_h$ to another. Here $\boldsymbol{\theta}$ is a parameter representing the frequency. Based on the orthogonal Fourier decomposition of any grid function, we can analyze the behavior of the operators involved in a multigrid cycle (such as S_h) by analyzing their operation on each Fourier mode.

Since a multi-level cycle aims to have a convergence rate comparable to that of a two-level cycle, much of the analysis focuses on two levels. Given the operator L_h on the fine grid $\mathcal{G}_h$, consider a coarse grid $\mathcal{G}_H$, with grid-spacing H, and a coarse-level operator denoted as L_H. For example, by *standard coarsening*, $\mathcal{G}_H = \left\{\mathbf{x} = \mathbf{j}\,H :\ \mathbf{j} \in \mathbb{Z}^d\right\}$, with $H = 2h$. The coarse-level operator L_H is said to have the *Galerkin* property if $L_H = I_h^H L_h I_H^h$, assuming the transfer operators are adjoint to each other, $I_H^h = (I_h^H)^*$. Suppose the coarse-level solve is exact, then the iteration operator for a $[\nu_1,\ \nu_2]$ cycle is given by

$$M_h \equiv M_h^H = S_h^{\nu_2} K_h^H S_h^{\nu_1}, \tag{9}$$

where K_h^H is the coarse-level correction operator

$$K_h^H = I_h - I_H^h L_H^{-1} I_h^H L_h, \tag{10}$$

with the restriction operator I_h^H and interpolation operator I_H^h. In a general multi-level cycle with grid-spacings $(h_0,\ h_1,\ \cdots,\ h_{l_{\max}})$ $(h_0 = h,\ l_{\max} > 1)$, the coarse-level correction operator on the finest level is given by

$$K_h = I_h - I_{h_1}^h \tilde{L}_{h_1}^{-1} I_h^{h_1} L_h, \tag{11}$$

where the coarse level solve $\tilde{L}_{h_1}^{-1}$ is in turn obtained recursively by an $l_{\max}$-level cycle. The multi-level iteration operator is then given by

$$M_h = S_h^{\nu_2}\, K_h\, S_h^{\nu_1}. \tag{12}$$

Through local Fourier analysis, we can represent the multigrid operators $S_h,\ K_h$, and in turn M_h, in Fourier space. In particular, these operators are represented as small matrices, since their eigenspaces are constructed from a basis of only a few Fourier modes. Given the matrix representation $\hat{M}_h(\boldsymbol{\theta})$ of the multigrid iteration operator M_h in Fourier space, we can thus find the asymptotic convergence rate ACR $= \rho$ (7), by

$$\rho = \sigma(M_h) = \sup_{\boldsymbol{\theta}\in\Theta} \sigma(\hat{M}_h(\boldsymbol{\theta})), \tag{13}$$

where the spectral radius $\sigma(\hat{M}_h(\boldsymbol{\theta}))$ of the *matrix* $\hat{M}_h(\boldsymbol{\theta})$ is given by its largest eigenvalue in magnitude. In practice we compute the LFA spectral properties approximately using the software from Wienands [27][1]. In particular, the spectral radius (13) is approximated by

$$\sigma(M_h) \approx \max_{\boldsymbol{\theta}\in\Theta_h} \sigma(\hat{M}_h(\boldsymbol{\theta})) \tag{14}$$

over a discrete set of frequencies $\boldsymbol{\theta} \in \Theta_h$. Here $\boldsymbol{\Theta}_h$ denotes a discretization of $\Theta = [-\pi,\ \pi]^d$ (e.g. using 64 points in each direction).

The core idea of a multigrid algorithm comes from a combination of the effect of a smoother, and a coarse-level correction; the coarse-level correction aims to tackle the low-frequency Fourier components of the error, whereas the smoother aims to reduce the high-frequency ones. The Fourier modes ϕ_h (8) on $\mathcal{G}_h$, are categorized as low-frequency, if they can be well-represented on the coarse grid $\mathcal{G}_H$; otherwise they are high-frequency. For example, with *standard coarsening*, $\phi_h(\cdot,\ \boldsymbol{\theta})$ is low-frequency if $\boldsymbol{\theta} \in \Theta^{\text{low}} = [-\frac{\pi}{2},\ \frac{\pi}{2})^d$, and high-frequency modes if $\boldsymbol{\theta} \in \Theta^{\text{high}} \stackrel{\text{def}}{=} \Theta \setminus \Theta^{\text{low}}$. Consider an 'idealized' coarse-level correction operator Q_h^H, instead of K_h^H (10) or K_h (11), which eliminates

[1]The LFA results presented in Section 3, in particular, are obtained using the software from Wienands [27], which we extended for sixth-order accuracy as well as for additional choices of coarse-level operators. The LFA results in Section 5 are produced using Matlab codes we developed.

the low-frequency modes completely, while keeping the high-frequency modes intact. That is, $Q_h^H\,\phi_h(\cdot,\ \boldsymbol{\theta}) = 0$, for $\boldsymbol{\theta} \in \Theta^{\text{low}}$; $Q_h^H\,\phi_h(\cdot,\ \boldsymbol{\theta}) = 1$, for $\boldsymbol{\theta} \in \Theta^{\text{high}}$. The resulting theoretical multigrid cycle will have an iteration operator $S_h^{\nu_2} Q_h^H\, S_h^{\nu_1}$, instead of M_h (9) or (12), which then characterizes the smoother's ability in reducing high-frequency components of the error. Accordingly, the *smoothing factor* of a multigrid cycle is defined as

$$\mu \overset{\text{def}}{=} [\sigma(S_h^{\nu_2} Q_h^H S_h^{\nu_1})]^{\frac{1}{\nu}}, \tag{15}$$

where $\nu = \nu_1 + \nu_2$, and again, $\sigma(\cdot)$ denotes the spectral radius of an operator. Multigrid convergence can be greatly improved by tuning the relaxation parameter ω of the smoother. For example, we usually show results for multigrid cycles with an *over-relaxed red-black* smoother, which is very effective for the discrete Laplace operator (and also effective for the anisotropic Laplace operator with moderate anisotropy). The factor μ^ν is used as a reference convergence rate for a multigrid cycle, which measures the asymptotic convergence rate of a theoretic cycle with an ideal coarse-level correction. Thus the smoothing parameter ω is often chosen in textbooks to optimize the smoothing factor μ (15). Through local Fourier analysis, however, we find that we can generally achieve faster convergence if we instead choose the value of ω to directly optimize the asymptotic convergence rate ρ (7) of the actual multigrid cycle. This is explained specifically for the model problem discretized to fourth-order accuracy in Section 3.

In this paper, the theoretical LFA results are often compared to computations on the finite domain $[0,\ 1]^d$ with periodic boundaries, discretized on the fine grid

$$\mathcal{G}_h = \left\{ \mathbf{x}_\mathbf{j} = \mathbf{j}\, h :\ \mathbf{j} \in \{0,\ 1,\ \cdots,\ N\}^d \right\}, \quad h = \frac{1}{N}, \tag{16}$$

with $N \in \mathbb{N}$. We present the computational results in terms of the convergence rate (CR) and *effective convergence rate* (ECR). The convergence rate is given by

$$\text{CR} \overset{\text{def}}{=} \frac{\|d_h^{(n+1)}\|_h}{\|d_h^{(n)}\|_h}, \tag{17}$$

which implicitly depends on the iteration number n, but we usually choose n to be sufficiently large so that the reduction of error and residual is close to the ACR $= \rho$. Note that CR approaches ACR as n gets large,

$$\lim_{n\to\infty} \text{CR} = \lim_{n\to\infty} \frac{\|d_h^{(n+1)}\|_h}{\|d_h^{(n)}\|_h} = \sigma(L_h M_h L_h^{-1}) = \sigma(M_h) = \rho = \text{ACR}. \tag{18}$$

The effective convergence rate is a normalized measure of the convergence rate of a multigrid cycle, defined by

$$\text{ECR} \overset{\text{def}}{=} \text{CR}^{\frac{1}{\text{WU}}}, \tag{19}$$

which takes into account the computational cost in each cycle, measured in work-units (WU). One one work-unit is defined to be the number of floating-point operations (FLOPS)[2] needed for a single Jacobi smoothing step for the difference equation on the finest level. (For example, for the two-dimensional model problem on a Cartesian grid, a V[2, 1] multi-level cycle can be estimated to have WU ≈ 5, for both the second- and fourth-order accurate schemes.) In particular, note that for the Jacobi iteration, its ECR is equal to its CR. The ECR can be interpreted as the convergence rate that a Jacobi (or Gauss-Seidel) iterative solver would need to achieve, per iteration, in order to match the convergence of the multigrid cycle. A good multigrid algorithm may be expected to have an ECR in the range 0.5 to 0.8, independent of h (while the ECR for the Jacobi iteration alone is of order $1-O(h^2)$ and thus approaches one as $h \to 0$). In this paper, we often show the consistency of the CR, computed from the convergence history of applying a multigrid solver to a numerical test on a finite grid, with the asymptotic convergence rate ACR $= \rho$, obtained from local Fourier analysis, in accordance with (18) (for example, in Figure 4 and Figure 6). We also show how the ACR $= \rho$, and ECR, depend on the smoothing parameter ω in LFA results. (Note that, for LFA results, the ECR (19) can be defined using the ACR, instead of the CR; the actual definition of the ECR is normally clear from the context.) It is of particular interest to see how far away the optimal ω (to minimize the convergence rate) is from the default value $\omega = 1$. (For example, in Figure 2.)

3 High-order accurate discretizations with second-order accurate coarse-level operators

In this section, we consider using lower-order accurate operators on coarse levels when solving the model problem of Poisson's equation to fourth- and sixth-order accuracy. This will be a desirable approach to use in practice, since it is likely to be less expensive, and more importantly, it eases the process of generating coarser levels for overset grids. We will show, through LFA results, that the use of lower-order accurate coarse-level operators is very effective and generally results in convergence rates comparable to, or better than, those obtained using high-order accurate coarse-level solves. In the next Section 4, we will show that the idea can be applied to overset grids by presenting some sample results.

There is a simple heuristic argument that suggests why the coarse-level solves can be obtained using lower-order accurate approximations. In Algorithm 1, the error equation on the first coarse level only needs to be solved *approximately*. The multigrid algorithm uses a coarse-level error equation to obtain an estimate of the current error, which is served as a *correction* to the current solution. We can gauge how accurate this coarse-level correction needs to be, so as to not degrade the performance of the overall cycle. The

[2]We acknowledge that FLOPS do not tell the whole story on modern computer architectures and so the results presented here are more of a rough guideline.

error reduction per cycle is approximately the asymptotic convergence rate ACR $= \rho$ (7), which is roughly in the range 10^{-2} to 10^{-1} in a typical (good) multigrid algorithm. Thus a typical coarse-level solve at the first coarse level ($l = 1$, obtained recursively with the multigrid cycle if $l_{\max} > 1$) provides a correction $\tilde{v}_h^{(n)}$ to the fine grid (at level $l = 0$), which, as an estimate to the current error $\bar{v}_h^{(n)} \equiv u_h - \bar{u}_h^{(n)}$, has a relative error of roughly the size of ρ, 10^{-2} to 10^{-1}. It therefore seems reasonable to suppose that a coarse-level correction with a relative error of such size can be provided by a lower-order (e.g. second-order) accurate approximation on the coarse level instead. Here we show this to be true for fourth- and sixth-order accurate fine-level discretizations of the model problem.

Consider the model problem (3a) with the fourth-order accurate operator $L_h = L_h^{(4)}$ given in (3b). We consider the red-black Gauss-Seidel smoother with an over-relaxation parameter ω, which consists of two partial Gauss-Seidel steps on the red and black points, $\mathcal{G}_h^R = \{\mathbf{x_j} = h\,[j_1,\ j_2]^T :\ j_1{+}j_2 \in \mathbb{Z}_1\}$, $\mathcal{G}_h^B = \mathcal{G} \setminus \mathcal{G}_h^R$ (where $\mathbb{Z}_1$ denotes odd integers), respectively. The smoothing operator is given by $S_h = S_h^B\ S_h^R$, where

$$S_h^R = \begin{cases} S_h^{GS} & \text{on} \quad \mathcal{G}_h^R \\ I_h & \text{on} \quad \mathcal{G}_h^B \end{cases}, \quad S_h^B = \begin{cases} S_h^{GS} & \text{on} \quad \mathcal{G}_h^B \\ I_h & \text{on} \quad \mathcal{G}_h^R \end{cases}, \tag{20}$$

in which $S_h^{GS}(\omega) = -(L_h^+)^{-1} L_h^-$, with $L_h^+ = \frac{1}{12h^2} \begin{bmatrix} & & 1 & & \\ & & 0 & & \\ 1 & 0 & 60\frac{1}{\omega} & 0 & 0 \\ & & 0 & & \\ & & 0 & & \end{bmatrix}_h$, $L_h^- = L_h - L_h^+$.

Consider a coarse grid $\mathcal{G}_H$ from standard coarsening, $H = 2h$. Typical transfer operators to use are the full-weighting restriction and linear interpolation operators

$$I_h^H = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h^{2h}, \quad I_H^h = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_{2h}^{h}, \tag{21}$$

of second-order accuracy. The natural choice of the coarse-level operator is either the non-Galerkin operator with the same stencil as the fine-level L_h, or the Galerkin coarse-level operator $L_H = I_h^H L_h I_H^h$. The main idea of this section is to consider coarse-level operators induced from the *second-order accurate* fine-level operator $L_h^{(2)}$ given in (3b), instead of $L_h = L_h^{(4)}$. That is, the non-Galerkin operator with the same stencil as $L_h^{(2)}$ and the "Galerkin"[3] operator given by $L_H = I_h^H L_h^{(2)} I_H^h$. Note that this makes the coarse-level problem

[3]Here "Galerkin" is in quotations since the coarse-level operator does not come from the actual operator on the fine level $L_h = L_h^{(4)}$.

$L_H\, v_H = d_H$ to be of second-order accuracy, which requires less complexity in both the grid and the solver on the coarse level.

Besides the order of accuracy of the coarse-level operators, for higher-order accurate fine-level discretizations, it may also be a natural consideration to use higher-order accurate transfer operators. Thus for completeness we also consider the cubic interpolation operator and its adjoint restriction operator,

$$I_h^H = \frac{1}{1024}\begin{bmatrix} 1 & 0 & -9 & -16 & -9 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -9 & 0 & 81 & 144 & 81 & 0 & -9 \\ -16 & 0 & 144 & 256 & 144 & 0 & -16 \\ -9 & 0 & 81 & 144 & 81 & 0 & -9 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -9 & -16 & -9 & 0 & 1 \end{bmatrix}_h^H, \tag{22a}$$

$$I_H^h = \frac{1}{256}\begin{bmatrix} 1 & 0 & -9 & -16 & -9 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -9 & 0 & 81 & 144 & 81 & 0 & -9 \\ -16 & 0 & 144 & 256 & 144 & 0 & -16 \\ -9 & 0 & 81 & 144 & 81 & 0 & -9 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -9 & -16 & -9 & 0 & 1 \end{bmatrix}_H^h, \tag{22b}$$

which are of fourth-order accuracy, in place of (21). In this case, we still consider the fourth-order accurate non-Galerkin coarse-level operator with the same stencil as the fine-level $L_h = L_h^{(4)}$, as well as the Galerkin operator constructed from the fourth-order accurate transfer operators (22).

In Figure 2 (top) we plot the asymptotic convergence rate ACR $= \rho$, of two-level V[1, 1] and three-level V[2, 1] cycles, as typical examples, versus the smoothing parameter ω[4]. We compare the various choices of coarse-level correction operators $K_h^H = I_h - I_H^h L_H^{-1} I_h^H L_h$ discussed above, that is, combinations of transfer and coarse-level operators (Table 1). We also show μ^ν, the expected ACR based on the smoothing rate with an ideal coarse-level correction, for comparison. We can observe that the fourth-order accurate transfer and coarse-level operators give ACRs that are closest to μ^ν, which reflects the fact that the resulting coarse-level correction operators are higher-order accurate approximations to the ideal Q_h^H. On the other hand, this is the most computationally expensive combination. We see that the second-order accurate coarse-level operators, along with second-order accurate transfers, do no worse than the fourth-order accurate alternatives. In fact, by adjusting the value of the smoothing parameter ω, we can usually achieve better ACRs. For example, from Figure 2 (top right), for a three-level V[2, 1] cycle, to minimize the smoothing rate μ (dashed black line represents μ^ν), one may choose $\omega \approx 0.97$ (circled in blue), whereas to minimize ρ given the second-order Galerkin coarse-level operator (yellow curve with cross), for example, the optimal $\omega \approx 1.1$

[4]For multi-level cycles the same ω for smoothing is used on every level.

(circled in purple). Thus with the second-order Galerkin coarse-level operator (yellow curve with cross), the optimal ACR is $\rho(\omega^* \approx 1.1) \approx 0.02$; in comparison, if ω is tuned to minimize μ, one would get ACR $= \rho(\omega \approx 0.97) \approx 0.04$. This supports our claim that faster convergence can be achieved by tuning ω to optimize ρ instead of μ, and with ω we can achieve comparable or even better ACRs with lower-order coarse-level operators.

Table 1 Transfer and coarse-level operators for fourth- or sixth-order fine-level operator L_h. These operators are represented for *the first coarse level*; the operators for coarser levels are defined recursively in the same way.

L_h	transfers I_h^H, I_H^h	coarse-level operators L_H		
$L_h^{(4)}$	$I^{(2)}$ 2nd-order accurate	$nG^{(4)}$	$L_H = L_H^{(4)}$ 4th-order accurate	non-Galerkin
		$nG^{(2)}$	$L_H = L_H^{(2)}$ 2nd-order accurate	
		G	$L_H = I_h^H L_h I_H^h$ (constructed from $L_h = L_h^{(4)}$ and $I^{(2)}$)	Galerkin
		$G^{(2)}$	$L_H = I_h^H L_h^{(2)} I_H^h$ 2nd-order accurate (constructed from $L_h^{(2)}$ and $I^{(2)}$)	'Galerkin'
	$I^{(4)}$ 4th-order accurate	$nG^{(4)}$	$L_H = L_H^{(4)}$ 4th-order accurate	non-Galerkin
		G	$L_H = I_h^H L_h I_H^h$ (4th-order accurate, constructed from $L_h = L_h^{(4)}$ and $I^{(4)}$)	Galerkin
$L_h^{(6)}$	$I^{(2)}$ 2nd-order accurate	$nG^{(6)}$	$L_H = L_H^{(6)}$ 6th-order accurate	non-Galerkin
		$nG^{(2)}$	$L_H = L_H^{(2)}$ 2nd-order accurate	
		G	$L_H = I_h^H L_h I_H^h$ (constructed from $L_h = L_h^{(6)}$ and $I^{(2)}$)	Galerkin
		$G^{(2)}$	$L_H = I_h^H L_h^{(2)} I_H^h$ 2nd-order accurate (constructed from $L_h^{(2)}$ and $I^{(2)}$)	'Galerkin'

From Figure 2 (top) we can make out the optimal ω for each choice of coarse-level correction operator K_h^H for a multigrid cycle. In Figure 3 (left) we record the optimal ACRs for the various choices of transfer and coarse-level operators at their respective optimal ω. It can be seen that the use of second-order accurate coarse-level operators gives ACRs comparable to those with the fourth-order accurate ones. Note that the LFA results in Figure 2 give good estimates for the ACRs of general multi-level V and W cycles in practice, with the V cycles being estimated from the corresponding three-level results, and the W cycles from the two-level results. This correspondence between cycle type and number of levels is well-known [28]. Thus from these LFA estimated ACRs we can find the optimal values of the smoothing parameter ω to choose for the corresponding multi-level cycles.

A more fair comparison between the multigrid cycles will also take into account the computational cost (WU[5]) and therefore in Figure 3 (right) we

[5]We estimate the work-units for a multigrid cycle based on using as many levels as possible [7]. Non-Galerkin coarse-level operators have some advantage in terms of WU over Galerkin operators because of their sparser stencils, especially for the simple example of the standard discrete Laplacians.

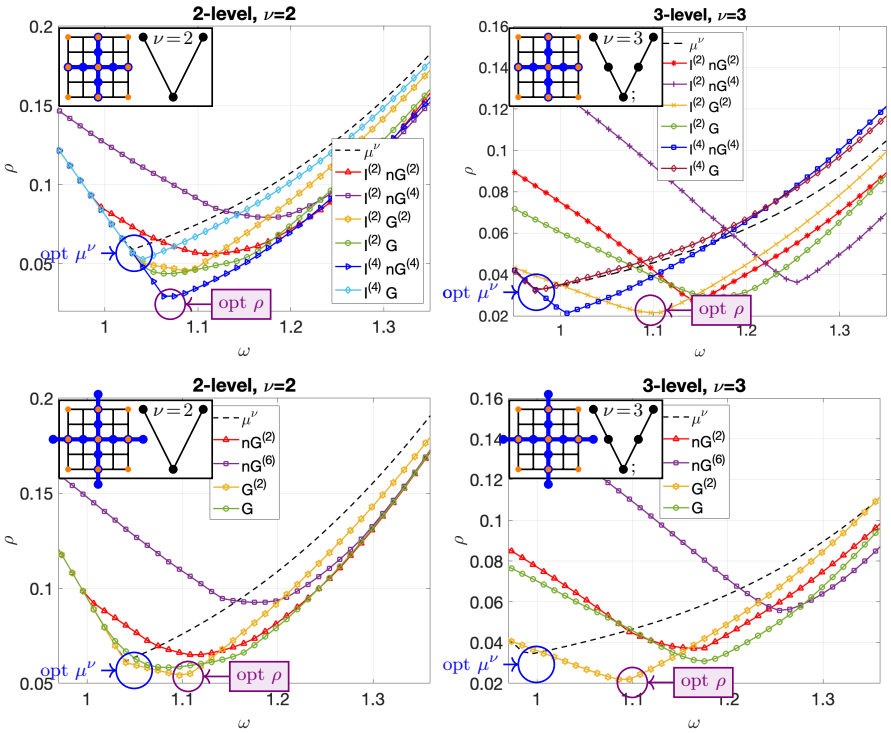

Figure 2 ACR $= \rho$ versus ω (LFA results). **2**D, 2-level and 3-level V cycles with red-black Gauss-Seidel smoothing (ν sweeps per cycle), with transfer and coarse-level operators as given in Table 1. Top: order **4**; bottom: order **6**. Left: 2-level V[1, 1] cycles; right: 3-level V[2, 1] cycles.

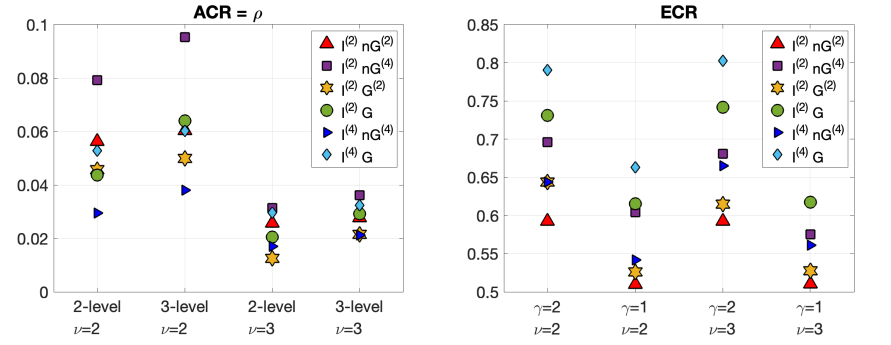

Figure 3 Optimal ACR $= \rho$ and ECR (LFA results). **2**D, order **4**, V ($\gamma = 1$) and W ($\gamma = 2$) cycles with red-black Gauss-Seidel smoothing (ν sweeps per cycle); with transfer and coarse-level operators as given in Table 1, at their respective optimal ω. The 3-level ACR is used to estimate the ECR of the corresponding multi-level V cycle ($\gamma = 1$), while the 2-level ACR is used to estimate the ECR of the corresponding multi-level W cycle ($\gamma = 2$).

provide estimates of the ECRs (here ECR is normalized from the ACR) for typical multi-level V ($\gamma = 1$) and W ($\gamma = 2$) cycles. Again, the three-level LFA estimated ACRs are used for corresponding multi-level V cycles, while the two-level ACRs are used for corresponding multi-level W cycles. The algorithms using second-order accurate coarse-level correction operators are found to be always as good as, or better, than those using higher-order accurate ones. These results provide some good justification for the use of second-order accurate coarse-level correction operators for fourth-order accurate fine-level discretizations, especially given the added benefits for grid generation associated with using lower-order accurate approximations on coarse levels.

We wish to see if second-order accurate coarse-level operators can also be used even for sixth-order accurate discretizations without any significant effect on the convergence rates. We consider the standard sixth-order accurate finite-difference stencil for the two-dimensional (negative) Laplacian

$$L_h^{(6)} = \frac{1}{180h^2} \begin{bmatrix} & & -2 & & \\ & & 27 & & \\ & & -270 & & \\ -2 & 27 & -270 & 980 & -270 \quad 27 \quad -2 \\ & & -270 & & \\ & & 27 & & \\ & & -2 & & \end{bmatrix}_h . \tag{23}$$

We use the second-order accurate transfer operators (21). On the coarse levels, we consider operators constructed from the second-order accurate $L_h^{(2)}$, in comparison with coarse-level operators constructed from the actual fine-level operator (23) (see Table 1). ACRs are given in Figure 2 (bottom) for the two-level V[1, 1] and three-level V[2, 1] cycles, which are similar to the fourth-order results. Overall the best convergence rates are obtained using the second-order accurate Galerkin coarse-level operators. We conjecture that the same conclusion will hold for eighth- and higher-order accurate discretizations.

4 Compatibility boundary conditions and sample overset grid results

In general, we aim to achieve fast multigrid convergence on overset grids, similar to LFA results on infinite Cartesian grids. An important consideration for a multigrid solver on general domains is the discretization of boundary conditions; this becomes even more prominent for *higher-order accurate* schemes. The goal is to minimize the effect of the domain boundary, so as to maintain multigrid convergence rates comparable to those on an infinite or periodic domain. If we can discretize the physical boundary conditions in such a way that the resulting system enjoys multigrid convergence properties comparable to LFA, then the conclusions we obtained based on LFA in Section 3, such as

lower-order accuracy on coarse levels, as well as optimal values of the smoothing parameter ω, can hopefully be then applied in our overset-grid multigrid solver to obtain similar convergence rates.

In this section, we argue that LFA multigrid convergence rates can be matched on general geometry when the boundary conditions are discretized numerically using *compatibility boundary conditions*. We first study, in Section 4.1, the effect of numerical boundary conditions on multigrid convergence via a model problem. Then, in Section 4.2, we present some representative convergence results on a sample overset grid, using compatibility boundary conditions, as well as incorporating some LFA-based conclusions from Section 3. Performance of the multigrid solver on overset grids is compared to an AMG solver and two selected Krylov-space solvers in Section 4.3.

4.1 Compatibility boundary conditions for a model problem

To illustrate how the discretization of boundary conditions affects multigrid convergence, let us consider a simple one-dimensional example, the following boundary value problem on the domain $[0,\ 1]$,

$$\begin{cases} -\partial_x^2 u = f, & x \in (0,\ 1); \\ u(0) = g_0, \quad \partial_x u(1) = g_1. \end{cases} \tag{24}$$

(Here the PDE reduces to just an ordinary differential equation, but to be consistent we still keep the notation $\partial_x^k = \frac{d^k}{dx^k}$ for the kth derivative.) The corresponding eigenvalue problem is

$$\begin{cases} -\partial_x^2 \phi = \lambda \phi, & x \in (0,\ 1); \\ \phi(0) = 0, \quad \partial_x \phi(1) = 0, \end{cases} \tag{25}$$

yielding the eigenfunctions

$$\phi(x) = \sin\left[(n - \frac{1}{2})\pi x\right], \quad n \in \mathbb{Z}, \tag{26}$$

which are Fourier modes with a certain phase and periodicity. Consider a fourth-order accurate discretization of the differential equation on the grid $x_j = jh,\ j = 0,\ 1,\ \cdots,\ N-1,\ N;\ h = 1/N$,

$$-D_{xx}^{(4)}\, U_j = f(x_j), \quad j = 1,\ \cdots,\ N-1, \tag{27}$$

where

$$D_{xx}^{(4)} \stackrel{\text{def}}{=} D_+ D_- (1 - \frac{h^2}{12} D_+ D_-) \tag{28}$$

is the standard fourth-order accurate central-difference approximation to the second derivative, in which D_+ and D_- are the standard forward and backward divided-difference operators,

$$D_+U_j \stackrel{\text{def}}{=} \frac{U_{j+1} - U_j}{h}, \qquad D_-U_j \stackrel{\text{def}}{=} \frac{U_j - U_{j-1}}{h}. \tag{29}$$

If the problem is defined on the whole infinite domain (or a periodic domain), the eigenfunctions of (27) will be the discrete Fourier modes $\phi_h(x,\ \theta)$ (8) ($d = 1$) used in LFA. Thus when solving the system with multigrid, the convergence rate can be estimated by examining how the multigrid iteration operates on each of the Fourier modes.

Compatibility Boundary Conditions (CBCs), as the name suggests, are designed so as to be compatible with both the PDE and the physical boundary conditions. The use of CBCs leads to centered numerical boundary conditions, and avoids one-sided approximations which effectively include artificial *extrapolation* conditions. In simple cases, such as with homogeneous Dirichlet or Neumann boundary conditions on a square, CBCs lead to the well-known odd or even symmetry conditions [29]. As a result the discrete eigenfunctions have the same form as the continuous counterparts, and thus the LFA results readily apply.

The CBCs are formally derived by assuming the solution is sufficiently smooth so that the PDE, and its derivatives, hold on the boundary. In particular, consider the model BVP (24). For a Dirichlet boundary condition, the first two CBCs based on the PDE are

$$\partial_x^2 u = -f, \tag{30a}$$
$$\partial_x^4 u = -\partial_x^2 f. \tag{30b}$$

For a Neumann boundary condition, the first CBC is given by applying the PDE on the boundary, while the second CBC is formed by applying the boundary operator (∂_x) to the PDE,

$$\partial_x^2 u = -f, \tag{31a}$$
$$\partial_x^3 u = -\partial_x f. \tag{31b}$$

For convenience we introduce ghost points beyond the boundaries: $x_j = jh,\ j = -2,\ -1,\ N+1,\ N+2$. The Dirichlet boundary condition and CBCs (30) are discretized as

$$U_0 = g_0, \tag{32a}$$
$$D_{xx}^{(4)}\ U_0 = -f(0), \tag{32b}$$
$$(D_+D_-)^2 U_0 = -\partial_x^2 f(0), \tag{32c}$$

while the Neumann boundary condition and CBCs (31) lead to

$$D_0(1-\frac{h^2}{6}D_+D_-)\,U_N = g_1, \tag{33a}$$

$$D_{xx}^{(4)}\,U_N = -f(1), \tag{33b}$$

$$D_0D_+D_-U_N = -\partial_x f(1). \tag{33c}$$

Here $D_0 = \frac{1}{2}(D_+ + D_-)$ is second-order accurate centered finite-difference approximation to ∂_x. Note that even though some compatibility conditions are only approximated to second-order accuracy the overall scheme is still fourth-order accurate [30]. We call the resulting numerical BCs, such as (32) and (33), *compatibility numerical BCs*, or *numerical CBCs*. It is straightforward to show that the eigenfunctions of the resulting system will be

$$\phi_h(x_j) = \sin\left[(n-\frac{1}{2})\pi x_j\right], \quad n = 1,\ \cdots,\ N+1. \tag{34}$$

Note that $\phi_h(x_j) = \phi(x_j)$. Since the eigenfunctions of the discrete system are just the Fourier modes (with certain periodicity), the multigrid algorithm is solving a similar problem to that studied in LFA on an infinite or periodic domain, its convergence rate can be expected to be similar as well.

In comparison, consider using the fourth-order accurate one-sided difference approximation and (fifth-order) extrapolation conditions at the boundaries (note that combining a centered approximation to the Neumann boundary condition with extrapolation conditions for the ghost points would lead to an one-sided approximation)

$$\begin{cases} U_0 = g_0, \\ U_{-1} = 5U_0 - 10U_1 + 10U_2 - 5U_3 + U_4; \\ \frac{1}{12h}(25U_N - 48U_{N-1} + 36U_{N-2} - 16U_{N-3} + 3U_{N-4}) = g_1, \\ U_{N+1} = 5U_N - 10U_{N-1} + 10U_{N-2} - 5U_{N-3} + U_{N-4}. \end{cases} \tag{35}$$

One can find the corresponding eigenvalues and eigenfunctions numerically. The eigenvalues are not very different from those corresponding to the system with CBCs (32) and (33); the eigenfunctions, however, will not be as regular as ϕ_h (34). While the low-frequency ones will remain close to the continuous Fourier modes, the numerical boundary conditions will introduce high-frequency eigenfunctions that deviate significantly from (i.e. cannot be seem as perturbations from) the continuous counterparts; thus raising challenges to a multigrid solver designed using principles from local Fourier analysis. Since the eigenspaces of a multigrid iteration consists of these different eigenfunctions, the convergence property corresponding to the system with (35) will deviate from LFA. More specifically, the eigenfunctions will no longer enjoy the aliasing property of the Fourier modes, and each of these different high-frequency eigenfunctions (modes), once represented on the coarse

grid, will generally have to be a combination of all of the low modes. This will result in a coarse-level correction operator that no longer enjoys a simple block structure in the eigen-basis, and is therefore likely to be less effective in reducing the low-frequency modes. The multigrid convergence rate, in turn, will very likely be hindered.

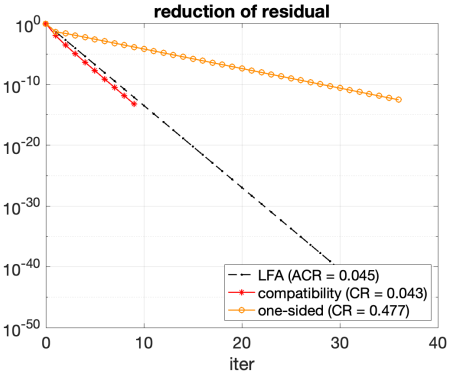

Figure 4 Comparison of multigrid convergence with compatibility and one-sided numerical boundary conditions, 1D, 4th-order accurate (red-black smoothing with $\omega = 1$, V[1, 1] cycle, 4th-order non-Galerkin coarse-level operator). The LFA estimated asymptotic convergence rate (ACR $= \rho$) is plotted in dashed black line as reference.

In Figure 4, we present results from a numerical test applying a multigrid solver for the model problem (24) using the scheme (27), paired with the numerical boundary conditions (32) and (33), versus (35). As we can see, the multigrid convergence rate with compatibility boundary conditions is very close to the asymptotic convergence rate estimated from local Fourier analysis, while with the one-sided boundary conditions the convergence is significantly hindered. We also observe from the numerical test that in the latter case the error (residual) near the boundary becomes dominant.

4.2 Sample overset-grid results

In this section, we present some evidence that good multigrid convergence rates can be obtained on complex geometry using overset grids, based on the knowledge gained from our local Fourier analysis of the model problem (such as in Section 3). In order to achieve similar convergence rates on an overset grid to those estimated from LFA, we employ many strategies in our multigrid solver. Two main issues that must be addressed are the numerical treatment of the *physical boundary conditions* on general curved boundaries, and the treatment of the residual near *interpolation boundaries* between the component grids. Our guiding principle is to attempt to retain smooth behavior of the residual (error) near physical or (artificial) interpolation boundaries so that the residual can be well-represented on the next coarser level. In this case, one might

expect that the residual (error) will be reduced during the multigrid iteration as if the computation were being performed on a single, infinite domain. In Section 4.1 we have studied the treatment of physical boundary conditions using compatibility boundary conditions for a model problem, and showed that good multigrid convergence can be obtained with little effect from the boundary. On an overset grid, CBCs are applied on the physical boundaries of all component grids. These conditions are discretized in the parameter (computational) space (see [31] for details). The effect of interpolation boundaries, and its treatment with a composite smoother, are studied in other works [1]. A detailed discussion of our multigrid algorithm on overset grids, using nonstandard coarse-level operators and optimized cycles with a composite smoother, is beyond the scope of the current paper. Instead, we present some representative results. The overset grids are generated using *Ogen*, and the multigrid results are generated using the *Ogmg* solver, both available in the *Overture*[6] framework. Figure 5 shows an overset grid for three shapes embedded in a rectangle. This *shapes* grid illustrates a typical configuration where most of the domain is covered by one or more Cartesian background grids, while boundary conforming grids are confined to narrow regions around the curved boundaries. As the mesh is refined, a larger and larger percentage of the grid points would lie in the Cartesian grids. Our multigrid algorithm on overset grids leads to very efficient overall computations; in particular, special optimizations on the smoother for Cartesian grids greatly increases the overall efficiency.

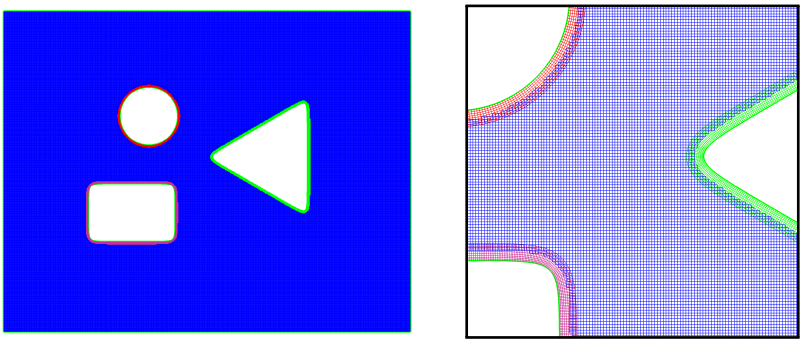

Figure 5 An overset grid for some shapes. Left: fine grid. Right: magnified view showing the boundary fitted curvilinear grids around each shape. A large majority of the grid points lie in the blue background grid.

Figure 6 compares the multigrid convergence when solving Poisson's equation with Dirichlet boundary conditions on an uniform square reference grid as well as on the *shapes* grid shown in Figure 5. For each of these two grids there are four convergence results shown, comparing simulations to fourth-order accuracy and those to second-order accuracy, using Galerkin and

[6] Available at: http://www.overtureframework.org

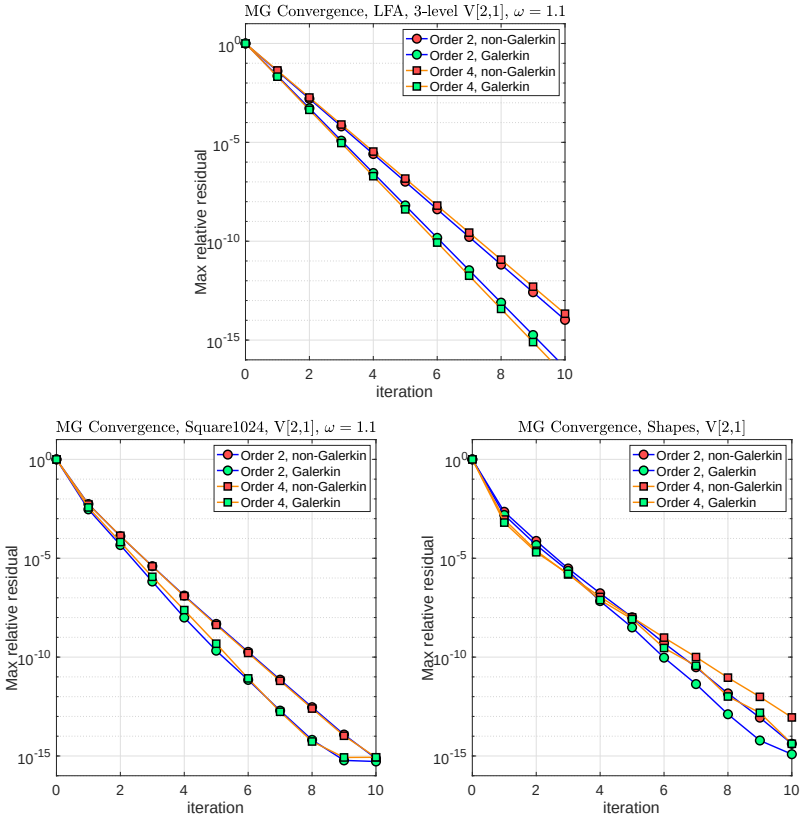

Figure 6 Multigrid convergence. Top: LFA estimated convergence (3-level). Bottom left: results for a square with 1024^2 grid points (10 levels); bottom right: corresponding results for the *shapes* overset grid (4 levels). Results are shown for second-order accuracy (circles), fourth-order accuracy (squares), with Galerkin (green markers) and non-Galerkin (red markers) coarse-level operators. Using compatibility boundary conditions, the convergence rates for the overset grid are comparable to those for a square.

non-Galerkin coarse-level operators. For the fourth-order accurate computations, in particular, *second*-order accurate operators are used on coarse levels, as discussed in Section 3. *Compatibility* boundary conditions are employed to discretize the physical boundary conditions to the appropriate order-of-accuracy. The first thing to observe is that the multigrid convergence rates for fourth-order accurate schemes are quite comparable with those for second-order accuracy. The square grid uses a V[2, 1] cycle with a red-black smoother using $\omega = 1.1$, which is the optimized parameter value for the fourth-order accurate scheme with second-order Galerkin coarse-level operators on Cartesian grids, based on the three-level LFA results in Section 3 ($\omega = 1.1$ is also a good parameter value for the other three multi-level cycles). The *shapes* overset grid uses an adaptive cycle [1] together with additional smoothing in the vicinity of interpolation boundaries. In particular, the background grid of the *shapes* grid uses the same V[2, 1] cycle and smoother (with $\omega = 1.1$) as the square

grid. Figure 6 (bottom left) shows convergence results for a square grid, while Figure 6 (bottom right) shows results for the *shapes* grid on the same vertical scale. For reference, we also present the LFA estimated convergence history (based on the corresponding three-level V cycles) in Figure 6 (top). The convergence on the square grid, as expected, agrees with the LFA estimates very well. The convergence on the *shapes* overset grid also compares favorably with the LFA estimates. It is seen that the convergence history for the overset grid is similar to that for the simple square, both showing very rapid convergence. (There are some differences between the two cases but these are fairly minor.) We can conclude from these results that it is possible to obtain *textbook*-like convergence results for overset grids even at fourth-order accuracy, when using second-order accurate coarse-level operators.

4.3 Comparison of *Ogmg* to AMG and Krylov solvers

In this section, we present some computational performance data to compare our multigrid solver for overset grids with some standard iterative solvers. The CPU setup and solve times, and memory usage, are presented for the overset-grid multigrid solver *Ogmg*, an algebraic multigrid (AMG) solver, and two Krylov solvers. Results are presented for the uniform square grid and *shapes* grid described in Section 4.2 when solving the model problem of Poisson's equation with Dirichlet boundary conditions, to second- and fourth-order accuracy.

					CPU time (s)			storage
Grid	order	Solver	its	$\|res\|_\infty$	total	setup	solve	reals/pt
Square	2	Ogmg	9	9.4e-10	0.28	0.01	0.27	8.7
Square	2	AMG	12	1.9e-07	6.26	3.07	3.19	99.8
Square	2	BiCGSt	371	8.6e-08	23.27	0.97	22.30	64.3
Square	2	GMRES	1307	8.8e-08	68.64	1.14	67.50	86.4
Square	4	Ogmg	9	1.7e-08	0.49	0.01	0.48	8.4
Square	4	AMG	24	3.6e-08	15.43	4.77	10.66	178.0
Square	4	BiCGSt	277	1.9e-08	27.42	1.79	25.63	113.1
Square	4	GMRES	657	9.6e-09	61.82	2.85	58.97	155.8
Shapes	2	Ogmg	9	3.5e-09	0.76	0.14	0.62	14.9
Shapes	2	AMG	35	6.6e-10	8.22	1.95	6.27	112.6
Shapes	2	BiCGSt	325	8.0e-09	11.67	0.58	11.10	61.9
Shapes	2	GMRES	710	3.3e-09	22.20	0.64	21.56	83.6
Shapes	4	Ogmg	10	6.3e-09	1.51	0.24	1.27	20.3
Shapes	4	AMG	241	1.9e-08	69.68	3.29	66.39	168.1
Shapes	4	BiCGSt	188	1.3e-08	11.74	1.05	10.69	107.9
Shapes	4	GMRES	254	2.1e-08	16.17	1.89	14.28	149.0

Table 2 Performance comparison of *Ogmg* to AMG (HYPRE/boomerAMG) and Krylov (Bi-CG-Stab+ILU(1), GMRES+ILU(3)) solvers.

Table 2 shows the timing and memory usage for the four solvers[7] considered:

- Ogmg: The *Ogmg* solver using V[2, 1] cycles.
- AMG: The BoomerAMG solver from HYPRE (version 2.25.0).

[7]The results were computed using a Intel Xeon 3.0 GHz processor.

- BiCGSt: The bi-CG-Stab Krylov solver with an ILU(1) preconditioner from PETSc (version 3.18.2).
- GMRES: The GMRES Krylov solver with an ILU(3) preconditioner from PETSc (version 3.18.2).

We note that bi-CG-Stab is generally our preferred Krylov-based solver. Based on the data in Table 2, Figure 7 shows the CPU times to solve (no setup) relative to the time for *Ogmg* (top), and the memory usage, in double-precision floating-point numbers per grid-point (reals/grid-pt) (bottom), for the four solvers.

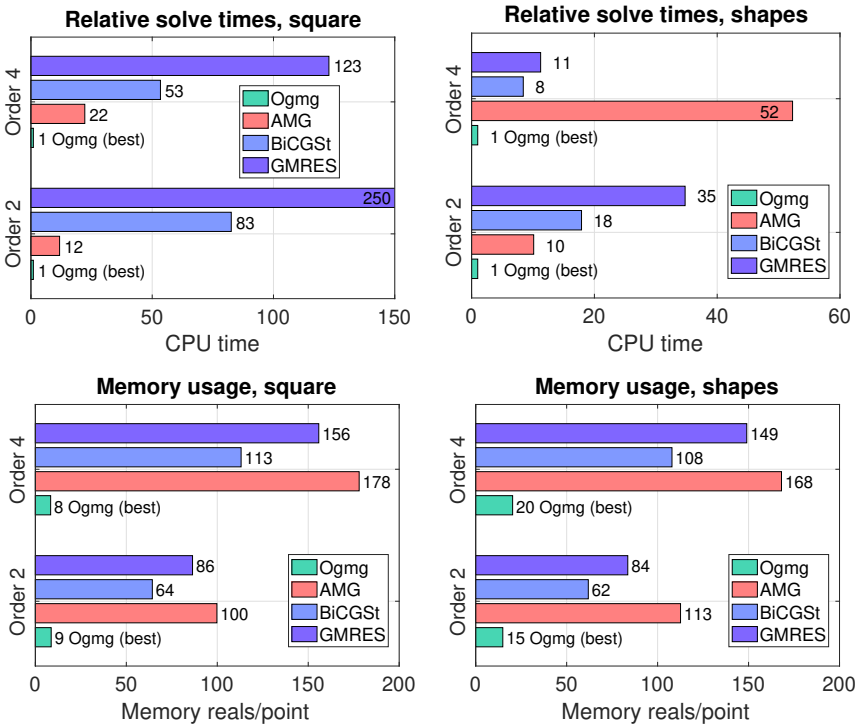

Figure 7 Performance comparison of *Ogmg* solver to other iterative solvers, using data from Table 2. Top: CPU times to solve (no setup) relative to the time for *Ogmg*; bottom: memory usage in reals/grid-point. Left: square grid with 1024^2 grid points; right: *shapes* overset grid (Figure 5).

The results show that the *Ogmg* solver is many times faster than the other solvers. For example, on the square grid at fourth-order accuracy, *Ogmg* is about 22 times faster than AMG, and 53 times faster than bi-CG-Stab, not counting time for setup. *Ogmg* has both smaller setup times and smaller solve times. Low setup times are particularly important for problems with moving or deforming geometry where the grid changes every time-step. *Ogmg* also uses significantly less memory compared to the other solvers. For example,

on the *shapes* grid at fourth-order accuracy, *Ogmg* uses about 20 reals/grid-pt, while AMG uses about 168 reals/grid-point, and GMRES with an ILU(3) preconditioner uses about 149 reals/grid-pt.

It should not be surprising that *Ogmg* outperforms more general-purpose solvers such as AMG and Krylov-based methods. Most of the CPU time for *Ogmg* is spent in the smoothers which can be highly optimized for both structured curvilinear grids and especially for Cartesian grids. *Ogmg* is also a matrix-free solver so that the matrix coefficients on a Cartesian grid do not need to be stored. On a curvilinear grid, the metric derivatives, $\partial r_i/\partial x_j$ (where $\mathbf{r}$ is in parameter space with change-of-variables from the physical space $\mathbf{x}$ [1]), are stored instead of the matrix coefficients, and this is a benefit for higher-order accurate approximations with wider stencils. We note that AMG performs rather poorly on the *shapes* grid at fourth-order accuracy. It is likely that the AMG algorithm can be adjusted, or a specialized preconditioner could be designed, to improve the performance, but we have not attempted that here.

5 Coarsening by a general factor r

As discussed in the introduction, the generation of coarse-level overset grids is a major challenge in multigrid solvers on overset grids [1]. Unlike for a single Cartesian grid, where one can simply force the number of grid points in each space direction to support a certain number of multigrid levels (for example, with sufficient factors of two), such strict constraints for each component grid will strongly restrict the construction of quality overset grids. Thus for flexibility in multi-level overset-grid generation, we wish to relax this kind of restriction, and allow coarsening by a general factor r. In this section, we study factor-r coarsening on the model problem (see Figure 8, right, for an illustration). Multigrid is a balance between smoothing, which reduces high-frequency components of the error, and the coarse-level correction, which reduces low-frequency components. The coarsening factor r affects this balance since it changes the relative decomposition of the error into its high and low-frequency components. In particular, we wish to determine whether using a coarsening factor r in the neighborhood of two achieves convergence rates similar to that with standard coarsening. In addition, it is also of interest to investigate, more generally, what value of r leads to the best multigrid convergence in any dimension.

Consider the model problem (3a) in d space dimensions with the standard second-order accurate finite-difference stencil, on a fine grid $\mathcal{G}_h$ as given in (16) with size N. Suppose we have a target grid-coarsening factor, denoted by r_{target} (> 1). To construct a coarse grid with spacing $H = rh$, where the actual coarsening factor $r \approx r_{\text{target}}$, set $M \stackrel{\text{def}}{=} \lfloor N/r_{\text{target}} \rfloor$[8], $H = 1/M$, $r \stackrel{\text{def}}{=} H/h = N/M$. Given the fine grid $\mathcal{G}_h$ and coarse grid $\mathcal{G}_H = \{\mathbf{k}H : \mathbf{k} \in \{0, 1, \cdots, M\}^d\}$, we construct the transfer operator I_H^h using linear interpolation, and define the restriction operator as the adjoint $I_h^H = (I_H^h)^* = (I_H^h)^T/r^d$.

[8]The floor function $\lfloor a \rfloor$ denotes the largest integer less than or equal to $a \in \mathbb{R}$.

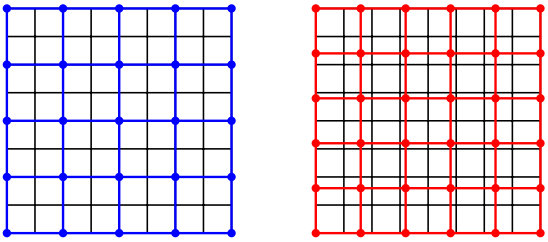

Figure 8 Some coarsening strategies in two dimensions. Left: standard coarsening, $H = 2h$. Right: coarsening by a general factor r, $H = rh$.

The non-Galerkin coarse-level operator L_H uses the same stencil as L_h, while the Galerkin $L_H = I_h^H L_h I_H^h$ can be derived from the transfer operators. Based on the two-level construction, it is straightforward to construct a general multi-level algorithm with a target coarsening factor r_{target}. Let r_l denote the actual coarsening factor from level l to level $l+1$. If there are $N_0 \equiv N = 1/h$ grid points in each direction on the finest level, then the number of grid points in each direction at level $l+1$ is $N_{l+1} = \lfloor N_l / r_{\text{target}} \rfloor$, and the corresponding coarsening factor $r_l = N_l/N_{l+1} = h_{l+1}/h_l$, for $l = 0,\ 1,\ \cdots,\ l_{\max} - 1$.

For a fair comparison in terms of the optimal r_{target}, we choose the number of coarse levels as

$$l_{\max}(r_{\text{target}}) \approx \log_{r_{\text{target}}} \frac{N_0}{N_{\min}} \propto \frac{1}{\log r_{\text{target}}}, \tag{36}$$

so that the coarsest level has approximately a fixed number of grid points $N_{\min}$. In one dimension it is not surprising that the optimal choice would be $r_{\text{target}} = 2$; in particular for the model problem at $r = 2$ the multigrid algorithm becomes a direct solver. The question of optimal r_{target} is more interesting in two or more space dimensions. The multigrid algorithm involves a trade-off between reducing the convergence rate, and increasing the computational cost (measured by WU). As r approaches one, the coarse-level solve becomes more close to exact (reducing the CR), but also more expensive (increasing the WU), and thus for a fair comparison we choose the ECR as a good metric for choosing the optimal r. Figure 9 shows convergence rates from multi-level V$[1,1]$ cycles with factor-r coarsening in two dimensions. (Results are shown for the non-Galerkin coarse-level operators, that is, the same stencil is used on every level; graphs for the Galerkin coarse-level operators are expected to be similar in form.) The multi-level CRs, which roughly mimic the two-level results, increase approximately monotonically as r increases. The convergence rate goes to zero at $r = 1$ since the coarse grids all approach the fine grid and there is an exact solve on the coarsest level. The ECR shows a fairly broad minimum around $r_{\text{target}} = 2$ with a dip at exactly 2, largely due to the work-units being discontinuously small at this value (see the discussion in Section 5.1). We can conclude that there is a range of coarsening factors around $r = 2$ where the multigrid algorithm has fast convergence, which is the

behavior we desire; and there is no obvious optimal value of the coarsening factor based on the ECR, except for the special behavior at $r = 2$.

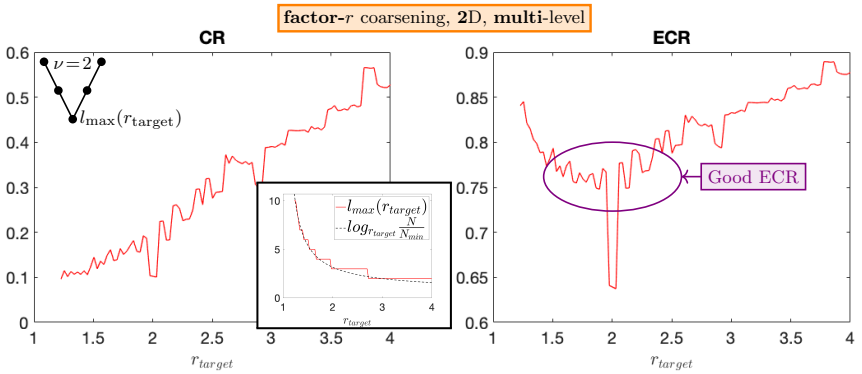

Figure 9 CR and ECR versus r_{target}. **2**D, **multi**-level $\mathbf{V}[1,1]$ cycle with red-black smoothing ($\omega = 1$) and **factor-**r coarsening $r_l \approx r_{\text{target}}$ (and non-Galerkin coarse-level operators). ($N = 64$, $N_{\min} = 8$.) A relatively good ECR can be obtained for a wide range of $r_{\text{target}} \in (1.5, 2.5)$.

5.1 Local Fourier smoothing analysis of factor-r coarsening with a Jacobi smoother

In this section, we use local Fourier analysis to better understand the computational results for factor-r coarsening given above. To simplify the analysis we consider a multigrid cycle that uses a Jacobi smoother with a relaxation parameter ω, combined with an ideal coarse-level correction. We choose to analyze the ω-Jacobi smoother instead of the red-black smoother, because the former is easier to analyze since the Fourier modes are its eigenfunctions, and because with Jacobi the smoothing rate generally gives a better estimate of the overall multigrid convergence rate (compared to the corresponding estimates with a red-black smoother). In addition, it is straightforward to choose an optimal relaxation parameter $\omega^*(r)$ for the Jacobi smoothing rate, and thus we can have a fairer comparison of the convergence rates at different r. At the same time, we expect that the general conclusions drawn from using the Jacobi smoother for studying factor-r coarsening will also hold for the red-black smoother.

We study the model problem of Poisson's equation on the infinite grid $\mathcal{G}_h$ (1). Let $L_h = -\Delta_h$ be the standard second-order accurate approximation to the negative Laplacian in d dimensions. The Jacobi smoothing operator, with a relaxation parameter ω, is given by

$$S_h = I_h - \frac{\omega h^2}{2d} L_h. \tag{37}$$

The Fourier symbol of S_h, corresponding to $\phi_h(\cdot,\ \boldsymbol{\theta}),\ \boldsymbol{\theta} = [\theta_1,\ \cdots,\ \theta_d]^T \in \Theta$, is

$$\hat{S}_h(\boldsymbol{\theta}) = 1 - 2\omega\xi_{\boldsymbol{\theta}}, \quad \xi_{\boldsymbol{\theta}} \stackrel{\text{def}}{=} \frac{1}{d}\sum_{k=1}^{d} \sin^2 \frac{\theta_k}{2}. \tag{38}$$

That is, $S_h\,\phi_h(\cdot,\ \boldsymbol{\theta}) = \hat{S}_h(\boldsymbol{\theta})\,\phi_h(\cdot,\ \boldsymbol{\theta})$. Consider a coarse grid $\mathcal{G}_{rh} = \{\mathbf{x} = \mathbf{j}\,rh :\ \mathbf{j} \in \mathbb{Z}^d\}$ on the first coarse level. We will estimate the multigrid convergence rate with factor-r coarsening based on the smoothing factor μ. For an $(h,\ rh)$ two-level cycle, the sets of low and high frequencies are given by

$$\Theta_r^{\text{low}} \stackrel{\text{def}}{=} \big[-\frac{\pi}{r},\ \frac{\pi}{r}\big)^d, \quad \Theta_r^{\text{high}} = \Theta \setminus \Theta_r^{\text{low}}. \tag{39}$$

The smoothing factor of ω-Jacobi is thus determined from the worst-case convergence rate over the high frequencies:

$$\mu(r,\ \omega) \stackrel{\text{def}}{=} \sup_{\boldsymbol{\theta}\in\Theta_r^{\text{high}}} |\hat{S}_h(\boldsymbol{\theta})| = \sup_{\xi\in[\zeta,\ 1]} |1 - 2\omega\xi| = \max\big\{\,|1-2\omega\zeta|,\ |1-2\omega|\,\big\}, \tag{40}$$

where

$$\zeta(r) \stackrel{\text{def}}{=} \frac{1}{d}\sin^2\frac{\pi}{2r}. \tag{41}$$

An optimal relaxation parameter ω^* can be chosen to minimize the smoothing factor:

$$\mu^*(r) \stackrel{\text{def}}{=} \mu(r,\ \omega^*(r)) = \frac{1-\zeta(r)}{1+\zeta(r)}, \quad \omega^*(r) = \frac{1}{1+\zeta(r)}. \tag{42}$$

Figure 10 (left) plots μ^* in different dimensions d, and it can be seen that $\mu^*(r)$ is a monotone increasing function for $r > 1$. For a multigrid cycle with ν smoothing steps, the convergence rate ρ might be expected to be approximately $(\mu^*)^\nu$.

We focus on the two-dimensional case ($d = 2$), and to compare to the theoretical estimates we compute the actual convergence rates of a multigrid algorithm with Jacobi smoothing and ω^* as given in (42). Computed CRs using V[1, 1] and W[1, 1] multi-level cycles are compared to the theoretical curve $(\mu^*)^\nu$ in Figure 10 (right). As might be expected, the W cycle results compare quite well with the theoretical estimates except for some deviation as r tends to one. The V cycle results compare well with the theoretical estimates for $r > 2$ but there is a more pronounced deviation for $r < 2$ where the computed convergence rate levels off to a value of CR ≈ 0.35. This deviation from the theoretical estimates comes from the distinction between the actual and ideal coarse-level corrections which is more prominent as r approaches one, and thus can not be explained by smoothing analysis alone.

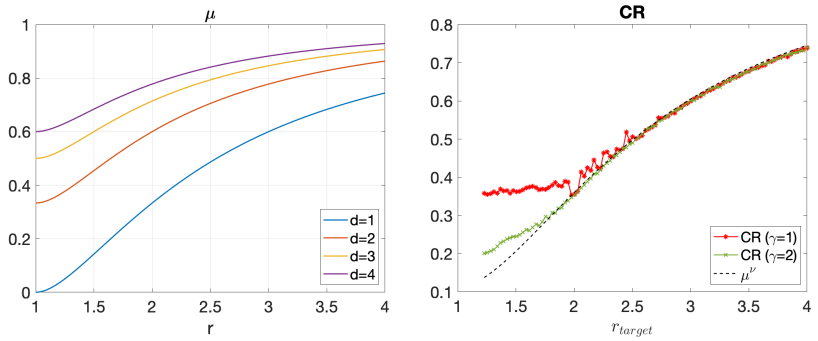

Figure 10 Left: Smoothing rate μ^* versus r in d dimensions. Right: CR versus r_{target}, in comparison with the estimate μ^ν in **2**D ($d = 2$), V ($\gamma = 1$) and W ($\gamma = 2$) cycles with ω^*-Jacobi smoothing ($\nu = 2$) and **factor-**r coarsening (and non-Galerkin coarse-level operators; $N = 64,\ N_{\min} = 8$).

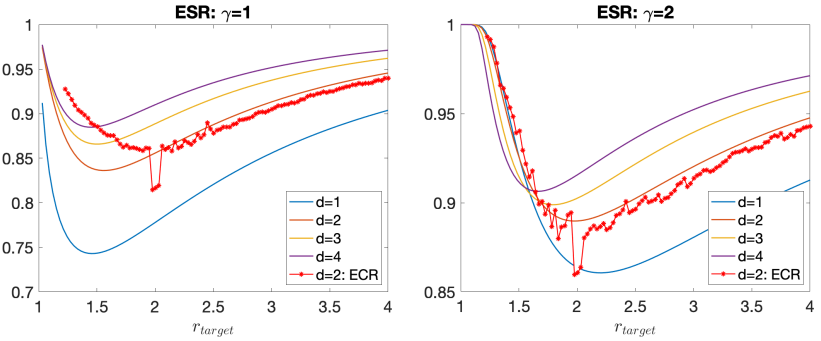

Figure 11 ESR (estimate of the ECR) versus r. V ($\gamma = 1$, left) and W ($\gamma = 2$, right) multi-level cycles with ω^*-Jacobi smoothing ($\nu = 2$) and **factor-**r coarsening; in **2**D ($d = 2$) in comparison with computed ECR (with non-Galerkin coarse-level operators; $N = 64,\ N_{\min} = 8$).

To get an idea of which coarsening factor r_{target} gives the fastest algorithm we compare the effective convergence rates. To do this, we estimate the work-units of a general multi-level γ-cycle with factor-r coarsening by

$$\text{WU}[r;\ \gamma,\ \nu] \approx (\nu + 3) \sum_{l=0}^{l_{\max}-1} (\frac{\gamma}{r^d})^{l-1}, \tag{43}$$

where $l_{\max}$ is chosen as before in (36). Note the change in behavior in the WU depending on whether $\gamma < r^d$ or $\gamma \geq r^d$. However, at exactly $r = 2$ (or factors of 2) the WU is somewhat reduced due to the alignment of the coarse grid points with half the fine grid points. Approximating the convergence rate by $\rho \approx (\mu^*)^\nu$, we define the *effective smoothing rate*,

$$\text{ESR} \stackrel{\text{def}}{=} (\mu^*)^{\frac{\nu}{\text{WU}}}, \tag{44}$$

as an estimated ECR for a multigrid cycle with ν smoothing steps. This estimated ECR is graphed in Figure 11 for V ($\gamma = 1$) and W ($\gamma = 2$) cycles with $\nu = 2$. Thus we can estimate the optimal coarsening factor from the ESR: for a V cycle, the optimal coarsening factor has a value $r < 2$, in particular for $d = 2$ the minimizer $r \approx 1.55$; for a W cycle the optimal coarsening factor $r \approx 1.97$ is close to 2 in two dimensions while it has a value $r \approx 1.79$ in three dimensions.

In two dimensions ($d = 2$) in particular, we also compare the estimate ESRs (44) with ECRs from actual computations for multigrid cycles with ω^*-Jacobi smoothing (the corresponding CRs have been presented in Figure 10, right), as shown in Figure 11. We can see that the ESR gives a fairly good estimate for the ECR for a W cycle. The comparison is not as good for the V cycle when $r_{\text{target}} < 2$, but this is to be expected due to the deviation in the CRs from the theory as already shown in Figure 10 (right). Also, note that the discontinuously good ECR at $r_{\text{target}} = 2$ comes from the fact that the WU is discontinuously low when the coarse grid is embedded in the fine grid.

Finally, let us compare the CRs in Figure 10 (right) and the ECRs in Figure 11 (left) with Jacobi smoothing ($\omega = \omega^*$), to the results in Figure 9 with red-black smoothing ($\omega = 1$), in two dimensions. We can see that the red-black smoother, as usual, gives faster convergence than the Jacobi smoother, but the CRs and ECRs seem to vary with r_{target} in a very similar manner. Thus local Fourier smoothing analysis with the ω-Jacobi smoother appears to reasonably describe the basic features of multigrid with general factor-r coarsening.

6 Conclusions

In this paper, we considered some nonstandard coarsening strategies for geometric multigrid by studying the solution to the model problem of Poisson's equation. We showed that second-order accurate coarse-level operators could be effectively used with fourth- and sixth-order accurate fine-level operators, with generally no degradation in effective convergence rates. We described how compatibility boundary conditions can be used to develop centered numerical boundary conditions that generally have better properties for multigrid solvers than one-sided approximations. We showed multigrid results for a sample overset grid on a complex geometry that had comparable convergence rates to those for a reference square domain, thus providing some justification for our application of local Fourier analysis results to multigrid solvers on overset grids. Performance data for the sample overset grid showed that the Ogmg solver was many times faster and used significantly less memory than an AMG solver and two Krylov solvers. We also considered coarsening by a general factor r, and showed that good effective convergence rates are obtainable for a range of coarsening factors around two. The algorithm was studied using both local Fourier analysis and numerical simulations.

References

[1] Henshaw, W.D.: On multigrid for overlapping grids. SIAM J. Sci. Comput. **26**(5), 1547–157228 (2005). https://doi.org/10.1137/040603735

[2] Henshaw, W.D.: A fourth-order accurate method for the incompressible Navier-Stokes equations on overlapping grids. J. Comput. Phys. **113**(1), 13–25240 (1994). https://doi.org/10.1006/jcph.1994.1114

[3] Henshaw, W.D., Schwendeman, D.W.: An adaptive numerical scheme for high-speed reactive flow on overlapping grids. J. Comput. Phys. **191**, 420–44796 (2003). https://doi.org/10.1016/S0021-9991(03)00323-1

[4] Henshaw, W.D., Chand, K.K.: A composite grid solver for conjugate heat transfer in fluid-structure systems. J. Comput. Phys. **228**, 3708–374175 (2009). https://doi.org/10.1016/j.jcp.2009.02.007

[5] Henshaw, W.D.: A high-order accurate parallel solver for Maxwell's equations on overlapping grids. SIAM J. Sci. Comput. **28**(5), 1730–176555 (2006). https://doi.org/10.1137/050644379

[6] Henshaw, W.D., Chesshire, G.S.: Multigrid on composite meshes. SIAM J. Sci. Stat. Comput. **8**(6), 914–92347 (1987). https://doi.org/10.1137/0908074

[7] Liu, K., Henshaw, W.D.: Multigrid with nonstandard coarsening (2020) https://arxiv.org/abs/2008.03885 [math.NA]

[8] Stüben, K., Trottenberg, U.: Multigrid methods: Fundamental algorithms, model problem analysis and applications. In: Hackbusch, W., Trottenberg, U. (eds.) Multigrid Methods, pp. 1–176. Springer, Berlin (1982)

[9] Briggs, W.L., Henson, V.E., McCormick, S.F.: A Multigrid Tutorial. SIAM, Philadelphia (2000)

[10] Trottenberg, U., Oosterlee, C.W., Schüller, A.: Multigrid. Academic Press, London (2001)

[11] Wesseling, P.: An Introduction To Multigrid Methods. John Wiley & Sons, New York (1991)

[12] Brandt, A.: Multi-level adaptive solutions to boundary-value problems. Math. Comp. **31**, 333–390 (1977). https://doi.org/10.1090/S0025-5718-1977-0431719-X

[13] Stüben, K.: A review of algebraic multigrid. Journal of Computational and Applied Mathematics **128**, 281–309 (2001). https://doi.org/10.1016/S0377-0427(00)00516-1

[14] Chang, Q., Wong, Y.S., Fu, H.: On the algebraic multigrid method. J. Comp. Phys. **125**, 279–292 (1996). https://doi.org/10.1006/jcph.1996.0094

[15] Grauschopf, T., Griebel, M., Regler, H.: Additive multilevel preconditioners based on bilinear interpolation, matrix-dependent geometric coarsening and algebraic multigrid coarsening for second-order elliptic PDEs. Appl. Numer. Math **23**, 63–95 (1997). https://doi.org/10.1016/S0168-9274(96)00062-1

[16] Krechel, A., Stüben, K.: Operator dependent interpolation in algebraic multigrid. In: Hackbusch, W., Wittum, G. (eds.) Multigrid Methods V. Lecture Notes in Computational Science and Engineering vol. 3, pp. 189–211. Springer, Berlin (1998). https://doi.org/10.1007/978-3-642-58734-4_11

[17] Brezina, M., Falgout, R., MacLachlan, S., Manteuffel, T., McCormick, S., Ruge, J.: Adaptive algebraic multigrid. SIAM J. Sci. Comput. **27**, 1261–1286 (2006). https://doi.org/10.1137/040614402

[18] Hinatsu, M., Ferziger, J.H.: Numerical computation of unsteady incompressible flow in complex geometry using a composite multigrid technique. Int. J. Numer. Meth. Fl. **13**, 971–997 (1991). https://doi.org/10.1002/fld.1650130804

[19] Johnson, R.A., Belk, D.M.: Multigrid approach to overset grid communication. AIAA J. **33**(12), 2305–2308 (1995). https://doi.org/10.2514/3.12984

[20] Tu, J.Y., Fuchs, L.: Calculation of flows using three-dimensional overlapping grids and multigrid methods. Int. J. Numer. Meth. Eng. **38**, 259–282 (1995). https://doi.org/10.1002/nme.1620380207

[21] Zang, Y., Street, R.L.: A composite multigrid method for calculating unsteady incompressible flows in geometrically complex domains. Int. J. Numer. Meth. Fl. **20**, 341–361 (1995). https://doi.org/10.1002/fld.1650200502

[22] Yavneh, I.: On red-black SOR smoothing in multigrid. SIAM J. Sci. Comput. **17**(1), 180–192 (1996). https://doi.org/10.1137/0917013

[23] Rodrigo, C., Gaspar, F.J., Zikatanov, L.T.: On the validity of the local fourier analysis (2017) https://arxiv.org/abs/1710.00408 [math.NA]

[24] Hackbusch, W.: On multi-grid iterations with defect correction. In: Hackbusch, W., Trottenberg, U. (eds.) Multigrid Methods, pp. 461–473. Springer, Berlin, Heidelberg (1982)

[25] Bernert, K.: τ-extrapolation—theoretical foundation, numerical experiment, and application to Navier-Stokes equations. SIAM J. Sci. Comput. **18**(2), 460–478 (1997). https://doi.org/10.1137/S1064827594276266

[26] Hemker, P.W.: On the order of prolongations and restrictions in multigrid procedures. Journal of Computational and Applied Mathematics **32**, 423–429 (1990). https://doi.org/10.1016/0377-0427(90)90047-4

[27] Wienands, R.: Lfa00_2d_scalar. Technical Report http://www.mgnet.org/mgnet-codes-wienands.html, GMD - Institute for Algorithms and Scientific Computing (SCAI) (2000)

[28] Stüben, K., Trottenberg, U.: Multigrid methods: Fundamental algorithms, model problem analysis and applications. In: Hackbusch, W., Trottenberg, U. (eds.) Multigrid Methods, pp. 1–176. Springer, ??? (1982)

[29] Hassanieh, N.A., Banks, J.W., Henshaw, W.D., Schwendeman, D.W.: Local compatibility boundary conditions for high-order accurate finite-difference approximations of PDEs. preprint arXiv:2111.02915 (2021)

[30] Henshaw, W.D., Kreiss, H.-O., Reyna, L.G.M.: A fourth-order accurate difference approximation for the incompressible Navier-Stokes equations. Comput. Fluids **23**(4), 575–59352 (1994). https://doi.org/10.1016/0045-7930(94)90053-1

[31] Chesshire, G.S., Henshaw, W.D.: Composite overlapping meshes for the solution of partial differential equations. J. Comput. Phys. **90**(1), 1–64527 (1990)

Statements and Declarations

- **Funding**:
  Research supported by the National Science Foundation under grants DMS-1519934 and DMS-1818926.

- **Competing Interests**:
  The authors have no relevant financial or non-financial interests to disclose.

- **Data Availability**:
  Not applicable.