Multigrid methods

Algebraic Multigrid methods

Algebraic Multilevel Iteration methods

TDB – NLA

Residual correction

$A\mathbf{x} = \mathbf{b}, \mathbf{x}_{exact},$ $\quad \mathbf{e}^{(k)} = \mathbf{x}_{exact} - \mathbf{x}^{(k)}$

$\mathbf{r}^{(k)} = \mathbf{b} - A\mathbf{x}^{(k)}$

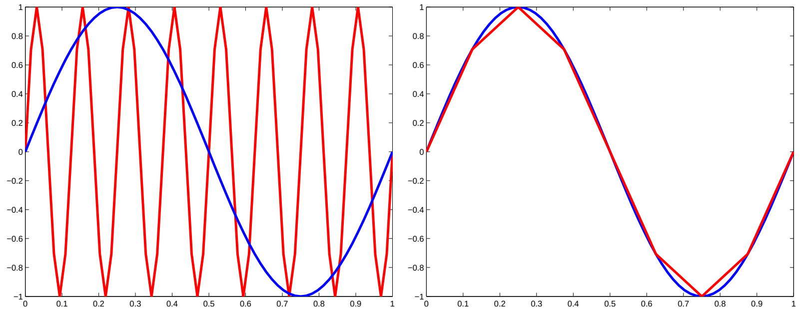
Residual equation:	$A\mathbf{e}^{(k)} = \mathbf{r}^{(k)}$
Residual correction:	$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{e}^{(k)}$
Recall:	$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + C^{-1}(\mathbf{b} - A\mathbf{x}^{(k)})$
Error propagation:	$\mathbf{e}^{(k+1)} = (I - C^{-1}A)\mathbf{e}^{(k)}$

TDB – NLA

Run Jacobi demo...

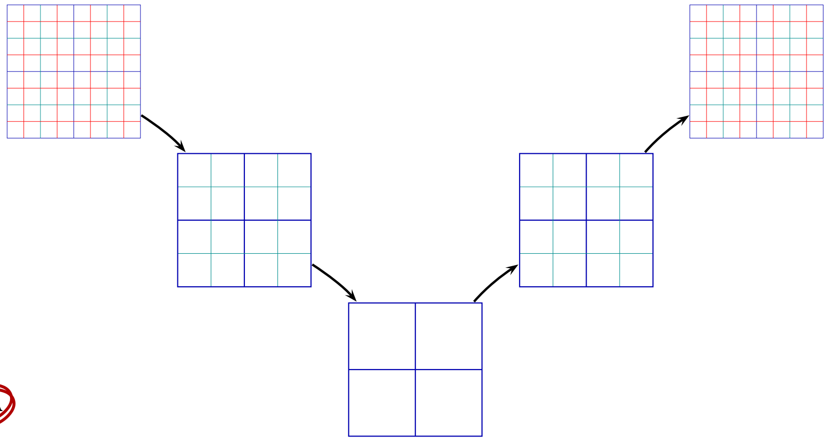
`student/NLA/Demos/Module3/L5`

High and low frequencies - nonsmooth, smooth



Main idea: R. Fedorenko (1961), N.S. Bakhvalov (1966)

Reduce the error $\mathbf{e}^{(k)} = \mathbf{x}_{exact} - \mathbf{x}^{(k)}$ on the given (fine) grid by successive residual corrections on a hierarchy of (nested) coarser grids.



TDB – NLA

Some numbers/contributors:

Years	MG	AMG	Years	MG	AMG
1966-1986	3420	873	2007-2017	21700	16800
1987-1996	15400	5370	2018-	4610	2360
1997-2006	22000	12800			

Archi Brandt	Jan Mandel	Tom Manteiffel
Wolggang Hackbusch	Steve McCormick	Yvan Notay
Jurgen Ruge	Petr Vanec	Irad Yavneh
Klaus Stüben	Piet Hemker	Panayot Vassilevski

.....

Ruge, J. W.; Stüben, K. Algebraic multigrid. Multigrid methods, 73-130, Frontiers Appl. Math., 3, SIAM, Philadelphia, PA, 1987.

TDB – NLA

Borrowed from Yvan Notay

Algebraic multigrid and multilevel methods

https://perso.uclouvain.be/paul.vandooren/Notay.pdf

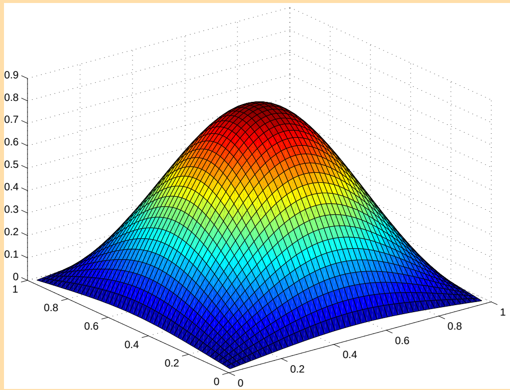
TDB – NLA

An example

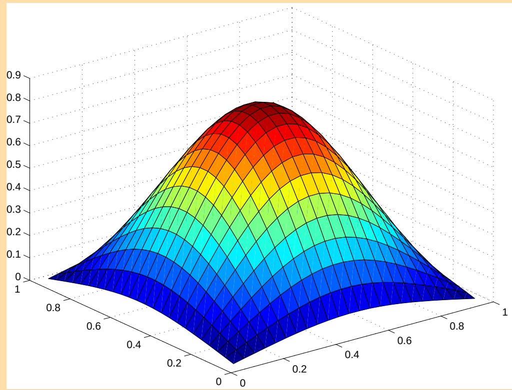
ULB

PDE: $$\begin{aligned} -\Delta u &= 20\, e^{-10\,((x-0.5)^2+(y-0.5)^2)} \quad \text{in } \Omega = (0,1)\times(0,1) \\ u &= 0 \quad \text{on } \partial\Omega \end{aligned}$$

Uniform grid with mesh size h, five-point finite difference.



Solution with $h^{-1} = 50$



Solution with $h^{-1} = 25$

An idea

ULB

Fine grid (system to solve):

$$A\,\mathbf{u} \;=\; \mathbf{b}\,.$$

Coarse grid (auxiliary system):

$$A_C\,\mathbf{u}_C \;=\; \mathbf{b}_C\,.$$

$\mathbf{u}_C$ may be computed and prolongated (by interpolation) on the fine grid:

$$\mathbf{u}^{(1)} = p\,\mathbf{u}_C$$

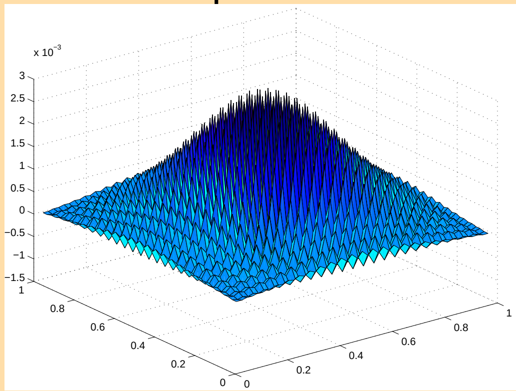
$\mathbf{u}^{(1)}$ may serve as initial approximation, i.e., one solves

$$A\,(\mathbf{u}^{(1)} + \mathbf{x}) \;=\; \mathbf{b} \qquad \text{or} \qquad A\,\mathbf{x} \;=\; \mathbf{b} - A\,p\,A_C^{-1}\,\mathbf{b}_C\,.$$

How it works

ULB

Error on the fine grid after interpolation



$$\frac{\|\mathbf{u}-\mathbf{u}^{(1)}\|}{\|\mathbf{u}\|} = 0.0019$$

Let us repeat

ULB

$$A\,(\mathbf{u}^{(1)} + \mathbf{x}) \;=\; \mathbf{b} \qquad \text{or} \qquad A\,\mathbf{x} \;=\; \mathbf{b} - A\,p\,A_C^{-1}\mathbf{b}_C \;=\; \mathbf{r}^{(1)}\;.$$

(1) Restrict on the coarse grid:

$$\mathbf{r}_C \;=\; r\,\mathbf{r}(1)\;.$$

(2) Solve on the coarse grid:

$$\mathbf{x}_C^{(2)} \;=\; A_C^{-1}\,\mathbf{r}_C\;.$$

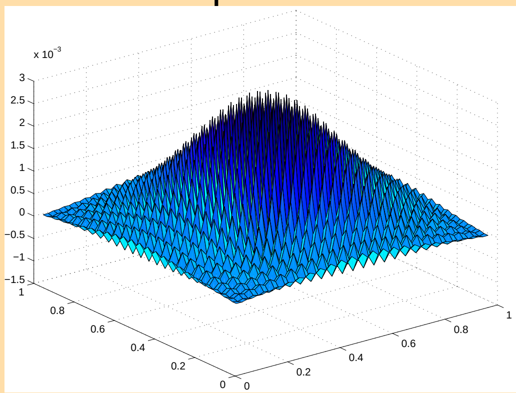
(3) Prolongate:

$$\mathbf{x}^{(2)} \;=\; p\,\mathbf{x}_C^{(2)}\;,$$
$$\mathbf{u}^{(2)} \;=\; \mathbf{u}^{(1)} + \mathbf{x}^{(2)}\;.$$

Still working ?

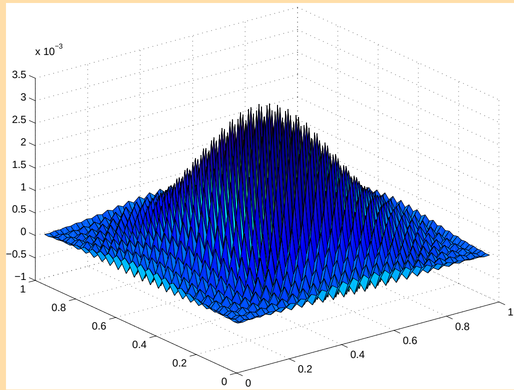
ULB

Error on the fine grid after interpolation



$$\frac{\|\mathbf{u}-\mathbf{u}^{(1)}\|}{\|\mathbf{u}\|} = 0.0019$$

Repeating the process . . .

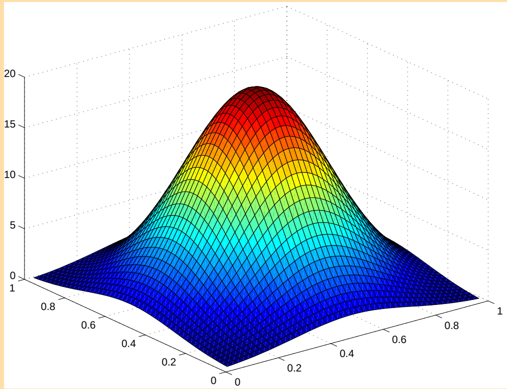


$$\frac{\|\mathbf{u}-\mathbf{u}^{(2)}\|}{\|\mathbf{u}\|} = 0.0018$$

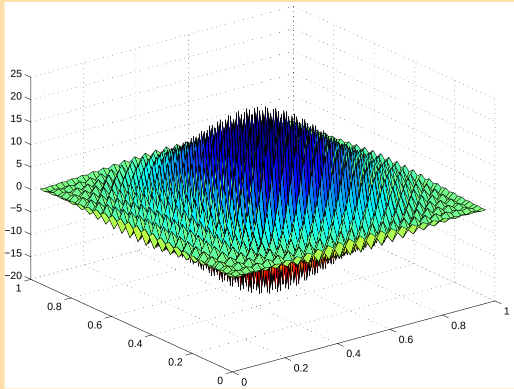
Error controlled through residual

ULB

Initial residual (r.h.s.)



After coarse grid correction



$$\frac{\|\mathbf{b} - A\,p\,A_C^{-1}\mathbf{b}_C\|}{\|\mathbf{b}\|} = 0.7142$$

Explanation

ULB

Assume (for simplicity) that $\mathbf{b}_C = r\,\mathbf{b}$.

One has

$$\begin{aligned} \mathbf{u} - \mathbf{u}^{(1)} &= \mathbf{u} - p\,A_C^{-1}\,r\,\mathbf{b} \\ &= \left(I - p\,A_C^{-1}\,r\,A\right)\,\mathbf{u}\,, \\ \mathbf{u} - \mathbf{u}^{(2)} &= \left(I - p\,A_C^{-1}\,r\,A\right)^2\,\mathbf{u}\,, \end{aligned}$$

etc. Similarly

$$\begin{aligned} \mathbf{r}^{(1)} &= \mathbf{b} - A\,p\,A_C^{-1}\,r\,\mathbf{b} \\ &= \left(I - A\,p\,A_C^{-1}\,r\right)\,\mathbf{r}^{(0)}\,. \end{aligned}$$

$p\,A_C^{-1}\,r$ has rank n_C $\longrightarrow$

$$\rho\left(I - A\,p\,A_C^{-1}\,r\right) = \rho\left(I - p\,A_C^{-1}\,r\,A\right) \geq 1\,.$$

Smoother enters the scene

ULB

$\mathbf{u}-\mathbf{u}^{(1)}$ and $\mathbf{r}^{(1)}$ very oscillatory
$\rightarrow$ improve $\mathbf{u}^{(1)}$ with a simple iterative method, efficient in smoothing the error & residual.

Example: symmetric Gauss-Seidel (SGS)

$$\begin{aligned} L\,\mathbf{u}^{(1+1/2)} &= \mathbf{b}-(A-L)\,\mathbf{u}^{(1)}\,, \qquad (L=\text{low}(A)) \\ U\,\mathbf{u}^{(2)} &= \mathbf{b}-(A-U)\,\mathbf{u}^{(1+1/2)}\,. \qquad (U=\text{upp}(A)) \end{aligned}$$

Same as

$$\mathbf{u}^{(2)} = \mathbf{u}^{(1)}+M^{-1}\mathbf{r}^{(1)}\,, \quad M=L\,D^{-1}\,U \quad (D=\text{diag}(A))$$

Thus:

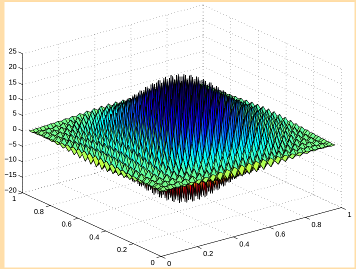
$$\begin{aligned} \mathbf{u}-\mathbf{u}^{(2)} &= (I-M^{-1}A)\,(\mathbf{u}-\mathbf{u}^{(1)}) \\ \mathbf{r}^{(2)} &= (I-A\,M^{-1})\,\mathbf{r}^{(1)} \end{aligned}$$

One may repeat: $\mathbf{r}^{(m+1)} = (I-A\,M^{-1})^m\,\mathbf{r}^{(1)}\,.$

Smoothing effect

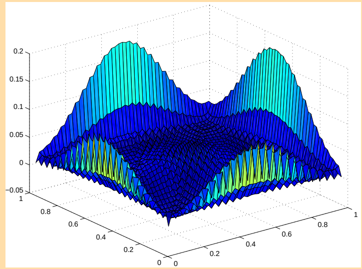
ULB

Residual after CG correction



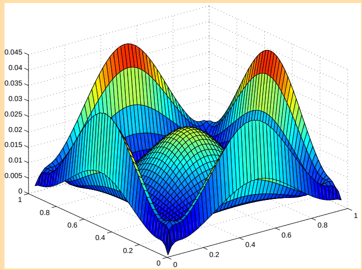
$\frac{\|\mathbf{r}\|}{\|\mathbf{b}\|} = 0.7142$

Adding 1 SGS step



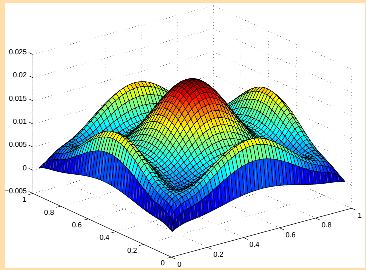
$\frac{\|\mathbf{r}\|}{\|\mathbf{b}\|} = 0.0039$

Adding 3 SGS steps



$\frac{\|\mathbf{r}\|}{\|\mathbf{b}\|} = 0.0018$

Adding 8 SGS steps

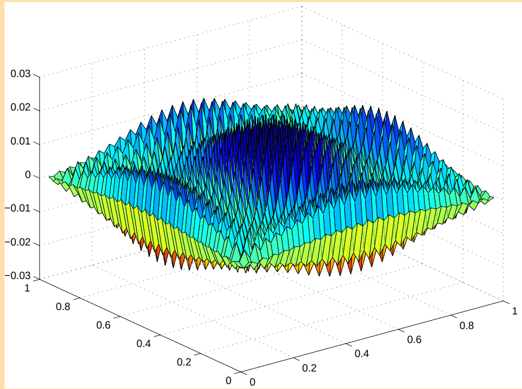


$\frac{\|\mathbf{r}\|}{\|\mathbf{b}\|} = 0.0012$

Smoothing + coarse grid correction

ULB

Adding now a CG correction

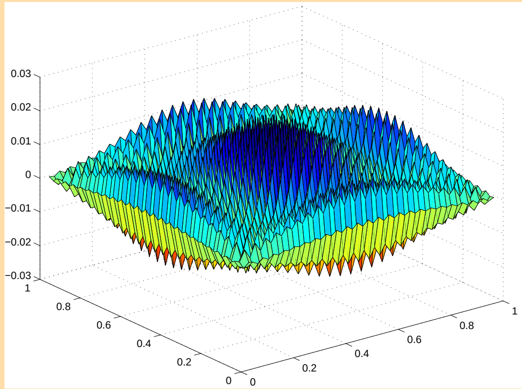


$$\frac{\|\mathbf{r}\|}{\|\mathbf{r}_{\text{previous}}\|} = 0.746$$

Smoothing + coarse grid correction

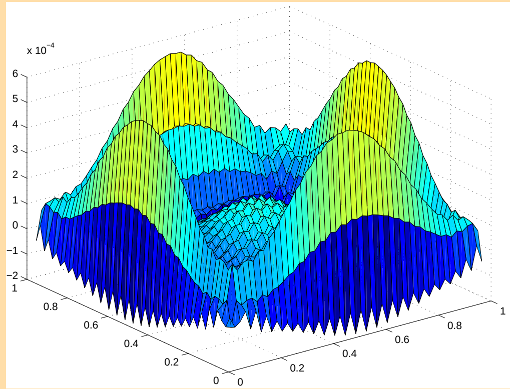
ULB

Adding now a CG correction



$$\frac{\|\mathbf{r}\|}{\|\mathbf{r}_{\text{previous}}\|} = 0.746$$

. . . and again 1 SGS step



$$\frac{\|\mathbf{r}\|}{\|\mathbf{r}_{\text{previous}}\|} = 0.0155$$

What we learned

ULB

For each coarse grid correction:

$$\mathbf{u} - \mathbf{u}^{(m+1)} = \left(I - p\, A_C^{-1}\, r\, A\right) \left(\mathbf{u} - \mathbf{u}^{(m)}\right) .$$

Cannot work alone because $\rho \left(I - p\, A_C^{-1}\, r\, A\right) \geq 1 .$

For each smoothing step

$$\mathbf{u} - \mathbf{u}^{(m+1)} = \left(I - M^{-1}\, A\right) \left(\mathbf{u} - \mathbf{u}^{(m)}\right) .$$

Not efficient alone because $\rho \left(I - M^{-1}\, A\right) \approx 1 .$

However

$$\rho\Big(\left(I - M^{-1}\, A\right) \left(I - p\, A_C^{-1}\, r\, A\right) \left(I - M^{-1}\, A\right) \Big) \ll 1$$

Rmk: if $A = A^T$, we assume $M = M^T$.

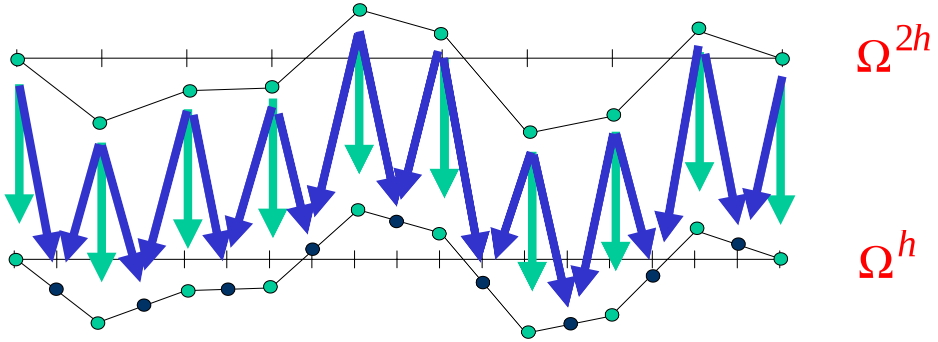
Borrowed from:

- W. Gropp, *A Multigrid Tutorial*
  Presentation by Van Emden Henson, LLNL
  https://www.math.ust.hk/~mawang/teaching/math532/mgtut.pdf
- R. Falgout, *An Algebraic Multigrid Tutorial*, Conference presentation 2010.
  https://mathinstitutes.org/videos/videos/5711

TDB – NLA

1D Interpolation (Prolongation)

- Values at points on the coarse grid map unchanged to the fine grid
- Values at fine-grid points NOT on the coarse grid are the averages of their coarse-grid neighbors



The prolongation operator (1D)

- We may regard I_{2h}^{h} as a linear operator from $\Re^{N/2-1} \longrightarrow \Re^{N-1}$
- e.g., for N=8,

$$\begin{pmatrix} 1/2 & & \\ 1 & & \\ 1/2 & 1/2 & \\ & 1 & \\ & 1/2 & 1/2 \\ & & 1 \\ & & 1/2 \end{pmatrix}_{7x3} \begin{pmatrix} v_1^{2h} \\ v_2^{2h} \\ v_3^{2h} \end{pmatrix}_{3x1} = \begin{pmatrix} v_1^{h} \\ v_2^{h} \\ v_3^{h} \\ v_4^{h} \\ v_5^{h} \\ v_6^{h} \\ v_7^{h} \end{pmatrix}_{7x1}$$

- I_{2h}^{h} has full rank, and thus null space $\{\phi\}$

1D Restriction by injection

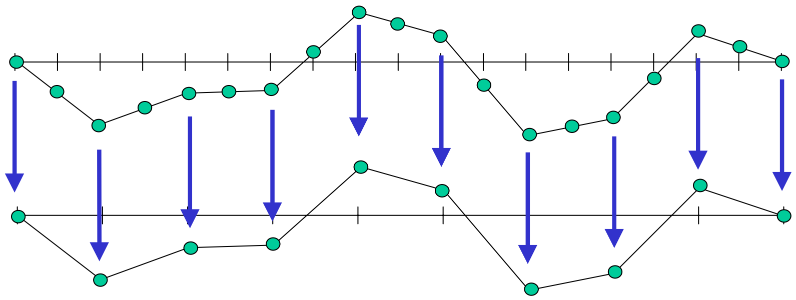
- Mapping from the fine grid to the coarse grid:

$$I_h^{2h} : \Omega^h \rightarrow \Omega^{2h}$$

- Let v^h, v^{2h} be defined on Ω^h, Ω^{2h}. Then

$$I_h^{2h} v^h = v^{2h}$$

where $v_i^{2h} = v_{2i}^h$.



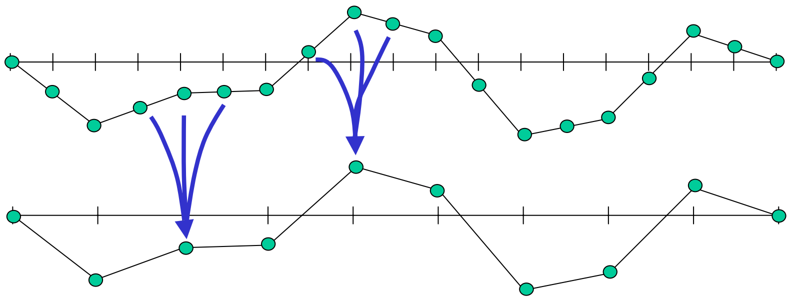
1D Restriction by full-weighting

- Let v^h, v^{2h} be defined on Ω^h, Ω^{2h}. Then

$$I_h^{2h} v^h = v^{2h}$$

where

$$v_i^{2h} = \frac{1}{4}(v_{2i-1}^h + 2 v_{2i}^h + v_{2i+1}^h)$$



The restriction operator R (1D)

- We may regard I_h^{2h} as a linear operator from $\Re^{N-1} \longrightarrow \Re^{N/2-1}$
- e.g., for N=8,

$$\begin{pmatrix} 1/4 & 1/2 & 1/4 & & & & \\ & & 1/4 & 1/2 & 1/4 & & \\ & & & & 1/4 & 1/2 & 1/4 \end{pmatrix} \begin{pmatrix} v_1^h \\ v_2^h \\ v_3^h \\ v_4^h \\ v_5^h \\ v_6^h \\ v_7^h \end{pmatrix} = \begin{pmatrix} v_1^{2h} \\ v_2^{2h} \\ v_3^{2h} \end{pmatrix}$$

- I_h^{2h} has rank $\sim \frac{N}{2}$, and thus dim(NS(R)) $\sim \frac{N}{2}$

Multilevel preconditioning methods: MG

Procedure MG**:** $\mathbf{u}^{(k)} \leftarrow MG\left(\mathbf{u}^{(k)}, \mathbf{f}^{(k)}, k, \{\nu_j^{(k)}\}_{j=1}^{k}\right);$

if $k = 0$, **then** solve $A^{(0)}\mathbf{u}^{(0)} = \mathbf{f}^{(0)}$ exactly or by smoothing,

else

$\mathbf{u}^{(k)} \underset{s_1}{\leftarrow} \mathcal{S}_1^{(k)}\left(\mathbf{u}^{(k)}, \mathbf{f}^{(k)}\right)$, perform s_1 pre-smoothing steps,

Correct the residual:

$\mathbf{r}^{(k)} = A^{(k)}\mathbf{u}^{(k)} - \mathbf{f}^{(k)}$; form the current residual,

$\mathbf{r}^{(k-1)} \leftarrow \mathcal{R}\left(\mathbf{r}^{(k)}\right)$, restrict the residual on the next coarser grid,

$\mathbf{e}^{(k-1)} \leftarrow MG\left(\mathbf{0}, \mathbf{r}^{(k-1)}, k-1, \{\nu_j^{(k-1)}\}_{j=1}^{k-1}\right);$

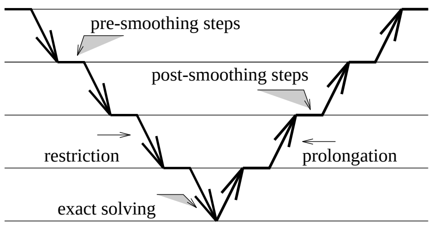
$\mathbf{e}^{(k)} \leftarrow \mathcal{P}\left(\mathbf{e}^{(k-1)}\right)$; prolong the error from the next coarser to the current grid,

$\mathbf{u}^{(k)} = \mathbf{u}^{(k)} - \mathbf{e}^{(k)}$; update the solution,

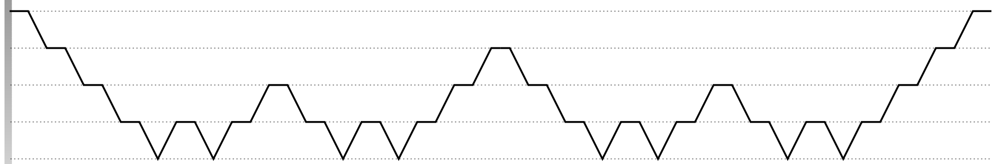
$\mathbf{u}^{(k)} \underset{s_2}{\leftarrow} \mathcal{S}_2^{(k)}\left(\mathbf{u}^{(k)}, \mathbf{f}^{(k)}\right)$, perform s_2 post-smoothing steps.

endif

end Procedure MG



One MG step (V-cycle)



The MG W-cycle

Nested iteration

Procedure NI**:** $\mathbf{u}^{(\ell)} \leftarrow NI\left(\mathbf{u}^{(0)}, \{\mathbf{f}^{(k)}\}_{k=1}^{(\ell)}, \ell, \{\nu^{(k)}\}_{k=1}^{\ell}\right);$

$\mathbf{u}^{(0)} = {A^{(0)}}^{-1}\mathbf{f}^{(0)},$

for $k=1$ **to** ℓ **do**

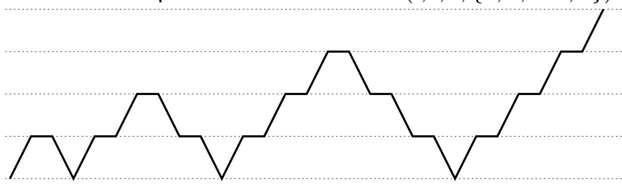
$\mathbf{u}^{(k)} = \mathcal{P}\left(\mathbf{u}^{(k-1)}\right);$

$\mathbf{u}^{(k)} \leftarrow MG\left(\mathbf{u}^{(k)}, \mathbf{f}^{(k)}, k, \{\nu_j^{(k)}\}_{j=1}^{k}\right);$

endfor

end Procedure NI

The so-called *full* MG corresponds to **Procedure** $NI(\cdot,\cdot,\ell,\{1,1,\cdots,1\})$



The full MG (V-cycle)

A compact formula presenting the MG procedure in terms of a recursively defined iteration matrix:

(i) Let $M^{(0)} = 0$,

(ii) For $k = 1$ to ℓ, define

$$M^{(k)} = \mathcal{S}^{(k)^{s_2}} \left(A^{(k)^{-1}} - \mathcal{P}_{k-1}^{k} \left(I - M^{(k-1)^{\nu}} \right) A^{(k-1)^{-1}} \mathcal{R}_{k}^{k-1} \right) A^{(k)} \mathcal{S}^{(k)^{s_1}},$$

where $\mathcal{S}^{(k)}$ is a smoothing iteration matrix (assuming $\mathcal{S}_1$ and $\mathcal{S}_2$ are the same), $\mathcal{R}_{k}^{k-1}$ and $\mathcal{P}_{k-1}^{k}$ are matrices which transfer data between two consecutive grids and correspond to the restriction and prolongation operators $\mathcal{R}$ and $\mathcal{P}$, respectively, and $\nu = 1$ and $\nu = 2$ correspond to the V- and W-cycles.

It turns out that in many cases the spectral radius of $M^{(\ell)}$, $\rho\left(M^{(\ell)}\right)$, is independent of ℓ, thus the rate of convergence of the NI method is optimal. Also, a mechanism to make the spectral radius of $M^{(\ell)}$ smaller is to choose s_1 and s_2 larger. The price for the latter is, clearly, a higher computational cost.

MG ingredients

- smoothers (many different)
  - Jacobi, weighted Jacobi ($\omega diag(A)$, GS, SOR, SSOR, SPAI
- restriction and prolongation operators
- coarse level matrix (approximation properties)

MG: Rate of convergence and computational complexity

Let one Work Unit (WU) be the cost of one relaxation sweep on the fine-grid.
– Ignore the cost of restriction and interpolation (typically about 20% of the total cost).
– Consider a V-cycle with 1 pre-smoothing and 1 post-smoothing sweep.
– In d-dimensions, each coarse grid has about 2^{-d} the number of points as the finer grid. – Cost of V-cycle (in WU):

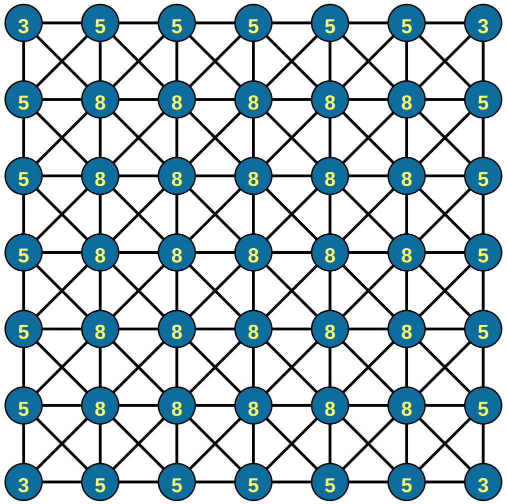
$$2(1 + 2^{-d} + 2^{-2d} + +2^{-3d} + \cdots + 2^{-\ell d} \leq \frac{2}{1 - 2^{-d}}.$$

– Total storage:

$$2N^d(1 + 2^{-d} + 2^{-2d} + +2^{-3d} + \cdots + 2^{-\ell d} \leq \frac{2N^d}{1 - 2^{-d}}.$$

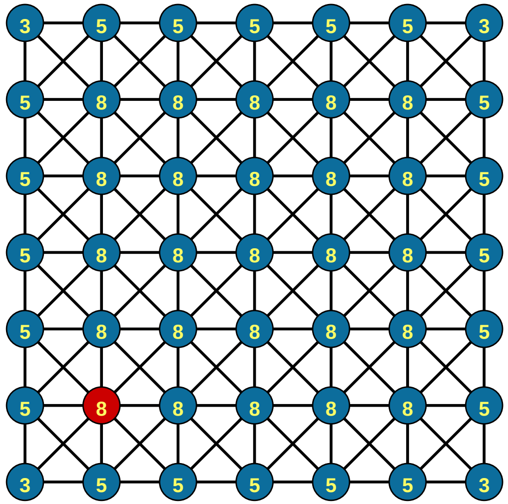
Algebraic Multigrid

C-AMG coarsening



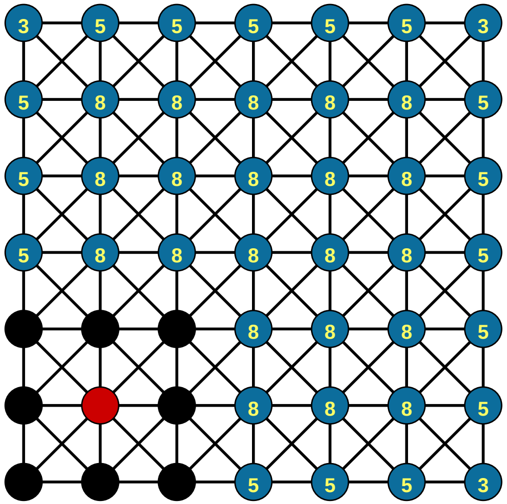
- ➔ select C-pt with maximal measure
- ➔ select neighbors as F-pts
- ➔ update measures of F-pt neighbors

C-AMG coarsening



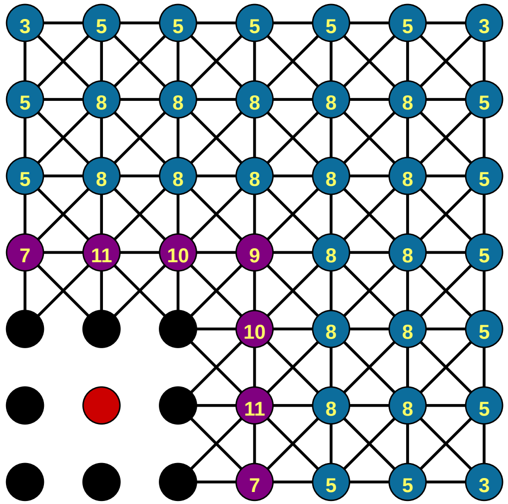
- ➔ select C-pt with maximal measure
- ➔ select neighbors as F-pts
- ➔ update measures of F-pt neighbors

C-AMG coarsening



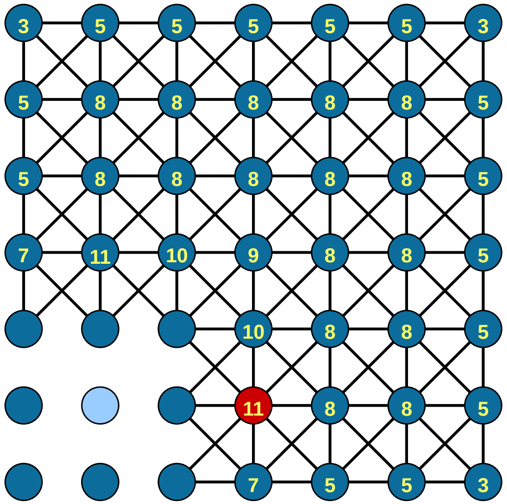
- ➔ select C-pt with maximal measure
- ➔ select neighbors as F-pts
- ➔ update measures of F-pt neighbors

C-AMG coarsening



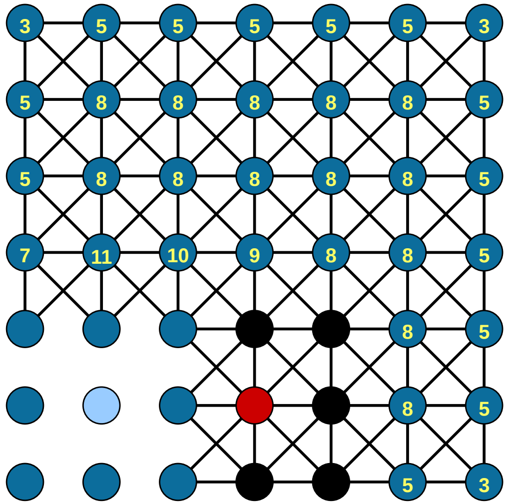
- ➔ select C-pt with maximal measure
- ➔ select neighbors as F-pts
- ➔ update measures of F-pt neighbors

C-AMG coarsening



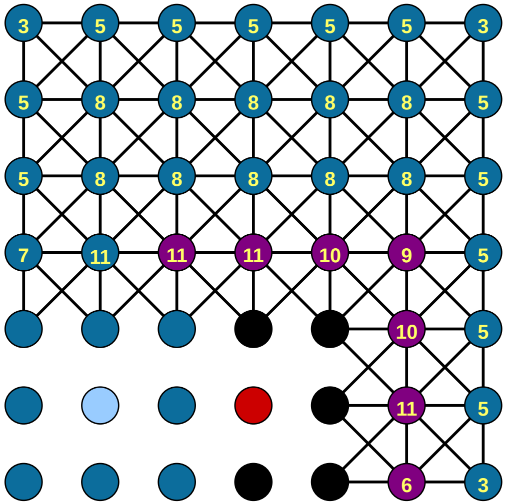
- ➔ select C-pt with maximal measure
- ➔ select neighbors as F-pts
- ➔ update measures of F-pt neighbors

C-AMG coarsening



- ➔ select C-pt with maximal measure
- ➔ select neighbors as F-pts
- ➔ update measures of F-pt neighbors

C-AMG coarsening

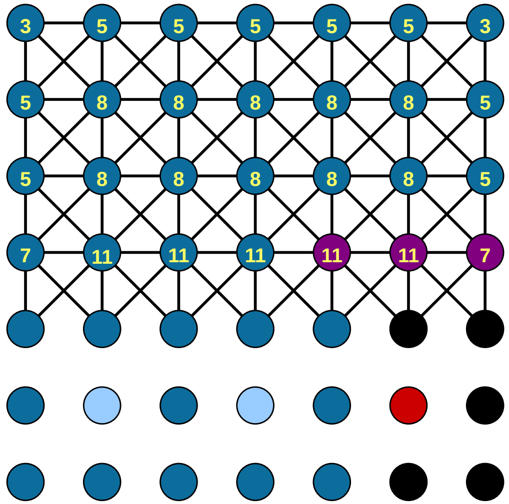


➔ select C-pt with maximal measure

➔ select neighbors as F-pts

➔ update measures of F-pt neighbors

C-AMG coarsening

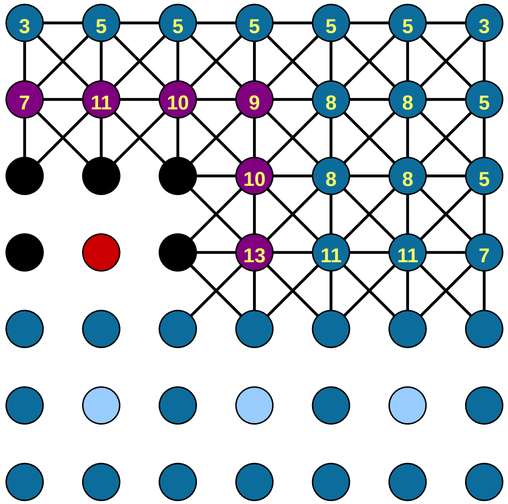


➔ select C-pt with maximal measure

➔ select neighbors as F-pts

➔ update measures of F-pt neighbors

C-AMG coarsening

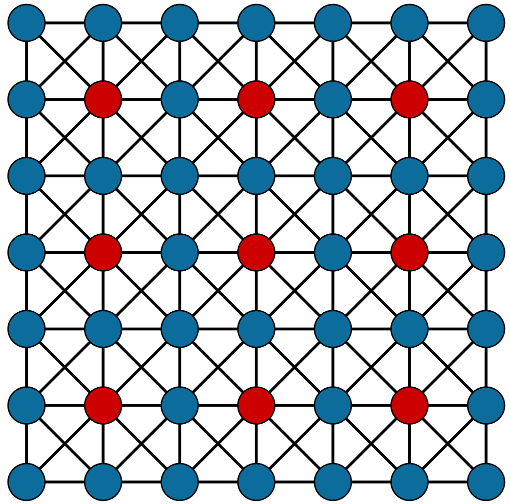


➔ select C-pt with maximal measure

➔ select neighbors as F-pts

➔ update measures of F-pt neighbors

C-AMG coarsening is inherently sequential



➔ select C-pt with maximal measure

➔ select neighbors as F-pts

➔ update measures of F-pt neighbors

AMG: The ideal prolongation and restriction

Reference: Wiesner, Tuminaro, Wall, Gee
Multigrid transfers for nonsymmetric systems based on Schur complements and Galerkin projections, *NLA, 2013*

AMG and the Schur complement

$$\begin{pmatrix} A_{ff} & A_{fc} \\ A_{cf} & A_{cc} \end{pmatrix} \begin{pmatrix} x_f \\ x_c \end{pmatrix} = \begin{pmatrix} b_f \\ b_c \end{pmatrix}.$$

Assuming A_{ff} to be invertible, A has the corresponding LDU decomposition

$$\begin{pmatrix} A_{ff} & A_{fc} \\ A_{cf} & A_{cc} \end{pmatrix} = \begin{pmatrix} I & 0 \\ A_{cf}A_{ff}^{-1} & I \end{pmatrix} \begin{pmatrix} A_{ff} & 0 \\ 0 & S \end{pmatrix} \begin{pmatrix} I & A_{ff}^{-1}A_{fc} \\ 0 & I \end{pmatrix}$$

where $S = A_{cc} - A_{cf}A_{ff}^{-1}A_{fc}$ and is referred to as the Schur complement.

TDB – NLA

Define

$$\mathcal{R}^{opt} = \begin{pmatrix} -A_{cf}A_{ff}^{-1} & I \end{pmatrix}, \; \mathcal{P}^{opt} = \begin{pmatrix} -A_{ff}^{-1}A_{fc} \\ I \end{pmatrix} \quad \text{and} \quad \hat{I} = \begin{pmatrix} I \\ 0 \end{pmatrix}.$$

One can easily verify that $S = \mathcal{R}^{opt}A\mathcal{P}^{opt}$,

$$\begin{pmatrix} I & 0 \\ A_{cf}A_{ff}^{-1} & I \end{pmatrix}^{-1} = \begin{pmatrix} \hat{I}^T \\ \mathcal{R}^{opt} \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} I & A_{ff}^{-1}A_{fc} \\ 0 & I \end{pmatrix}^{-1} = \begin{pmatrix} \hat{I} & \mathcal{P}^{opt} \end{pmatrix}.$$

Application of the inverses of the three operators in the exact factorization is equivalent to restriction at the c-points , followed by solution of two systems: A_{ff} which can be interpreted as relaxation and $\mathcal{R}^{opt}A\mathcal{P}^{opt}$ which is the coarse correction. Finally, the coarse correction is interpolated and added to the relaxation solution. As this procedure is exact, it converges in one iteration.

TDB – NLA

Further work:
how to approximate $\mathcal{R}^{opt}$, $\mathcal{P}^{opt}$ and S, or rather the coarse correction $\mathcal{R}^{opt}A\mathcal{P}^{opt}$, which is nothing but $A_{cf}A_{ff}^{-1}A_{fc}$.

We enter the full block factorized preconditioning framework, that can be seen as purely algebraic and not related to MG.

TDB – NLA

Algebraic Multilevel Iteration Methods (AMLI)

The so-called AMLI methods have been developed by Owe Axelsson and Panayot Vassilevski in a series of papers betwee 1989 and 1991.
These methods were originally developed for elliptix problems and spd matrices, and are the first regularity-free optimal order preconditioning methods.

Sequence of matrices $\left\{A^{(k)}\right\}_{k=k_0}^{\ell}$

$$N_{k_0} \subset N_{k_0+1} \subset \ldots \subset N_\ell$$

$$A^{(k)} = \begin{bmatrix} A_{11}^{(k)} & A_{12}^{(k)} \\ A_{21}^{(k)} & A_{22}^{(k)} \end{bmatrix} \begin{matrix} \} \, N_k \backslash N_{k-1} \\ \} \, N_{k-1} \end{matrix} .$$

TDB – NLA

$A^{(k)}$ has to approximate $S_{A^{(k+1)}}$ in some way. For instance,

$$A^{(k)} = A_{22}^{(k+1)} - A_{21}^{(k+1)} B_{11}^{(k+1)} A_{12}^{(k+1)}.$$

where $B_{11}^{(k+1)}$ is some sparse, positive definite, nonnegative and symmetric approximation of ${A_{11}^{(k+1)}}^{-1}$.
How to split N_{k+1} into two parts: the order n_k of the matrices $A^{(k)}$ should decrease geometrically:

$$\frac{n_{k+1}}{n_k} = \rho_k \geq \rho > 1.$$

TDB – NLA

$M^{(k_0)} = A^{(k_0)}$,

for $k = k_0, k_0+1, \ldots \ell - 1$

$$M^{(k+1)} = \begin{bmatrix} A_{11}^{(k+1)} & 0 \\ A_{21}^{(k+1)} & \widetilde{S}^{(k)} \end{bmatrix} \begin{bmatrix} I_1^{(k+1)} & {A_{11}^{(k+1)}}^{-1} A_{12}^{(k+1)} \\ 0 & I_2^{(k+1)} \end{bmatrix},$$

endfor

where $\widetilde{S}^{(k)}$ can be, for instance:

$$\widetilde{S}^{(k)} = A^{(k)} \left[I - P_\nu({M^{(k)}}^{-1} A^{(k)}) \right]^{-1},$$

$P_\nu(t)$ denotes a polynomial of degree ν.

We could use some other way of stabilization.

TDB – NLA

Forward sweep:

Solve $\begin{bmatrix} A_{11}^{(k+1)} & 0 \\ A_{21}^{(k+1)} & \widetilde{S}^{(k)} \end{bmatrix} \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}$, i.e.

$$
\begin{aligned}
(F1) \quad \mathbf{w}_1 &= {A_{11}^{(k+1)}}^{-1} \mathbf{y}_1, \\
(F2) \quad \mathbf{w}_2 &= {\widetilde{S}^{(k)}}^{-1} \left(\mathbf{y}_2 - A_{21}^{(k+1)} \mathbf{w}_1 \right).
\end{aligned}
$$

Backward sweep:

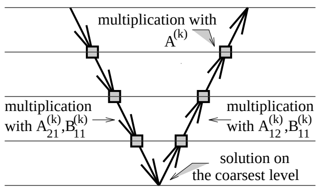
Solve $\begin{bmatrix} I_1^{(k+1)} & {A_{11}^{(k+1)}}^{-1} A_{12}^{(k+1)} \\ 0 & I_2^{(k+1)} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{bmatrix}$, i.e.

$$
\begin{aligned}
(B1) \quad \mathbf{x}_2 &= \mathbf{w}_2, \\
(B2) \quad \mathbf{x}_1 &= \mathbf{w}_1 - {A_{11}^{(k+1)}}^{-1} A_{12}^{(k+1)} \mathbf{x}_2.
\end{aligned}
$$

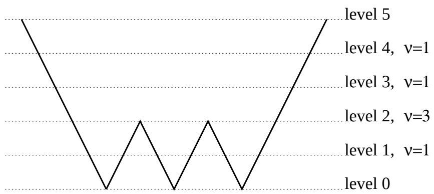
Procedure $AMLI$: $\mathbf{u}^{(k)} \leftarrow AMLI\left(\mathbf{f}^{(k)}, k, \nu_k, \{a_j^{(k)}\}_{j=0}^{\nu_k}\right)$;

- $[\mathbf{f}_1^{(k)}, \mathbf{f}_2^{(k)}] \leftarrow \mathbf{f}^{(k)}$,
- $\mathbf{w}_1^{(k)} = B_{11}^{(k)} \mathbf{f}_1^{(k)}$,
- $\mathbf{w}_2^{(k)} = \mathbf{f}_2^{(k)} - A_{21}^{(k)} \mathbf{w}_1^{(k)}$,
- $k = k - 1$,
- **if** $k = 0$ **then** $\mathbf{u}_2^{(0)} = A^{(0)} \mathbf{w}_2^{(1)}$, solve on the coarsest level exactly;
- **else**
  - $\mathbf{u}_2^{(k)} \leftarrow AMLI\left(a_{\nu_k}^{(k)} \mathbf{w}_2^{(k)}, k, \nu_k, \{a_j^{(k)}\}_{j=0}^{\nu_k}\right)$;
  - **for** $j = 1$ **to** $\nu_k - 1$:
    - $\mathbf{u}_2^{(k)} \leftarrow AMLI\left(A^{(k)} \mathbf{u}_2^{(k)} + a_{\nu_k - j}^{(k)} \mathbf{w}_2^{(k)}, k, \nu_k, \{a_j^{(k)}\}_{j=0}^{\nu_k}\right)$;
  - **endfor**
- **endif**
- $k = k + 1$,
- $\mathbf{u}_1^{(k)} = \mathbf{w}_1^{(k)} - B_{11}^{(k)} A_{12}^{(k)} \mathbf{u}_2^{(k)}$,
- $\mathbf{u}^{(k)} \leftarrow [\mathbf{u}_1^{(k)}, \mathbf{u}_2^{(k)}]$

end Procedure $AMLI$

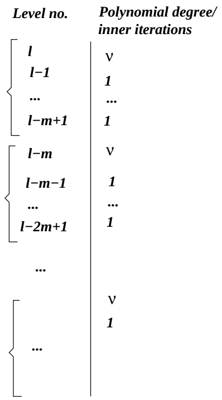


One AMLI step (V-cycle)



ν-fold W-cycle, $[1, 1, 3, 1]$

AMLI: Computational complexity



$$
\begin{aligned}
w_\ell = & C(n_\ell + \cdots + n_{\ell-\mu}) \\
& + C\nu(n_{\ell-\mu-1} + \cdots + n_{\ell-2\mu-1}) \\
& + C\nu^2(n_{\ell-2\mu-2} + \cdots + n_{\ell-3\mu-2}) \\
& + \cdots \\
\leq & C n_\ell \left[1 + \frac{1}{\rho} + \cdots + \left(\frac{1}{\rho}\right)^\mu\right] \frac{1}{1 - \nu\rho^{-(\mu+1)}},
\end{aligned}
$$

where $1 < \rho \leq \rho_k = \frac{n_{k+1}}{n_k}$, $k = 0, 1, \ldots \ell - 1$. Hence

$$\nu < \rho^{\mu+1}$$