AMG+: Extended Principles
Illustrated on the 1D Helmholtz Equation

Achi Brandt
James Brannick
Karsten Kahl
Oren Livne

Numerical Analysis and Lattice QCD
CIRM - August 15, 2022

<u>Paris to Fréjus:</u>

July 1-7, 1990

1020km: $\uparrow 8.7km$

Saint-Gervais-les-Bains:
Le Brévent

AMG+

MOTIVATION

Make AMG work for many new problems
Formulate generalized guiding principles

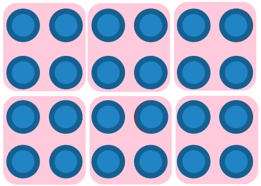
Multilevel Methods

Multitude of variables/unknowns

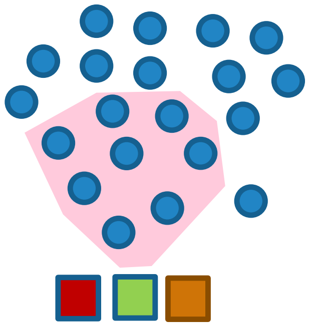
➡ **multilevel organization**

- At each level:
  1. Relaxation
  2. Identifying coarse-level variables
  3. Interpolation
  4. Coarse-level equations
- For each step:
  1. Quality Measure
  2. Construction method

Coarse Variable construction



- AMG: coarse variable = a representative/average of several “strongly connected” fine variables.



AMG+

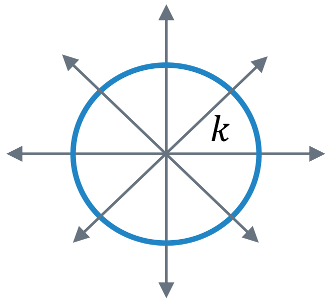
- In many problems (non-elliptic, NN, …) there may not be particularly strong connections.
- Can coarsen only larger aggregate of moderately connected variables, with several coarse variables per aggregate.
- Coarse variables are of different type than fine vars.

PDE Example: Helmholtz Equation

$$[\Delta + k^2]u(x,y) = f(x,y), \qquad\qquad k = k(x,y)$$

Constant k: slowly converging errors are

$$\sum_{\alpha^2+\beta^2=k^2} A_{\alpha\beta} e^{i(\alpha x+\beta y)} = \sum \text{"}ray\text{"}$$



- ▷ Moderate correlation between neighboring error values.
- ▷ "Multi-coarsening" is required.
- ▷ Past **specialized** multi-coarsening **failed**.
- ▷ The **general** systematic approach proposed here **works**.

AMG+: Guiding Quality Measures

- **Relaxation:** Residual shrinkage factor/work
- **Coarse variables:** Mock cycle convergence factor
- **Interpolation:** Test functions
- **Coarse equation:** 2-level convergence factor
  Sparsity, symmetry

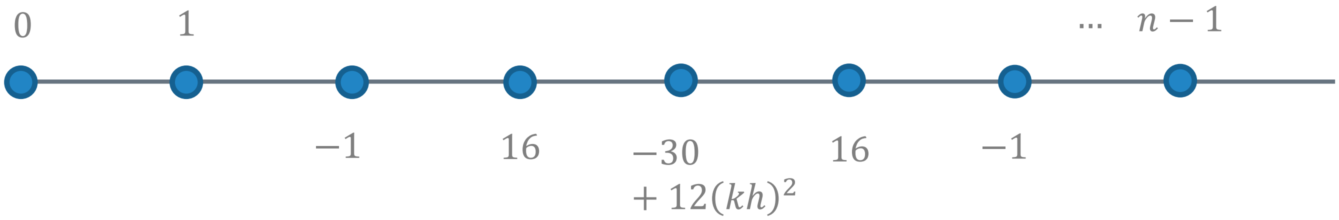
predicts

AMG+ for 1D Helmholtz

PDE Example: 1D Helmholtz

$$[\Delta + k^2]u(x) = f(x) \qquad x \in [0, L)$$

$$u(x + L) = u(x)$$



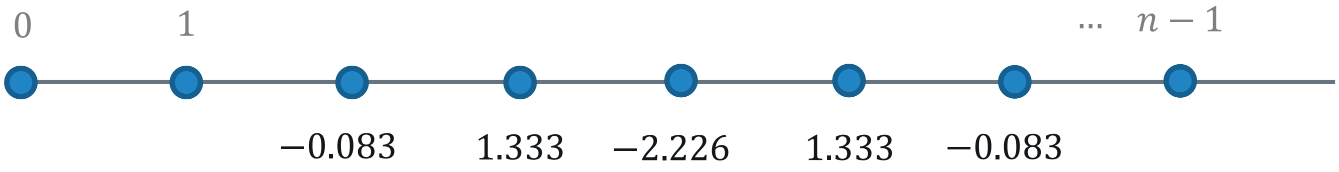
$$Ax = b$$

- Discretization: 5-point, $O(h^4)$-accurate.
- Solve fast on a fixed domain size.

PDE Example: 1D Helmholtz

$$[\Delta + k^2]u(x) = f(x) \qquad x \in [0, L)$$

$$u(x + L) = u(x)$$



$$Ax = b$$

- Fixed domain size. $L = n, n = 96, kh = 0.523$ (difficult case in practice)

Repetitiveness
We exploit the equations' repetitiveness for simplicity & eyeing upscaling.

1. Relaxation

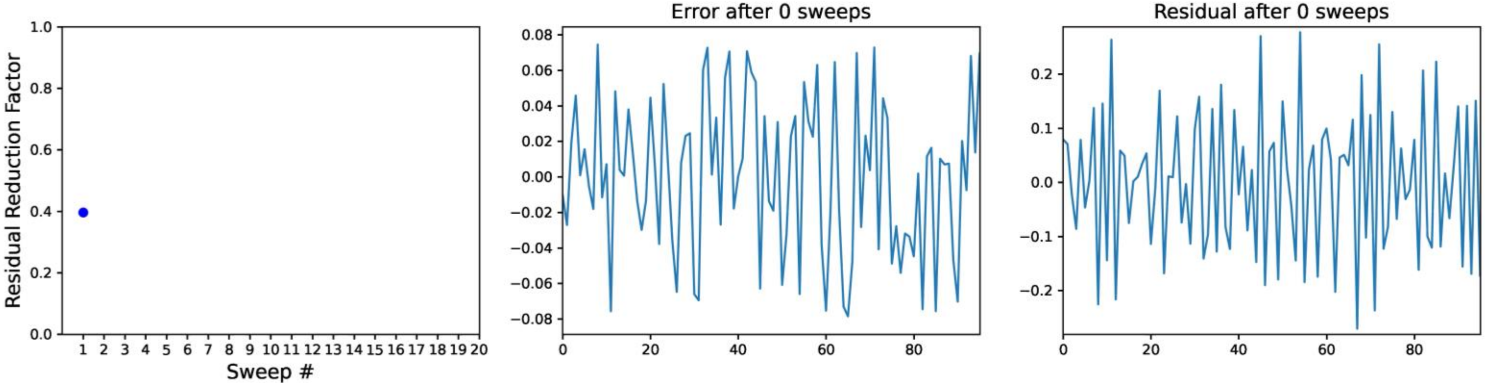
should exhibit a fast initial residual reduction, starting from a random initial error.

Assessing Relaxation: Shrinkage Factor

- GMG: error is smoothed; smoothing factor.
- AMG: shrink the error's information content: $||r|| \ll ||A||||e||, \quad r = Ae$.
- AMG+: Relax $Ax = 0$, starting from $rand[-1,1]$.
    - $\mu_\nu = (||r_\nu||/||r_0||)^{1/\nu}$, $\bar{\mu} := \mu_\nu$ = **shrinkage factor**, at point of diminishing returns.
    - Average conv factor over 5 cases.

Shrinkage Factor In Action

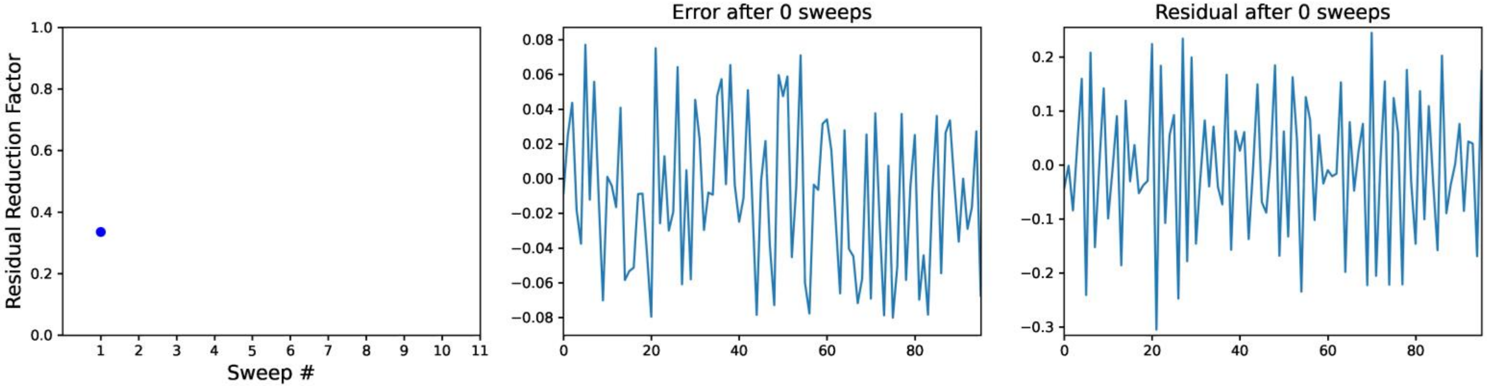
The shrinkage's ν tells us how many fine-level sweeps to use in the cycle.



Laplace GS

$\bar{\mu} = .5$

$\bar{\nu} = 6$



Helmholtz Kaczmarz

$\bar{\mu} = .38$

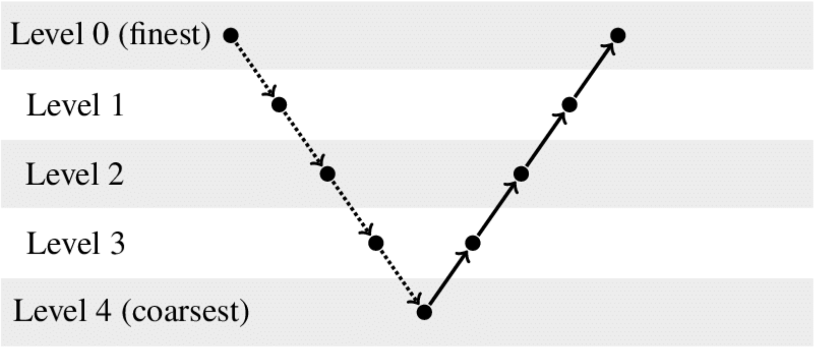
$\bar{\nu} = 2$

2. Test Functions

are examples of slowly-converging errors,
which reveal connection strength between variables.

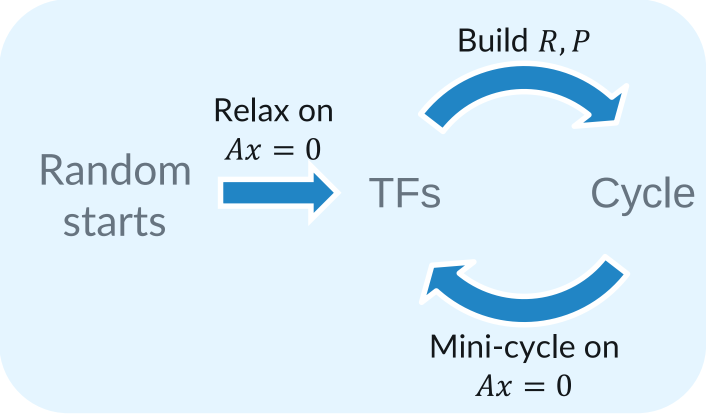
Quick Notation

- **Coarsening**: the definition of coarse variables. R
- **Interpolation**: transfer coarse vector to fine level. P
- **Restriction**: transfer fine vector to coarse level. Q For instance, R; P^T



Test Functions (TFs)

- Examples of slowly converging errors.
- Finding Neighborhoods
  - Neighbors are **highly correlated across TFs.**
  - May be very different from strong coupling.
- Bootstrap
  - Improve TFs while adding levels.



In non-linear systems,
can still produce such examples.

Separating species of vars
is important in systems.

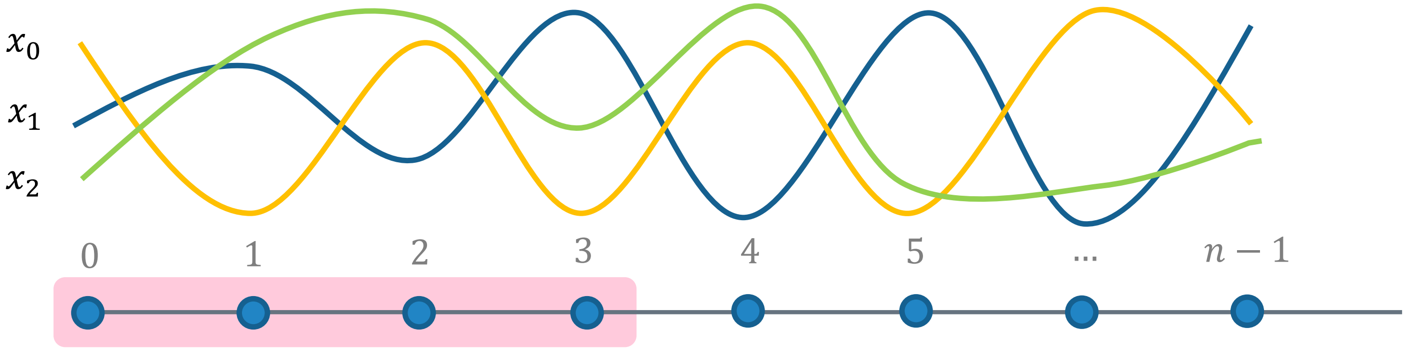
Near null-space functions
If exist, are eventually revealed as slow-to-converge functions in the cycle.

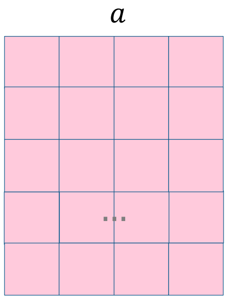
3. SVD Coarsening

reveals the right type of coarse variables, and **guides** the choice of coarsening ratio.

Sample Windows from single TF

$$X = U\Sigma V^T$$



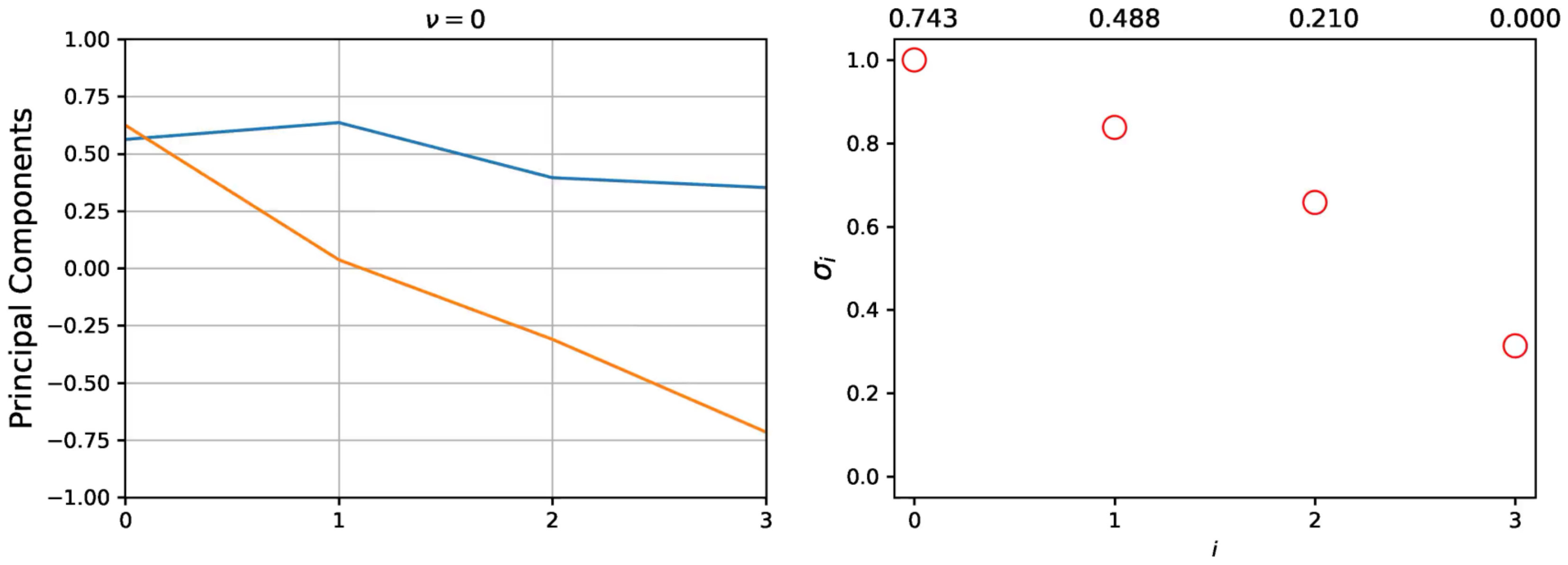


- Windows of size a.
- Coarse variables $R_{n_c \times a}$ = n_c principal components of X
- Compute the interpolation stencil(s) from samples obtained from windows (assuming shift invariance).

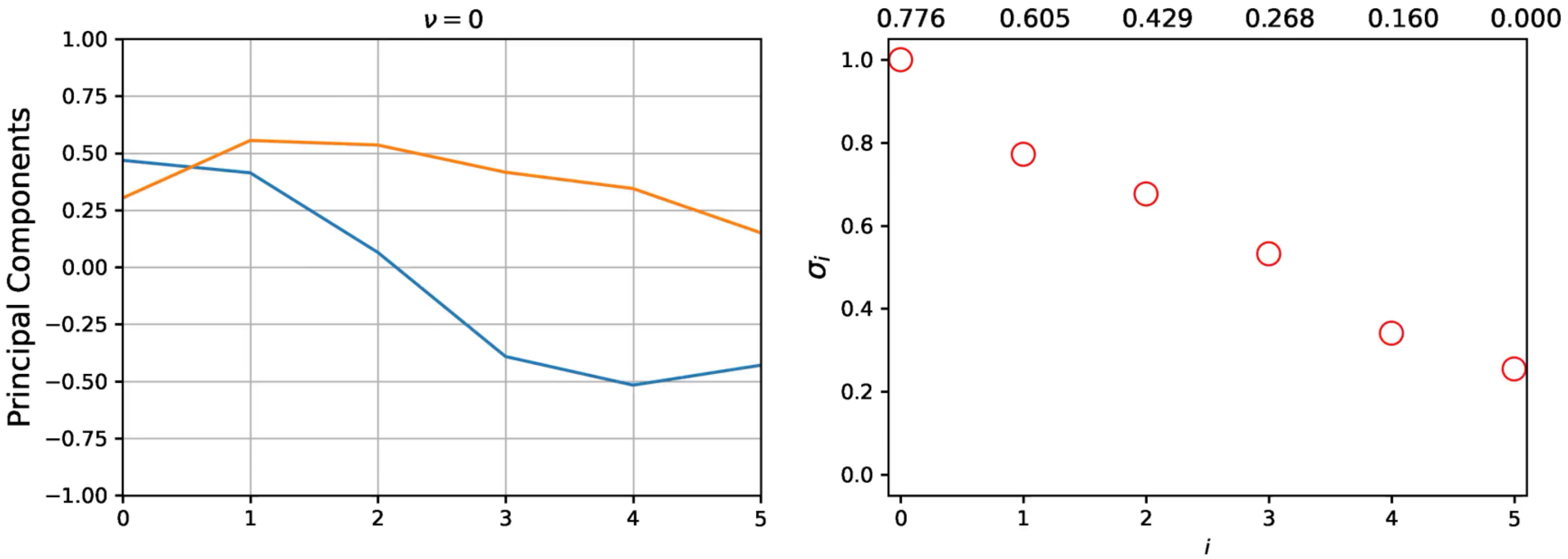
Multiple coarse vars per aggregate.
As in SA interpolation.

Repetitive case: tile R
From aggregate to entire domain.

SVD Coarsening in Action: Agg Size = 4



SVD Coarsening in Action: Agg Size = 6



n_c = # coarse variable **species.**

Helmholtz: $n_c = 2$ is natural: cos/sin, left/right waves.

How many components to use?

- SVD is best rank-n_c Frobenius approximation. $||X - XR^T R||_2$
- Unexplained variance = relative interpolation error $\left(\sum_{i \geq n_c} \sigma_i^2 / \sum_i \sigma_i^2\right)^{1/2}$
- For Laplace, R is piecewise constant, so error can be large!

Use SVD only to provide **tentative** values of coarsening ratios.
The actual number of components to be determined by the **quantitative predictor of cycle convergence**.

4. Quantitative Quality Prediction

allows designing each multilevel component separately and reliably (coarsening & cycle).

Mock Cycle: Predictor of 2-level Convergence

- Start with random $x = x_0$.
- $\mathrm{MockCycle}(R, \nu)$:
  - Relax ν times.
  - Update x such that $Rx = Rx_0$.

Direct solver/Kaczmarz
To project.
$x \leftarrow x - R^T (RR^T)^{-1} R(x - x_0)$

Use Mock Cycle asymptotic convergence factor as the **ultimate** coarsening quality test.
Determines the optimal aggregate size and #components.

Mock Cycle: Predictor of 2-level Convergence

- Start with random $x = x_0$.
- $\mathrm{MockCycle}(R, \nu)$:
  - Relax ν times.
  - Update x such that $Rx = Rx_0$.

Direct solver/Kaczmarz
To project.
$x \leftarrow x - R^T(RR^T)^{-1}Rx$

Accurate for $X \sim A$ (A, X SPD), where

$$\|\pi_X(R) - \pi_A(R)\|_A^2 = \|\pi_X(R)\|_A^2 - 1 \quad (\text{Mock cycle: } X = I)$$

Mock Cycle in Action: Convergence

	$\nu = 1$	2	3	4	5	6	7
4/2	.49	.29	.18	.12	.088	.056	.054
6/3	.52	.29	.18	.12	.098	.078	.064
6/2	.84	.67	.62	.53	.45	.37	.33

Guides the coarsening R.
Compare different choices. Reduction per relaxation sweep ≈ smoothing rate, up to accuracy limit.

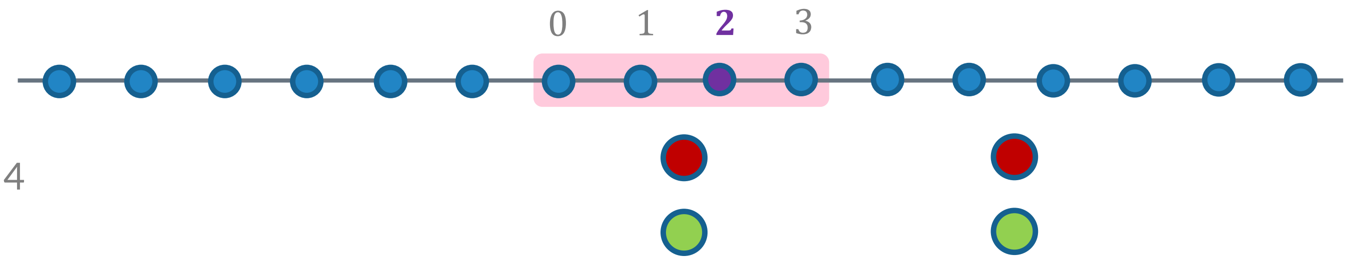
Guides the interpolation P.
2-level rates should attain mock cycle rates.

Prefer small aggregate size.
Aggressive coarsening can be too ambitious.

5. Interpolation

is constructed to accurately reproduce TFs, but tested via two-level cycle rates.

Least-Squares Fitting to TFs



Caliber = 4

$P = R^T$

$$\min_{p_i} \sum_s \left(x_{is} - \sum_j p_{ij} x_{js}^c \right)^2 \qquad i = 0..a-1; \quad x^c = Rx.$$

P_i tiled from aggregate to entire domain

Mock cycle predicts the ‘ideal’ 2-level rates

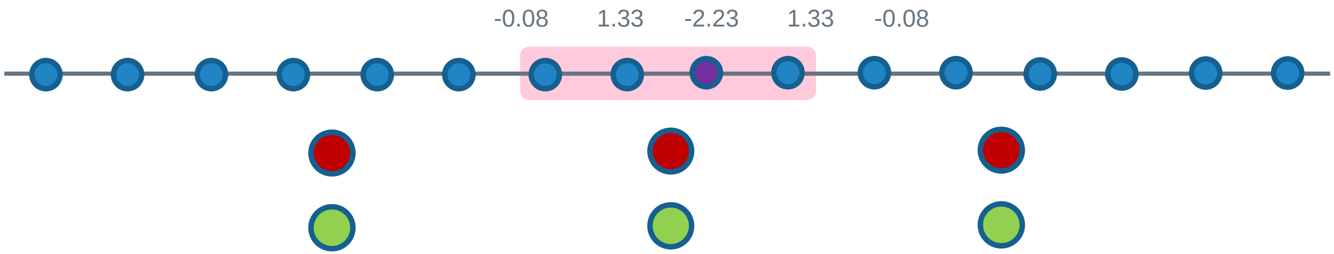
	$\nu = 1$	2	3	4	5	6	7	Fill-in	$\lvert A - A^T \rvert$
4/2	.49	.29	.18	.12	.09	.06	.05	-	-
RAR^T	.58	.46	.41	.37	.36	.32	.32	1.2	0
RAP	.51	.28	.17	.12	.09	.07	.06	1.2	0.0046
$P^T AP$	.52	.27	.14	.09	.07	.05	.04	2.0	0

P: caliber 4

2-level rates guide the choice of P.

TF reconstruction l_2 or energy error are bad predictors.

Coarse-level Equations



Wave Equation

RAR^T

0.20	0.34	-0.17	0.01	0.20	-0.32
-0.32	-0.56	0.01	-1.46	0.34	-0.56

Ray Equations

RAP

0.15	0.17	-0.03	0.00	0.15	-0.17
-0.17	-0.16	0.01	-0.63	0.17	-0.16

6. Bootstrap

Improves and reveals test functions as more levels are gradually added to the multilevel solver.

3-level Cycle Works

	$\nu = 3$	4	5	6	7	Fill-in	$\lvert A - A^T \rvert$
Mock	.18	.12	.09	.06	.05	-	-
0 -> 1	.36	.24	.23	.23	.11	1.2	0
1 -> 2	.34	.24	.16	.13	.11	1	0
3-level, V	.41	.31	.20	.16	.11	-	-

P: caliber 4

Save Setup Cost by Switching Coarse Variables

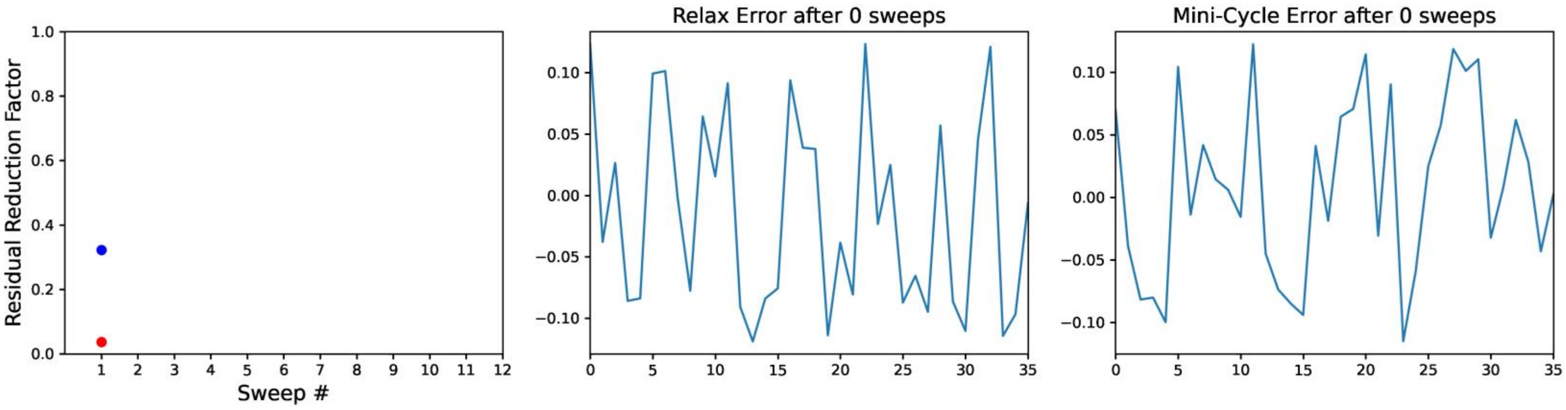
- Smooth TFs more efficiently computed by mini-cycles with to obtain high-order P.
  - R requires only little smoothing.
  - Setup work dominated by SVD, though.

- Temporarily work with expensive, accurate operator for 2-level bootstrapping ($A^c = P^T AP$).

AMG+ uses R!
For the first time, R is not just a prediction tool (mock cycle), but used in bootstrap & solver.

- Post-processing:
  - Sparsify: $A^c = RAP$.
  - Symmetrize $A^c = QAP$.

Mini-Cycle Shrinkage in Action



Kaczmarz

$\bar{\mu} = .58$
$\bar{\nu} = 6$

2-level Cycle(2,2)

$\bar{\mu} = .04$
$\bar{\mu}^{1/W} = .38$
$\bar{\nu} = 1$

Mini-cycle shrinks
more efficiently than relaxation,
even if asymptotically slow.

Using sampling (shift invariance)
gives convergent adaptive setup.

7. Coarse-level Construction is Local,

which is especially useful for repetitive problems & upscaling.

Coarse-level Construction is Local (**In Principle)

- Relaxation is local, so producing TFs is local.
- In repetitive problems, sample across the domain.
- Mock cycle rate well estimated on domain size = $4 \times \text{aggregate\_size}$.
- Two-level cycle rate well estimated on domain size = $4 \times \text{aggregate\_size}$.
  - Only shrinkage is important, not asymptotic rate.

Recap of AMG+ Principles

1. Relaxation should have good shrinkage.
2. Test Functions (TFs) are used to construct coarsening & interpolation and.
3. Coarse-level variables are obtained by local TF SVD and tested by the mock cycle.
4. Quantitative quality prediction tools, e.g., the mock cycle, guide the separate design of each multigrid component.
5. Interpolation is constructed by least-squares fitting of TF values, but ultimately tested via 2-level cycle shrinkage factor.
6. TFs & cycle iteratively improve each other via bootstrap.
7. Coarse-level construction is a local process and with shift invariance the process converges.

Thanks!

Questions/Ideas for Applications?

Email us at

achibr@gmail.com

jjb23@psu.edu

oren.livne@gmail.com

kkahl@uni-wuppertal.de

Challenging Problems for AMG

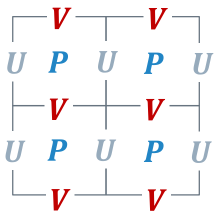
- **Nearly Singular**

  $[\Delta - \varepsilon(x)^2]u(x) = f(x)\,, \quad |\varepsilon(x)| \ll \frac{1}{L}$

- **Systems**

  Species are not explicitly identified

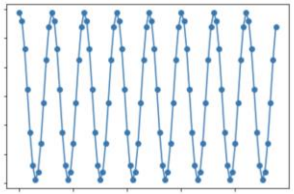
  Stokes



- **Highly indefinite**

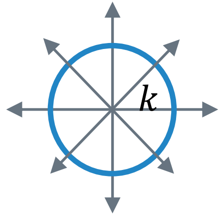
  $[\Delta + k^2]u = f$

  $k \gg \frac{1}{L}$



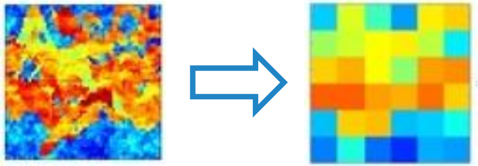
- **Emerging types of coarse variables**

  Multiple types: directional rays



Challenging Problems for AMG (Cont.)

- Upscaling
  - Deriving equations at increasingly coarser scale
  - Creating interpretable coarse-level variables



- Non-linear systems
  - where a coarse version is not given
- Inverse Problems
- Non-local equations

$$\int g(x,y)u(y)dy = f(x,u)$$

- Stochastic Optimization
- No geometric locations
- No locality graph

Neural networks

Goals

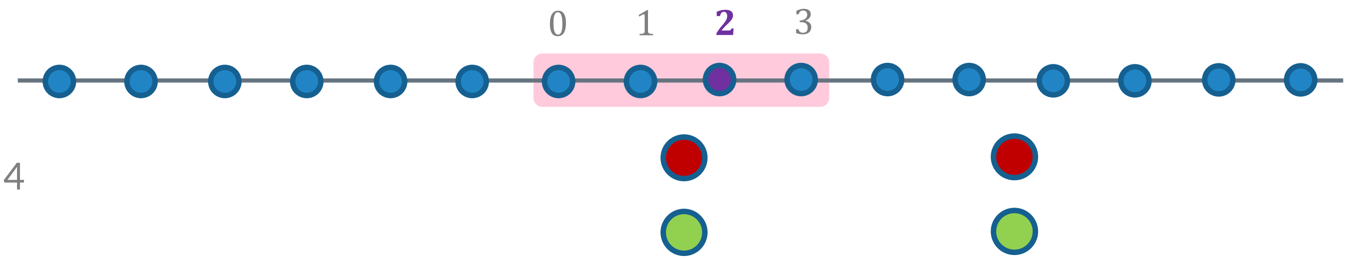
SOLVE THE 1D HELMHOLTZ EQUATION

- Key challenge: automatically derive ray coarse variables from wave fine variables.
- Don't exploit particulars of Helmholtz or 1D.
- Factor out other, unrelated difficulties.

DEVELOP GENERAL MULTILEVEL PRINCIPLES

that can apply to a wide variety of problems.

Ridge Least-Squares Fitting to TFs



Caliber = 4

$P = R^T$

$$\min_{p_i} \sum_s W_{is} \left(x_{is} - \sum_j p_{ij} x_{js}^c \right)^2 + \alpha \sum_s W_{is} \, (x_{is})^2 \qquad i = 0..a-1; \quad x^c = Rx.$$

- Weighting $W_{is} = ||r_{is}||^{-2}$, local norm; unimportant for comparable TFs.
- α determined by minimizing interpolation error on validation samples. (Use SVD!)
- $P_{a\times ?}$ tiled from aggregate to domain; stride = 2 is possible, but 4 is easier.