

# Algebraic Multigrid algorithm for the indefinite Helmholtz equations

Ira Livshits

Ball State University

Fluid Flow Simulations: from Particles to PDE's

Delft University of Technology

November 10, 2011

# Outline

- 1 Multigrid methods
- 2 Helmholtz equation and MG
- 3 Bootstrap Algebraic Multigrid
- 4 Setup/solver
- 5 Numerics
- 6 Conclusions

# Iterative solvers

Consider a system of linear equations

$$Ax = b$$

Iterative solvers are generally of the form

$$x^{m+1} = BAx^m + Bb$$

and for the error  $e^{m+1} = u - u^{m+1}$

$$e^{m+1} = (I - BA)e^m$$

Convergence:  $\|e^{m+1}\| < \|e^m\|$

As a convergence factor often considered is  $\rho(I - BA)$ .

## More on convergence

Consider eigen pairs of  $A$ :  $(\lambda_j, v_j)$ :  $Av_j = \lambda_j v_j$ ,  $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ .

## More on convergence

Consider eigen pairs of  $A$ :  $(\lambda_j, v_j)$ :  $Av_j = \lambda_j v_j$ ,  $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ .

For errors representation  $e^{m+1} = \sum \alpha_j^{m+1} v_j$  and  $e^m = \sum \alpha_j^m v_j$ ,

## More on convergence

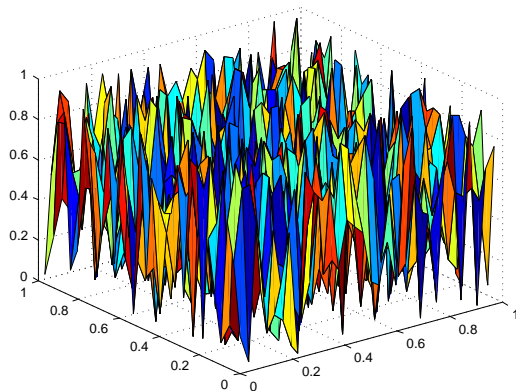
Consider eigen pairs of  $A$ :  $(\lambda_j, v_j)$ :  $Av_j = \lambda_j v_j$ ,  $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ .

For errors representation  $e^{m+1} = \sum \alpha_j^{m+1} v_j$  and  $e^m = \sum \alpha_j^m v_j$ ,

*Overall convergence* is defined by  $\max_j \left| \frac{\alpha_j^{m+1}}{\alpha_j^m} \right|$

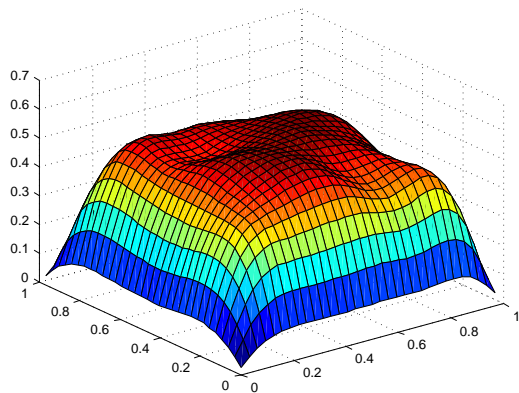
# Relaxation of $\Delta u = 0$

Random initial guess:



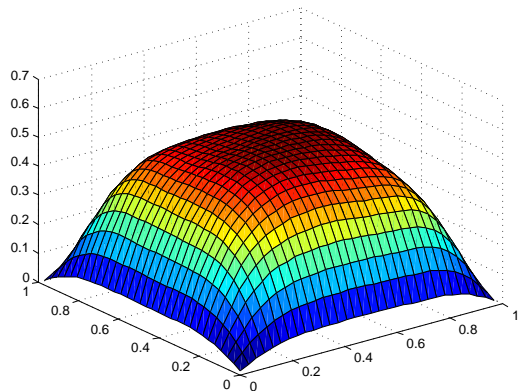
# Relaxation of the discretized $\Delta u = 0$

Error after 10 Gauss-Seidel relaxations:



# Relaxation of $\Delta u = 0$

Error after 10 more Gauss-Seidel relaxations:



# Convergence factor vs smoothing factor

- *Convergence factor* (GS, Jacobi, wJacobi, SOR) is

$$\rho = 1 - O(h^2)$$

- Jacobi:  $x_J^{m+1} = (I - D^{-1}A)x_J^m + D^{-1}b$
- Gauss Seidel:  $x_{GS}^{m+1} = (I - (L + D)^{-1}A)x_J^m + (L + D)^{-1}b$

# Convergence factor vs smoothing factor

- *Convergence factor* (GS, Jacobi, wJacobi, SOR) is  $\rho = 1 - O(h^2)$ 
  - Jacobi:  $x_J^{m+1} = (I - D^{-1}A)x_J^m + D^{-1}b$
  - Gauss Seidel:  $x_{GS}^{m+1} = (I - (L + D)^{-1}A)x_J^m + (L + D)^{-1}b$
  
- The *smoothing factor* is a convergence factor for  $\alpha_{n/2}, \dots, \alpha_n$ , i.e., for the high energy eigenfunctions  $v_{n/2}, \dots, v_n$

# Convergence factor vs smoothing factor

- *Convergence factor* (GS, Jacobi, wJacobi, SOR) is  $\rho = 1 - O(h^2)$ 
  - Jacobi:  $x_j^{m+1} = (I - D^{-1}A)x_j^m + D^{-1}b$
  - Gauss Seidel:  $x_{GS}^{m+1} = (I - (L + D)^{-1}A)x_j^m + (L + D)^{-1}b$
  
- The *smoothing factor* is a convergence factor for  $\alpha_{n/2}, \dots, \alpha_n$ , i.e., for the high energy eigenfunctions  $v_{n/2}, \dots, v_n$

It is mesh-independent. (GS for 2D Laplace has 0.25.)

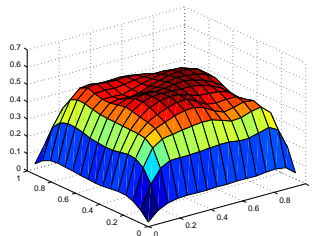
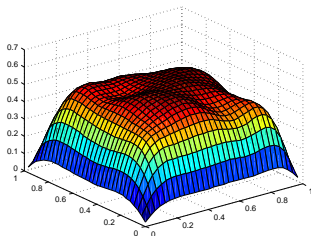
## Convergence factor vs smoothing factor

- *Convergence factor* (GS, Jacobi, wJacobi, SOR) is  $\rho = 1 - O(h^2)$ 
  - Jacobi:  $x_j^{m+1} = (I - D^{-1}A)x_j^m + D^{-1}b$
  - Gauss Seidel:  $x_{GS}^{m+1} = (I - (L + D)^{-1}A)x_j^m + (L + D)^{-1}b$
- The *smoothing factor* is a convergence factor for  $\alpha_{n/2}, \dots, \alpha_n$ , i.e., for the high energy eigenfunctions  $v_{n/2}, \dots, v_n$

It is mesh-independent. (GS for 2D Laplace has 0.25.)

- Remaining non reduced low energy eigenfunctions  $v_1, \dots, v_{n/2-1}$  are smooth.

# Motivation to use coarse grids



The remaining after relaxation, aka smoothing, error has a good coarse grid representation.

## Reason for using coarse grids

Coarse grids can be used to compute an improved initial guess for the fine-grid relaxation. This is advantageous because:

- Relaxation on the coarse-grid is much cheaper ( $1/2$  as many points in  $1D$ ,  $1/4$  in  $2D$ ,  $1/8$  in  $3D$ );
- Relaxation on the coarse grid has a marginally better convergence rate, for example  $1 - O(4h^2)$  instead of  $1 - O(h^2)$ ;
- Smooth eigenfunctions on the finest grid become more oscillatory on the coarse grid.

# Multigrid components

- Relaxation (smoother): efficiently dumps high-energy (eigenfunctions with large eigenvalues) error components; leaves unchanged low eigenmodes;
- A coarse-grid system operator: accurately resolve the low eigenmodes;
- Restriction operator (averaging): transfers fine-grid information (residual) to coarse grid;
- Prolongation operator (interpolation): transfers coarse-grid correction back to the fine grid; has to be accurate for low eigenmodes.

## Nice example: $\Delta u = f$

- Relaxation: Gauss-Seidel.  $V(1, 1)$  cycle reduces residual for factor  $0.25^2$ ;
- A coarse-grid operator: discretization of the differential Laplace operator on coarse scale ( $2h, 4h, \dots$ )
- Restriction operator: full weighting of the residual (annihilates remaining oscillatory components), transferring only smooth ones to coarser grids;
- Interpolation: polynomial (linear) which is accurate for smooth components.

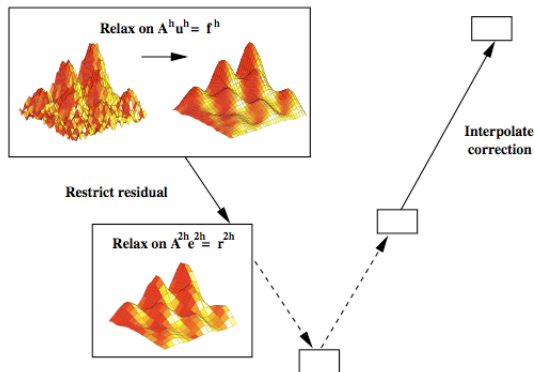
## Nice example: $\Delta u = f$

- Relaxation: Gauss-Seidel.  $V(1, 1)$  cycle reduces residual for factor  $0.25^2$ ;
- A coarse-grid operator: discretization of the differential Laplace operator on coarse scale ( $2h, 4h, \dots$ )
- Restriction operator: full weighting of the residual (annihilates remaining oscillatory components), transferring only smooth ones to coarser grids;
- Interpolation: polynomial (linear) which is accurate for smooth components.

Works great because the lowest eigenmodes are physically smooth.

# One picture is worth a thousand words

## Geometric Multigrid (MG)



# Helmholtz equation

$$Lu = -\Delta u(x) - k^2 u(x), \quad x \in \Omega \subset \mathcal{R}^d$$

with large wave numbers  $k$ :  $2\pi/k \ll \dim(\Omega)$

# A Comment on Terminology

The *indefinite* Helmholtz equation is

$$Lu = (-\Delta - k^2)u = f,$$

in contrast to

$$(\Delta - \eta)u = f, \eta > 0$$

often also called Helmholtz equation, or Helmholtz equation with the good sign, or *positive definite* Helmholtz equation.

The subject of this talk is the indefinite Helmholtz equation.

## Discretization error

Discrete wave number vs exact wave number:

$$|k^h| \approx k \left( 1 + \frac{k^2 h^2}{\gamma} \right), \quad 24 \leq \gamma \leq 48.$$

The total (accumulated) *phase* error is

$$E(kd, kh) = kd \frac{k^2 h^2}{2\pi\gamma}.$$

For the discrete solutions to be accurate approximation to the differential solutions need

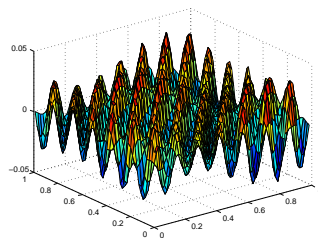
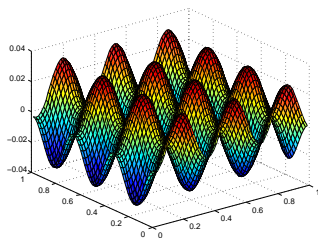
$$E(kd, kh) \ll 1.$$

For fixed  $k, d$  this means small  $h$  and large problem size,  $n$ .

## Obvious bottlenecks for multigrid

- Dominant phase discretization error;
- Quality of standard coarse-grid and prolongation operators deteriorates on coarse grids;
- Poor multigrid relaxation - standard "fast" relaxation scheme diverge;

# Lowest eigenmodes



Lowest eigenmodes for  $k = 6\pi$  and  $k = 18\pi$ .

## More bottlenecks for multigrid

- Oscillatory low eigenmodes;
- There are many of them;
- They are different from each other;
- On sufficiently coarse grids, there is no single prolongation or coarse grid operator of a reasonable sparsity that works for all such eigenmodes.

## Flashback ...Wave-ray algorithm (c. 1997)

The zero eigenmodes of the Helmholtz differential operator:

$$v_j = \exp(i\mathbf{k}_j\mathbf{x}), \quad |\mathbf{k}_j| = k$$

The unreduced error

$$e = \sum c_j \exp(i\mathbf{w}_j\mathbf{x}), \quad (1 - \alpha)k \leq |\mathbf{w}_j| \leq (1 + \beta)k, \quad \alpha, \beta \approx 0.3$$

and can be represented in terms of *basis functions* as

$$e = \sum_q e_q(x) \exp(i\mathbf{k}_q\mathbf{x}), \quad \mathbf{k}_q = k(\cos \theta_q, \sin \theta_q), \quad \theta_q = q2\pi/Q$$

## Flashback ... cont'd

Both error and residual are of the form

$$e = \sum_q e_q(x) \exp(i\mathbf{k}_q \mathbf{x}) \quad \text{and} \quad r = \sum_q r_q(x) \exp(i\mathbf{k}_q \mathbf{x}),$$

where  $e_q, r_q$  are smooth (enough to be approximated on the scale  $kH \gg 1$ ).

For example: for  $Q = 8$  the scale is  $\leq KH \leq 4$ .

*Idea*: Instead of solving  $Ae = r$  on fine grid,  
solve  $A_q e_q = r_q$  on coarse grids ( $Q = 8$ )

# Technicalities

Differential operators for  $A_q$ :

$$L(e(x)\exp(i\mathbf{k}_q\mathbf{x})) = \exp(i\mathbf{k}_q\mathbf{x})L_q e_q(x)$$

Residual separation:

$$W(\exp(-i\mathbf{k}_q\mathbf{x})r) = W(r_q + \sum r_j \exp(i\mathbf{k}_j - \mathbf{k}_q\mathbf{x})) \approx r_q,$$

where  $W$  is a full weighting

(It annihilates oscillatory components and preserves smooth ones)

Solve separately  $A_q e_q = r_q$  and reconstruct  $e = \sum e_q \exp(i\mathbf{k}_q\mathbf{x})$

# Wave-ray algorithm

(1997, 2006), Brandt and Livshits

- Linear dependence of costs on the problem size: cost  $O(n)$ ;
- Convergence independent on both  $k$  and  $kh$  - fully scalable;
- Combines standard multigrid and special near null space treatment;

# Back to multigrid: What makes it work

- (1) Each coarse-grid operator(s) approximates all error components not reduced by relaxation on previous finer grids;
- (2) Prolongation operator(s) accurately transfers such components from the coarse grid to the next finer grid.
- (3) In Algebraic multigrid (2) yields (1) when using Galerkin method:  $A_c = P^t A_f P$ .

## Least Squares prolongation operator $P$

Find  $P$  that best fits a given set of f.g. functions  $u_1, \dots, u_k$ .

We choose  $P$  that minimizes

$$\mathcal{LS}(P) = \sum \omega_m \|u_m - PU_m\|^2$$

where  $U_m$  are c.g. representatives of  $u_m$ ., e.g.  $U_m = Ju_m$ ;  $\omega_m > 0$

Here  $u_m$  is a set of functions with  $Lu_m \approx 0$ , analytically known or computed numerically using standard multigrid or bootstrap.

## Least Squares prolongation operator $P$

Find  $P$  that best fits a given set of f.g. functions  $u_1, \dots, u_k$ .

We choose  $P$  that minimizes

$$\mathcal{LS}(P) = \sum \omega_m \|u_m - PU_m\|^2$$

where  $U_m$  are c.g. representatives of  $u_m$ ., e.g.  $U_m = Ju_m$ ;  $\omega_m > 0$

Here  $u_m$  is a set of functions with  $Lu_m \approx 0$ , analytically known or computed numerically using standard multigrid or bootstrap.

*Brandt, Brannick, Kahl, Livshits, Bootstrap AMG: Least Squares Interpolation and Multigrid Eigen-Approximations*

*Bolten, Brandt, Brannick, Fromer, Kahl, Livshits: A Bootstrap Algebraic Multilevel Method for Markov Chains*

*Brandt, Brannick, Kahl, Livshits, An Algebraic Distances Measure of AMG Strength of Connections*

## Multigrid error

Components that must be accurately represented on all grids are the ones with small relative residuals, lowest eigenmodes of  $A$ :

$$Av_j = \lambda_j v_j, \quad \lambda_j \approx 0$$

Error not reduced by relaxation (on each grid) is a combination of eigenfunctions

$$e = \sum \alpha_j v_j$$

The corresponding residual  $r = Ae$  can be also represented

$$r = \sum \beta_j v_j$$

Helmholtz:  $v_j = \cos(\mathbf{w}_j \mathbf{x})$  and  $\sin(\mathbf{w}_j \mathbf{x})$ ,  $|\mathbf{w}_j| \approx k$ , subject to b.c.

## Two scenarios

- (1) There is a single  $P$  such that all  $v_j \in \mathcal{R}(P)$ . Then there is a coarse grid function  $E$  such that  $e \approx PE$  which can be approximated by solving  $P^t A P E = P^t r$ , where  $r = Ae$ .

## Two scenarios

- (1) There is a single  $P$  such that all  $v_j \in \mathcal{R}(P)$ . Then there is a coarse grid function  $E$  such that  $e \approx PE$  which can be approximated by solving  $P^t A P E = P^t r$ , where  $r = Ae$ .
- (2) There is a sequence of  $P_1, \dots, P_Q$ ,  $Q = O(1)$  such that

## Two scenarios

- (1) There is a single  $P$  such that all  $v_j \in \mathcal{R}(P)$ . Then there is a coarse grid function  $E$  such that  $e \approx PE$  which can be approximated by solving  $P^t A P E = P^t r$ , where  $r = Ae$ .
  
- (2) There is a sequence of  $P_1, \dots, P_Q$ ,  $Q = O(1)$  such that
  - (a) Each  $v_j \in \cup \mathcal{R}(P_q)$ :  $e \approx \sum P_q E_q$  for some c.g. functions  $E_q$ ;

## Two scenarios

- (1) There is a single  $P$  such that all  $v_j \in \mathcal{R}(P)$ . Then there is a coarse grid function  $E$  such that  $e \approx PE$  which can be approximated by solving  $P^t A P E = P^t r$ , where  $r = A e$ .
- (2) There is a sequence of  $P_1, \dots, P_Q$ ,  $Q = O(1)$  such that
  - (a) Each  $v_j \in \cup \mathcal{R}(P_q)$ :  $e \approx \sum P_q E_q$  for some c.g. functions  $E_q$ ;
  - (b)  $\mathcal{R}(P_p) \cap \mathcal{R}(P_q) \approx \emptyset, p \neq q$ .

## Two scenarios

- (1) There is a single  $P$  such that all  $v_j \in \mathcal{R}(P)$ . Then there is a coarse grid function  $E$  such that  $e \approx PE$  which can be approximated by solving  $P^t A P E = P^t r$ , where  $r = A e$ .
- (2) There is a sequence of  $P_1, \dots, P_Q$ ,  $Q = O(1)$  such that
  - (a) Each  $v_j \in \cup \mathcal{R}(P_q)$ :  $e \approx \sum P_q E_q$  for some c.g. functions  $E_q$ ;
  - (b)  $\mathcal{R}(P_p) \cap \mathcal{R}(P_q) \approx \emptyset$ ,  $p \neq q$ .
  - (c) Helpful if  $v_j \in \mathcal{R}(P_{q_j})$  but then  $Q$  is large

## Second scenario: multiple c.g. representations (cont.)

Consider error  $e$  in the form  $e = \sum e_q$  such that each  $e_q \approx P_q E_q$

Prolongations  $P_q$  act from very coarse grids, functions  $E_q$  are considered there and they are smooth.

*Idea :*

- Approximate smooth  $E_q$  instead of oscillatory  $e$ ;
- Reconstruct  $e \approx \sum P_q E_q$ .

## Building a sequence $\{P_q\}$

- (a) If *closeness* in the test set is known (analytically) regroup into  $\{v_{j_1}^q, \dots, v_{j_m}^q\}$  and build LS  $P_q$ .

## Building a sequence $\{P_q\}$

- (a) If *closeness* in the test set is known (analytically) regroup into  $\{v_{j_1}^q, \dots, v_{j_m}^q\}$  and build LS  $P_q$ .
- (b) If  $v_j$  are computed numerically:

## Building a sequence $\{P_q\}$

- (a) If *closeness* in the test set is known (analytically) regroup into  $\{v_{j_1}^q, \dots, v_{j_m}^q\}$  and build LS  $P_q$ .
- (b) If  $v_j$  are computed numerically:
- (1) Choose a function  $v_{j_0}^q$  and build  $P_q$  such that  $v_{j_0}^q = P_q V_{j_0}^q$  by putting the largest weight in LS formulation;
  - (2) Orthogonalize  $v_j$  with respect to  $P_q$ :  $v_j = v_j - P_q V_j^q$ ;
  - (3) If  $\max \|v_j\| < \epsilon$  stop, else build a new  $P_{q+1}$ .

## Residual separation

Residual  $r = Ae$  has a similar to  $e$  form

$$r = \sum r_q \approx \sum P_q R_q$$

If (b) holds then  $P_p^t r = \sum_q P_p^t P_q R_q \approx R_p$

*Very important as otherwise there is a redundant correction as*

$$r \neq \sum P_q P_q^t r$$

## Coarse grid systems

Expansion  $e = \sum P_q E_q$  leads to  $Ae = \sum AP_q E_q$ .

Multiplication by  $P_p^t$  yields a system of equations

$$CE = R$$

where  $C_{pq} = P_p^t AP_q$ ,  $E = [E_1, \dots, E_Q]^t$ ,  $R = [P_1^t r, \dots, P_Q^t r]^t$ .

## Coarse grid systems

Expansion  $e = \sum P_q E_q$  leads to  $Ae = \sum AP_q E_q$ .

Multiplication by  $P_p^t$  yields a system of equations

$$CE = R$$

where  $C_{pq} = P_p^t AP_q$ ,  $E = [E_1, \dots, E_Q]^t$ ,  $R = [P_1^t r, \dots, P_Q^t r]^t$ .

If each  $v_j \in R(P_{q_j})$  then  $P_p^t AP_q \approx 0$  if  $p \neq q$   
 then solving  $CE = R$  is equivalent to solving systems

$$C_{q,q} e_q = r_q, \quad q = 1, \dots, Q,$$

## MG cycle - Correction Scheme

- On fine grids use single representation until not accurate;
- Compute the residual  $R_q$  and separate it into multiple residuals on coarse grids,  $r_q$ ;
- There, use multiple representations to compute corrections,  $E_q$
- Reconstruct to the single function,  $e$
- Use the single operators on the way to the finest grid

## Multiple representation – one grid example

*Input:*  $\{C, P_q\}, r^{\ell_s}$

*Output:* Updated  $e^{\ell_s}$

Separate  $Q$  residuals  $r_q = (P_q)^t r^{\ell_s}$

**for**  $q = 1 : Q$  **do**

    Approximate  $(CE)_q = r_q$

**end for**

Make  $Q$  updates  $e^{\ell_s} = e^{\ell_s} + P_q e_q$

# MG structure - 1D

- Single representation:  $kh \approx \pi/4$  and finer;
- Double representation:  $kh \approx \pi/2$ .
- Single representation for each of the the two:  $kh \approx \pi$  and coarser.

# Model problem

$$-u_{xx} - k^2(x)u = f, \quad x \in [0, 1]$$

Periodic boundary conditions

Coarsest grid : 4 unknowns

Smoothers:

Gauss-Seidel (1-1) on fine grids

Kaczmarz (4-4) on coarse grids

Linear prolongation pattern for all prolongation operators

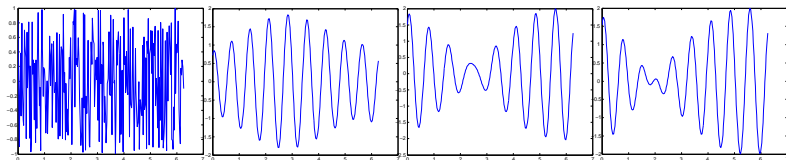
## Using analytical test functions

The initial test set  $\{\cos(kx), \sin(kx), \sin((k+1)x), \cos((k+1)x)\}$

| $n$      | 128  | 256  | 512  | 1024 | 2048 |
|----------|------|------|------|------|------|
| $k = 10$ | 0.17 | 0.09 | 0.09 | 0.09 | 0.09 |
| $k = 30$ | *    | *    | 0.16 | 0.11 | 0.11 |
| $k = 60$ | *    | *    | *    | 0.16 | 0.11 |

**Table:** The number of levels,  $L$ , ranges from six ( $n = 256$ ) to ten ( $n = 2048$ ).

# Numerical test vectors



**Figure:** First is an example of a random initial vector, uniformly distributed in  $[-1, 1]$ ; the rest are examples of TVs obtained by applying three multigrid V-cycles to the homogenous Helmholtz equation

# Using numerical test functions

Initial test functions are random - subjected to standard multigrid

| $n$      | 128  | 256  | 512  | 1024 | 2048 |
|----------|------|------|------|------|------|
| $k = 10$ | 0.16 | 0.16 | 0.14 | 0.15 | 0.16 |
| $k = 30$ | 0.16 | 0.16 | 0.17 | 0.18 | 0.18 |
| $k = 60$ | 0.15 | 0.17 | 0.18 | 0.18 | 0.18 |

Variable  $k(x) = k(1 + 0.2 \cos 20x)$

|          |      |      |      |      |      |
|----------|------|------|------|------|------|
| $n$      | 128  | 256  | 512  | 1024 | 2048 |
| $k = 20$ | 0.16 | 0.12 | 0.08 | 0.09 | 0.09 |
| $k = 40$ | 0.14 | 0.09 | 0.09 | 0.09 | 0.09 |

Discontinuous  $k(x)$ 

$$k(x) = \begin{cases} 0.8k, & x \in [7\pi/8, 9\pi/8], \\ k, & \text{otherwise.} \end{cases}$$

| $n$      | 128  | 256  | 512  | 1024       | 2048       |
|----------|------|------|------|------------|------------|
| $k = 20$ | 0.16 | 0.18 | 0.16 | 0.21(0.18) | 0.24(0.18) |
| $k = 40$ | 0.17 | 0.18 | 0.20 | 0.23(0.20) | 0.25(0.20) |

**Table:** The results with local relaxation are given in parenthesis for  $n = 1024, 2048$

# MG structure - 2D

- (a) Single representation:  $kh \approx \pi/4$  and finer;  
(Eight test vectors in LS, bilinear pattern)
- (b)  $Q = 6$  representations:  $kh \approx \pi$ ;
- (c) If this is not coarse enough - aggressive coarsening with increase in the number of representations (work in progress).

# Model problem

$$-\Delta u - k^2 u = f, \quad x \in [0, 1]^2$$

Coarsest grid :  $4^2$  unknowns

Smoothers:

Gauss-Seidel (1-1) on fine grids

Kaczmarz (4-4) on coarse grids

Pattern of LS prolongations is bilinear.

# Test set - approximation of the lowest eigenmodes

- Random eight vectors;  $V(4, 4)$  (Kaczmarz)
- Two V-cycles;
- The resulting test vectors (after Rayleigh Ritz) have (the lowest) eigenvalues within  $|\lambda_h - \lambda| \leq 0.2\lambda$
- $Q = 6$  prolongation operators;
- Preconditioned GMRES

Brandt, Livshits, Wave-ray algorithm for standing wave equation, 1997

# Numerical results

| $k$ | $64^2$ | $128^2$ | $256^2$ |
|-----|--------|---------|---------|
| 8   | 12     | 13      | 14      |
| 16  | 11     | 12      | 12      |
| 32  | 11     | 11      | 13      |

Table: Number of preconditioned GMRES

# Conclusions

- Works nicely for 1d and promising for 2d;
- Adding multiple representation barely affects costs; it takes place on coarse grids;
- Next step is optimal coarsening (compatible relaxation and algebraic distances);
- The goal is to be able to deal with different types of  $k$  as in 1d.

# Invitation

12<sup>th</sup> Copper Mountain Conference on Iterative Methods

March 25-March 30, 2012

Student Competition Papers: January 11, 2012

Author Abstracts: January 18, 2012