

# Automatic Algebraic Multigrid Solvers with Application on the 1D Helmholtz Equation

June 4, 2021

## Abstract

Multigrid solvers have been successful in solving numerous scientific and engineering systems. Typically, a hierarchy of increasingly coarser levels are constructed from the original fine equation, such that each level reduces a certain range of error components, and combined in a multigrid cycle that has an optimal convergence rate. Notwithstanding, optimal multigrid design is still very difficult to design for indefinite problems, highly nonlinear problems, and huge systems where even linear complexity is not good enough. Algebraic multigrid solvers automate the coarsening process, but are still limited in application.

In this work we develop AutoAMG, a new Bootstrap Algebraic Multigrid (BAMG) algorithm for constructing an efficient multilevel hierarchy. We focus on developing automatic criteria for deriving the coarse variables (based on local singular value decompositions) and interpolation operators (via regularized least-squares fitting) with as little assumptions as possible about the input system. The approach is demonstrated on the one-dimensional Helmholtz equation on a periodic domain. This model is an example where standard multigrid solvers has been notoriously difficult to develop, as the coarse variables have a completely different character (rays) than fine-level variables (waves). AutoAMG produces both near-null space functions, and a multilevel hierarchy of highly accurate coarse levels.

We discuss future generalizations to the multi-dimensional Helmholtz equation, and systematic upscaling of highly repetitive non-linear systems arising in fluid flows and molecular dynamics.

## 1 Introduction

Multigrid (MG) solvers have been very successful in solving a wide variety of scientific and engineering problems with optimal efficiency [?]. Typically, a hierarchy of coarse levels is constructed from the original system, and combined in a multigrid cycle to obtain a fast iterative solver. There are several reasons for further generalizing these multigrid methods: (a) it takes expertise and tedious manual development process to tailor the methods to *systems* of Partial

Differential Equations (PDEs) and *indefinite problems*; (b) they do not perform well for *highly nonlinear cases*, where configurations cannot be decomposed into weakly-interacting local and non-local parts; and (c) for many systems, even attaining linear complexity is not good enough, since the number of variables is huge. Such systems on the other hand are typically *highly repetitive*, in the sense that the same small set of governing equations (or Hamiltonian terms) keep repeating itself throughout the physical domain. In this work we focus on (a). The methodology can be generalized to address (b) and (c) via *systematic upscaling* [?], as discussed in Sec. 5.2.

We pick a specific model problem: the 1D Helmholtz equation on a periodic domain, discretized on a uniform grid of a specific size. There are two main difficulties here: (a) the coarse-level variables have a completely different character (rays/geometric optics) than fine-level variables (waves/electromagnetics). The proper type of variables needs to be automatically selected at each level; (b) the Helmholtz operator has Almost Zero Modes (AZMs; i.e., near-null space components), which are not reduced by the fine-level relaxation and need to be accurately approximated by coarse levels. Unlike diffusion PDEs, these are not geometrically smooth functions. In 1D, they can be represented by two basis functions  $\varphi_-$  and  $\varphi_+$ . That is, each AZM can be written as  $A_- \varphi_- + A_+ \varphi_+$  where  $A_{\pm}$  are smooth functions (“rays”). (The 2D problem introduces the further difficulty that AZMs are composed of waves in an infinite number of directions.

Creating a multigrid solver that properly approximates AZMs and automatically detects the transition from waves to rays, has been a notoriously difficult task [?, ?].) Our algorithm finds both low-residual “test functions” that contain linear combinations of those AZMs, and a multilevel hierarchy that accurately represents them at all levels.

This setting factors out other difficulties that a numerical solver of the PDE must treat (radiation boundary conditions are necessary; boundary effects; and diffraction effects of domain features of size comparable with the wavelength; phase error), to focus on the derivation of interior-domain coarsening.

## 1.1 Our Contribution

We develop AutoAMG, a multilevel algorithm finds low-residual test functions and an accurate multilevel hierarchy. AutoAMG is a completely automatic Bootstrap Algebraic Multigrid (BAMG) procedure that does not exploit specific properties of the Helmholtz operator, such as the exponential form the AZMs, which is much simpler than a specifically tailored approach [?, ?]. In particular, it provides a stepping stone towards a multilevel solver of the Helmholtz equation.

Furthermore, we focused on deriving general principles and rules so that the coarsening proces could be also applied to other problems, including nonlinear problems and systems of PDEs (such as the Navier Stokes equations). It can potentially be used for a fast iterative solution, systematic upscaling (coarsening of a repetitive equation on a large domain using local windows), and numerical homogenization, in which the fine levels are never or rarely visited during the

simulation.

Bootstrapping is an essential component in generating accurate coarse levels. We build on top of recent advances in multilevel methods (adaptive MG, BAMG [?, ?, ?] and MG for neural networks - **pending publication, add citation**) to improve the various multigrid components beyond the framework outlined in [?]:

- Coarse variables (both their type and number) are automatically derived using a local Singular Value Decomposition (SVD), as opposed to expert knowledge (e.g., averages of fine variables in PDEs, or centers of mass in molecular dynamics). We do utilize the locality graph, though, by defining an aggregate as a small contiguous segment of gridpoints, although the approach can be generalized to non-geometric and non-local cases as well (cf. Sec. ??).
- Interpolation is derived by least-squares. We use a large set of samples and a validation set (as well as potentially regularization) to avoid over-fitting.
- The coarsening and interpolation are efficiently derived on small, local aggregates. This approach lends itself to parallelization, and can be generalized to upscaling: deriving a local coarsening of a repetitive system on a large domain, e.g., the Helmholtz equation with a constant wave number on a meshsize small enough to obtain a small pointwise discretization error; cf. Sec. 5.2

## 2 Goal

We consider the 1D Helmholtz equation

$$(Au)(x) := \Delta u(x) + K(x)^2 u(x) = F(x), \quad x \in [0, L], \quad (1)$$

with periodic boundary conditions  $u(x) = u(x + L)$ . The problem is discretized on a uniform grid with  $n$  points  $x_j = jh$ ,  $j = 0, \dots, n-1$ ; references to indices larger than  $n-1$  or negative are understood as periodic, i.e.,  $j \bmod n$ , and meshsize  $h = L/n$ . We denote the discrete approximation by  $u^h := (u_0^h, \dots, u_{n-1}^h)$  and the discrete algebraic system by

$$A^h u^h = f^h. \quad (2)$$

Our goals are (a) find  $M$  *test functions*:  $u_1$  *dots*,  $u_M$  of unit  $l_2$  norm with small  $\|A^h u^h\|$ ; i.e., near-null-space components of  $A^h$ ; (b) construct a multilevel hierarchy that accurately approximates the test functions, and gives rise to an optimal linear solver of (2), i.e., it attains  $\varepsilon$ -residual accuracy  $\|A^h u^h - f^h\| \leq \varepsilon$  in  $O(n \log 1/\varepsilon)$  operations.

The meshsize  $h$  is assumed to be small enough to resolve the wavelength and provide good local pointwise discretization accuracy. (We do not worry at this point about decreasing  $h$  further to obtain pointwise accuracy on a

large domain that does not have a phase error, as in such resolutions simple geometric coarsening can be employed. Thus they are irrelevant to the challenge of coarsening away from boundaries.) For simplicity (to avoid special treatment near boundaries), we assume that  $n$  is a power of 2.

Our test functions are an approximation to the eigenvectors of  $A$  corresponding to the  $M$  smallest eigenvalues. We do not insist on finding the exact orthogonal basis here; we do know that for the 1D Helmholtz case, a small  $M$  should be sufficient to represent the subspace of AZMs required for an accurate multilevel hierarchy

## 2.1 Discretization

To ensure that special properties of tridiagonal matrices are not being exploited (as in total reduction), do not mask the true efficiency of the algorithm, we test both the 3-point second-order finite difference discretization

$$(A^h u^h)_j = u_{j-1} + (-2 + k_j^2)u_j + u_{j+1} = f_j$$

and 5-point fourth-order discretization

$$(A^h u^h)_j = \frac{1}{12} (-u_{j-2} + 16u_{j-1} + (-30 + 12k_j^2)u_j + 16u_{j+1} - u_{j+2}) = f_j,$$

for all  $j = 0, \dots, n-1$ , where  $f_j := h^2 f(x_j)$  and  $k_j := hK(x_j)$ . Since this is a homogeneous equation on a periodic domain, the stencils are normalized (i.e., scaled by  $h^2$ ), so the operator only depends on  $k = Kh$ .

## 3 Adaptive Multilevel Hierarchy

Since  $A$  is indefinite, we use a robust smoother: Kaczmarz relaxation in lexicographic ordering. While for special cases (e.g.  $kh = 0$ ), better smoothers are available (e.g., red-black Gauss-Seidel), Kaczmarz relaxation is guaranteed to smooth the error [?], so we keep the approach general.

### 3.1 Initial Vectors

Initial vectors are obtained by  $\nu$  Kaczmarz relaxations on  $Ax = 0$ , starting from  $\text{rand}[-1, 1]$ .  $x$  is normalized to unit  $l_2$  norm at the end. There is no need to normalize after each relaxation sweep, since we are solving a homogeneous equation (and anyway, the norm doesn't change much unless a huge number of relaxations is performed).

**What  $\nu$  should we use to get a good enough initial guess for two-level bootstrapping?**

## 3.2 Coarse-level Construction

We omit the  $h$  superscripts in this section and use quantities without superscripts (e.g.,  $A, x$ ) to denote fine-level quantities and  $c$ -superscripted quantities (e.g.,  $A^c, x^c$ ) to denote coarse-level quantities.

### 3.2.1 SVD Coarsening

Given a fixed-size discrete domain with  $n$  and test functions  $x_1, \dots, x_M$  on this domain, we derive the coarsening matrix  $R_w : w_c \times w$  separately in each *aggregate* (consisting of  $w = 2 - 4$  consecutive gridpoints). The coarsening matrix over the entire domain is the block diagonal matrix  $R : n_c \times n$

$$R := \text{diag} \left\{ \underbrace{R_w, R_w, \dots, R_w}_{\frac{n}{w}} \right\}. \quad (3)$$

$R_w$  depends on an accuracy threshold  $\varepsilon$  and minimum coarsening ratio  $0 < \rho < 1$ . Start with  $w = 2$ . The test functions provide  $M$  samples of low-residual functions over the aggregate,  $u_j^m = (u_{m,j}, \dots, u_{m,j+w})$  for  $m = 1, \dots, M$ . We compute the SVD decomposition  $X = U\Sigma V^T$  of the  $m \times w$  matrix  $X$  whose rows are the windows, where  $\Sigma = \text{diag} \{\sigma_1, \dots, \sigma_m\}$  and  $\sigma_1 \geq \sigma_2 \geq \dots$ . Let  $w_c$  be the smallest number such that

$$\left( \frac{\sum_{j=n_c+1}^m \sigma_j^2}{\sum_{j=1}^m \sigma_j^2} \right)^{\frac{1}{2}} < \varepsilon. \quad (4)$$

Then  $R_w := V(1 : w_c, :)$  is the sub-matrix of  $V$  consisting of the first  $w_c$  rows of  $V$ . (4) guarantees an  $\varepsilon$ -relative reconstruction error of  $X$  in the Frobenius norm [?].  $M$  must be larger than  $w_c$ , say  $M \geq 4w$ , to avoid SVD over-fitting.

If  $w_c/w > \rho$ , we double the window size  $w$  and recalculate  $R$  until a large enough window is reached such that  $w_c/w \leq \rho$ . If the relaxation method is a good smoother that quickly reduces the error (and thus our test functions, since we relax on  $Ax = 0$ ) into a lower-dimensional space, such a value of  $w$  can be found. **How can we determine  $M$  in advance if we don't know the window size  $w$  in advance?**

The SVD yields coarse variables that are less sensitive to relaxation than pointwise coarsening (i.e., selecting one or several fine gridpoints as coarse variables, which corresponds to  $R_w = [1, \underbrace{0, \dots, 0}_{w-1}]$ ), as shown by comparing mock

cycle rates Sec. 4.3. It also automatically selects the number of coarse variables  $w_c$  per aggregate.

**Add explanation on the particular nature of the coarse vars for the 1D Helmholtz: two waves, i.e., combination of sines and cosines. Add figure.**

### 3.2.2 Least-squares Interpolation

As for  $R$ , we derive an interpolation matrix  $P_w$  on an aggregate of size  $w$ , and then create a block diagonal  $P_{n \times n_c}$  interpolation matrix over the entire domain from the aggregate interpolation matrices.

The *interpolatory set* (the set of coarse variables used to interpolate to a target fine gridpoint) can be the set of coarse variables of its aggregate. In this case, the optimal interpolation that minimizes the interpolation error of the test functions is  $P_w = R_w^T$  (as the truncated SVD is the best low-rank approximation in the Frobenius norm).

While this choice may be sufficient in some problems, even in the Laplace case ( $k = 0$ ),  $R_w^T$  is a piecewise-constant geometric interpolation (with the window size  $w = 2, w_c = 1$ , where only one principal component already explains most of the energy – the constant vector), which only yields marginal multilevel efficiency (two-level, but not multilevel  $h$ -independent convergence). Linear interpolation is required to obtain the full multilevel efficiency.

Thus, we increase  $P_w$ 's accuracy by including in the interpolatory set as many coarse variables as possible without increasing the coarse operator's density. A general rule is to use the union of all coarse variables in all aggregates of the fine gridpoints appearing in the target gridpoint's stencil.

The *interpolation weights* are determined by least-square fitting to test samples.

**include a description of the LS process: fitting set, fit with regularization parameter  $\alpha$ . Validation set: use to determine  $\alpha$ .  $\alpha$  is global across all fine gridpoints in an aggregate. Test set: test the interpolation accuracy on samples we haven't used to ensure no over-fitting.**

$$\min_p \sum_m \left( x_{m,i} - \sum_j p_{ij} x_{m,j} \right)^2 + \alpha \left( \sum_m (x_{m,i})^2 \right) \left( \sum_j p_{ij}^2 \right) \quad (5)$$

### 3.3 Multilevel Bootstrapping

## 4 Numerical Results

We test two discretizations of the Helmholtz operator  $A = -\Delta + k^2 I$ : a 3-point second order stencil, and a 5-point 4th order finite-differences  $A^h = [-1, 16, -30 + 12(kh)^2, 16, -1]$ . The 5-point stencil avoids any properties particular to 1D tridiagonal systems that lead to easier coarsening (e.g., as in total reduction).

## 4.1 Quantitative Performance Prediction

### 4.2 Smoothing Factor

The Fourier mode analysis is done for a fixed  $k = Kh$ .

The traditional smoothing factor [?, Sec. ??] assumes that the coarse level perfectly reduces all scaled frequencies  $\theta$  with  $|\theta| \leq \frac{\pi}{\rho}$  and leaves intact all high frequencies  $|\theta| > \frac{\pi}{\rho}$ , where  $\rho$  is the coarsening factor. However, since our coarsening is designed to represent near null-space components of the Helmholtz operator instead of smooth functions, the smoothing factor definition should be modified to

$$\bar{\mu} := \max_{\theta \in HF} |R(\theta)|, \quad (6)$$

where  $R$  is the relaxation amplification factor. HF is the set of relevant high-frequency components: the set of  $[n/\rho]$  frequencies  $\theta$  with smallest  $|A(\theta)|$  value.

**TODO: James and Karsten calculated the traditional Kaczmarz smoothing factor to be 0.75, but for which  $k$  and coarsening factor? Add a table here of  $\bar{\mu}$  vs.  $k$  and  $\rho$ . Also add a column with the smoothing factor to the mock cycle table below so we can compare relevant numbers. Also note: can calculate  $\bar{\mu}$  by running Fourier components through Kaczmarz to numerically calculate its amplification matrix.**

### 4.3 Mock Cycle

To gauge the quality of the coarse variable set, we compared the mock cycle convergence factors with SVD coarsening, 1 : 2 pointwise coarsening ( $R_w = [1, 0]$ ) and averaging coarsening with 1 : 2 coarsening ratio ( $R_w = [1, 1]$ ). for different  $kh$  values and  $\nu$  relaxations per cycle. Kaczmarz relaxation was used in all cases; for  $kh = 0$  only, we also tested Gauss-Seidel relaxation for comparison. The SVD coarsening was based on test vectors obtained by 100 relaxations of random  $[-1, 1]$  starting vectors, a  $w = 4$ -point aggregate to calculate the SVD, and accuracy threshold of  $\varepsilon = 0.1$ . Two principal components were selected as coarse variables for all  $kh$ , so all three coarsening schemes have coarsening ratio  $\rho = 0.5$ .

The pointwise coarsening convergence factors is bounded by 0.5 as the error function  $1, 0, 1, \dots$  is reduced only by a meager 0.5 factor by relaxation. In practice, a multigrid cycle with pointwise coarsening converges faster. Generally, mock cycle predictions can be pessimistic, and cases where their prediction is quantitatively accurate should be identified based on numerical experimentation. Here, it is more accurate when the coarse variables are averages. As expected, the SVD coarsening is equivalent to averaging for the Laplace case, but trumps simple averaging for  $k = O(1)$ , which is substantiates its use for upscaling. The convergence improves with  $\nu$  up to 3 relaxation sweeps.

Finally, in line with geometric multigrid theory, Gauss-Seidel is a better smoother for the Laplace case, but Kaczmarz relaxation provides adequate con-

$k$	SVD Coarsening				Pointwise Coarsening				Averaging Coarsening			
	$\nu = 1$	2	3	4	1	2	3	4	1	2	3	4
GS	0.30	0.11	0.07	0.02	0.65	0.60	0.57	0.54	0.32	0.12	0.07	0.02
0.0	0.42	0.22	0.15	0.12	0.49	0.55	0.49	0.50	0.45	0.21	0.19	0.15
0.1	0.46	0.22	0.15	0.13	0.50	0.55	0.48	0.48	0.42	0.20	0.19	0.15
0.2	0.45	0.24	0.15	0.12	0.51	0.58	0.50	0.48	0.45	0.21	0.19	0.16
0.3	0.46	0.23	0.15	0.12	0.49	0.56	0.50	0.49	0.42	0.20	0.20	0.15
0.4	0.49	0.24	0.16	0.12	0.50	0.58	0.53	0.49	0.49	0.20	0.20	0.18
0.5	0.51	0.25	0.16	0.13	0.53	0.55	0.49	0.47	0.47	0.22	0.21	0.20
0.6	0.52	0.30	0.13	0.14	0.56	0.52	0.53	0.48	0.52	0.22	0.20	0.21
0.7	0.57	0.25	0.15	0.13	0.61	0.51	0.53	0.46	0.54	0.25	0.22	0.24
0.8	0.60	0.30	0.15	0.11	0.64	0.44	0.50	0.52	0.49	0.24	0.23	0.26
0.9	0.63	0.35	0.17	0.15	0.67	0.39	0.49	0.51	0.63	0.34	0.20	0.27
1.0	0.60	0.29	0.14	0.10	0.74	0.46	0.43	0.50	0.69	0.48	0.25	0.25

Table 1: Mock cycle convergence factors for a grid of size  $n = 32$ . The first row corresponds to the Laplace operator and Gauss-Seidel. All other rows refer to Kaczmarz relaxation.  $\nu$  is the number of relaxations per cycle.

vergence as well in that case, while its efficiency is maintained for all values of  $0 \leq k \leq 1$ . All results are independent of  $n$ , for  $n \geq 16$ .

#### 4.4 2-level

- Start with  $n = 96$  at level 1.
- Show 2-level rates of solving  $Ax = 0$  for  $n = 96$  as a function of number of bootstrap cycles. Test function residuals.
- Then show convergence for increasingly larger  $n$  with the same coarsening.
- How do the SVD components look like.
- Performance vs. window size. Number of components.

#### 4.5 3-level

#### 4.6 Wave to Ray Transition

Show limit of stencil coefficients of coarse equations after many coarsening levels and after few. Is it elliptic / Ray?



## 5 Future Directions

### 5.1 The Multi-dimensional Helmholtz Equation

### 5.2 Systematic Upscaling

Many physical systems involve a huge number of variables, such that even attaining linear complexity is not good enough, since the number of variables is huge. Such systems on the other hand are typically *highly repetitive*, in the sense that the same small set of governing equations (or Hamiltonian terms) keep repeating itself throughout the physical domain.

Consider for instance the 1D Helmholtz equation with constant  $K(x) \equiv K$  on a large domain of size  $L$ . AutoAMG can be adapted to take advantage of the periodicity to derive the coarse equations locally, and use the

- **Step 1: local derivation but still using the entire level for relaxation.** We start with a small  $L$  and construct the AutoAMG hierarchy and test functions. The coarsening matrix is derived on a single aggregate; that is, we compute  $R_w : w_c \times w$  of an *aggregate* (consisting of  $w = 2 - 4$  consecutive gridpoints) using the SVD, and then tile (copy) these coarse variables to obtain an  $n_c \times n$  block diagonal coarsening matrix  $R$

$$R := \text{diag} \left\{ \underbrace{R_w, R_w, \dots, R_w}_{\frac{n}{w}} \right\}. \quad (7)$$

To calculate the local SVD and  $R^w$ , we can derive from each test function *many test samples*:  $n$  windows of size  $w$ :  $x_i^k = (x_{k,i}, \dots, x_{k,i+w})$ , a total of  $Kw$  windows. Thus, in fact, a very small  $M$  (even  $M = 1$ ) is sufficient here.  $P$  is similarly derived over a single aggregate and tiled to the entire domain.

- **Step 2: domain growing; but still using the entire fine level for relaxation.** We then double  $L$  and tile the test vectors and  $P$ ,  $R$  and  $A^c$  over the twice-larger domain. For each domain doubling, we add another coarse level to the hierarchy.
- **Step 3: use only local windows on the fine levels and FAS** to obtain sub-linear complexity.

When do we grow the domain? Do we add more levels on a fixed-size domain of size  $n$  gridpoints until we get to  $O(\log n)$  levels and a coarsest grid with only a few points, before growing the domain, so that we get a full multilevel cycle that reduces the error by a fixed factor in  $O(n)$  operations; or, do we start with level 1 on  $n$  points, create level 2 with  $n/2$  points, improve the hierarchy with 2-level cycles, then consider level 2 on  $n$  points as our finest level, and coarsen it using 2-level cycles, and so on, i.e., recursive homogenization with 2 level-cycles only, but increasingly larger domains?

### 5.3 $\tau$ -Homogenization

The FAS coarse-level equations depend on the fine level only through the fine-to-coarse  $\tau$  correction:

$$A^c x^c = P^T b + \tau, \quad \tau := A^c \bar{x}^c - P^T A x = (A^c R - P^T A) x, \quad (8)$$

where  $\bar{x}^c := R x$  is the restricted fine-level approximate solution (and initial coarse level approximation). Thus,  $\tau_I$  depends on a local neighborhood of fine-level values near the coarse gridpoint  $I$ , and thus in turn on neighboring coarse-level values  $x_J^c$ . We can thus attempt to *homogenize* (8) to only depend on  $x^c$  by fitting a rule  $\tau \approx Q x^c$  with some interpolation matrix  $Q$ .  $Q$  can be found using a least-squares fit, just like for the BAMG interpolation  $P$ , namely,

$$\min_p \sum_m \left( \tau_{m,I} - \sum_j q_{IJ} x_{m,J}^c \right)^2 + \alpha \left( \sum_m (\tau_{m,I})^2 \right) \left( \sum_j q_{IJ}^2 \right). \quad (9)$$

- Non-linear problems (e.g., fluid flows): use FAS; fine-level windows for different coarse solution values.
- Molecular dynamics:  $\tau$  homogenization may lead to discovering different physical laws at different scales.
- Non-geometric and non-local problems (e.g., NNs): use correlations instead of locality graph to define aggregates.