Multilevel Preconditioning Based on Element Agglomeration

Johannes Kraus
University of Leoben
Department of Applied Mathematics

ELEVENTH COPPER MOUNTAIN CONFERENCE
ON MULTIGRID METHODS
Copper Mountain, April 1, 2003

Contents

- Type of problem and method
- 3 test problems
- Element agglomeration
- The Schur Complement Approximation
- ILU based on element agglomeration
- Factorization of the pivot matrix
- Nonlinear Algebraic Multilevel Iteration
- Outlook

Type of Problem:

- elliptic PDEs (salar equations, or systems)
- finite element problems
- (general) SPD matrices

Type of Method:

- preconditioning based on element agglomeration
- hierarchical incomplete factorization
- nonlinear Algebraic Multilevel Iteration
- inner and outer iterations of CG or GCG typ

The main emphasis is on non-M matrices.

Problem 1

$$-(\Delta u + 2\alpha \frac{\partial^2 u}{\partial x \partial y}) = f \quad on \quad \Omega = [0,1]\times[0,1]$$
$$u = 0 \quad on \quad \Gamma, \quad |\alpha| < 1$$

Grid: orthogonal and uniform.

Discretization by the finite difference stencil:

$$\frac{1}{h^2} \cdot \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} + \frac{\alpha}{h^2} \cdot \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

or, equivalently, the element matrix:

$$A_e = \frac{1}{h^2} \cdot \begin{pmatrix} 1 & -\frac{1+\alpha}{2} & -\frac{1+\alpha}{2} & \alpha \\ -\frac{1+\alpha}{2} & 1+\alpha & 0 & -\frac{1+\alpha}{2} \\ -\frac{1+\alpha}{2} & 0 & 1+\alpha & -\frac{1+\alpha}{2} \\ \alpha & -\frac{1+\alpha}{2} & -\frac{1+\alpha}{2} & 1 \end{pmatrix}$$

For $\alpha > 0$ the stiffness matrix will be a non-M matrix!

Problem 2

$$
\begin{aligned}
a(u,v) &= (f,v) \quad \forall \quad v \in H^1(\Omega) \\
a(u,v) &= \int_\Omega (\nabla v)^T D(x) \nabla u dx + \int_\Omega \sigma u v dx \\
\frac{\partial u}{\partial n} &= 0 \quad on \quad \Gamma \\
D(x) &= \begin{pmatrix} 1 & 0 \\ 0 & \epsilon \end{pmatrix}, \quad 0 < \epsilon \leq 1
\end{aligned}
$$

Domain: Ω is the unit square.

Discretization: quadratic finite elements with bilinear Ansatz functions.

Element matrix:

$$
A_e = \frac{1}{\epsilon h^2} \cdot \begin{pmatrix} 2+2\epsilon^2 & 1-2\epsilon^2 & -2+\epsilon^2 & -1-\epsilon^2 \\ 1-2\epsilon^2 & 2+2\epsilon^2 & -1-\epsilon^2 & -2+\epsilon^2 \\ -2+\epsilon^2 & -1-\epsilon^2 & 2+2\epsilon^2 & 1-2\epsilon^2 \\ -1-\epsilon^2 & -2+\epsilon^2 & 1-2\epsilon^2 & 2+2\epsilon^2 \end{pmatrix}
$$

For $|\epsilon| < \sqrt{2}/2$ the stiffness matrix will be a non-M matrix!

Problem 3

$$\begin{aligned} \frac{\partial^2 u}{\partial x^2} + \frac{1-\mu}{2} \cdot \frac{\partial^2 u}{\partial y^2} + \frac{1+\mu}{2} \cdot \frac{\partial^2 v}{\partial x \partial y} &= f \quad on \quad \Omega \\ \frac{1+\mu}{2} \cdot \frac{\partial^2 u}{\partial x \partial y} + \frac{1-\mu}{2} \cdot \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} &= g \quad on \quad \Omega \\ u = v &= 0 \quad on \quad \Gamma \end{aligned}$$

Domain: Ω is the unit square.

Discretization: quadratic finite elements.

Element matrix $A_e = \frac{1}{3\gamma_1\gamma_2 h^2} \begin{pmatrix} B_e & -C_e \\ -C_e^T & B_e \end{pmatrix}$:

$$B_e : \begin{pmatrix} 4(1+\gamma_1) & 3\gamma_2 & 2(1-2\gamma_1) & \gamma_3 \\ 3\gamma_2 & 4(1+\gamma_1) & -\gamma_3 & -2(2-\gamma_1) \\ 2(1-2\gamma_1) & -\gamma_3 & 4(1+\gamma_1) & -3\gamma_2 \\ \gamma_3 & -2(2-\gamma_1) & -3\gamma_2 & 4(1+\gamma_1) \end{pmatrix}$$

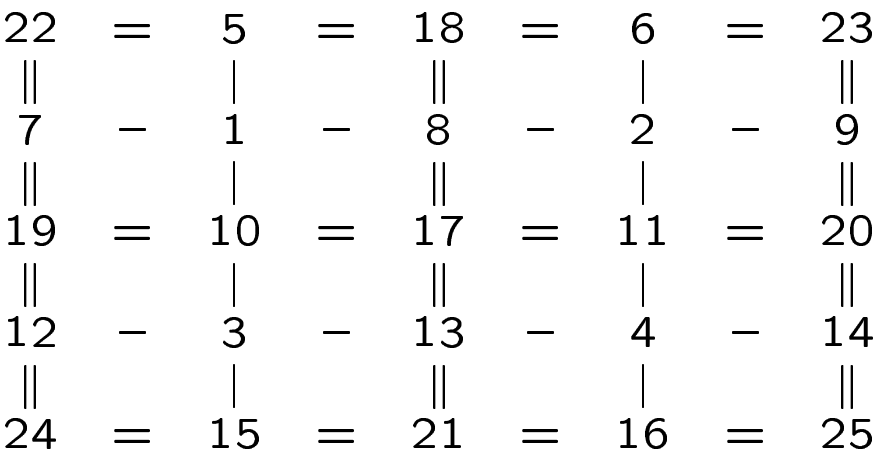
$$C_e : \begin{pmatrix} 2(1+\gamma_1) & 3\gamma_2 & 2(2-\gamma_1) & \gamma_3 \\ 3\gamma_2 & 2(1+\gamma_1) & -\gamma_3 & -2(1-2\gamma_1) \\ 2(2-\gamma_1) & -\gamma_3 & 2(1+\gamma_1) & -3\gamma_2 \\ \gamma_3 & -2(1-2\gamma_1) & -3\gamma_2 & 2(1+\gamma_1) \end{pmatrix}$$

$\gamma_1 = (1-\mu)/2$, $\gamma_2 = (1+\mu)/2$ and $\gamma_3 = 3(1-3\mu)/2$.

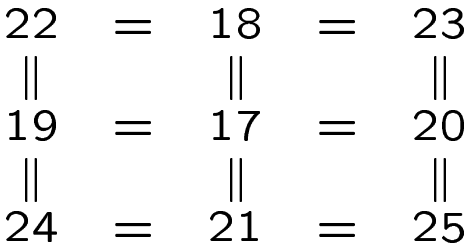
We consider a simple element agglomeration of the following type:

- dofs are associated with vertices of elements, that is, no dofs on edges (or faces) of elements
- elements are agglomerated along edges in 2D (and faces in 3D)
- agglomerates build coarse grid elements at each level, where coarse dofs are associated with vertices of coarse-grid elements

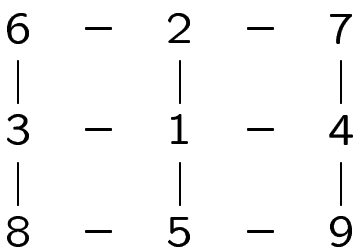
Fine-grid elements:



Coarse-grid elements:



Local ordering of dofs within an agglomerate:



Fine-grid elements:

$$
\begin{array}{ccccccccc}
43,44 & = & 9,10 & = & 35,36 & = & 11,12 & = & 45,46 \\
\| & & | & & \| & & | & & \| \\
13,14 & - & 1,2 & - & 15,16 & - & 3,4 & - & 17,18 \\
\| & & | & & \| & & | & & \| \\
37,38 & = & 19,20 & = & 33,34 & = & 21,22 & = & 39,40 \\
\| & & | & & \| & & | & & \| \\
23,24 & - & 5,6 & - & 25,26 & - & 7,8 & - & 27,28 \\
\| & & | & & \| & & | & & \| \\
47,48 & = & 29,30 & = & 41,42 & = & 31,32 & = & 49,50
\end{array}
$$

Coarse-grid elements:

$$
\begin{array}{ccccc}
43,44 & = & 35,36 & = & 45,46 \\
\| & & \| & & \| \\
37,38 & = & 33,34 & = & 39,40 \\
\| & & \| & & \| \\
47,48 & = & 41,42 & = & 49,50
\end{array}
$$

Local ordering of dofs within an agglomerate:

$$
\begin{array}{ccccc}
11,12 & - & 3,4 & - & 13,14 \\
| & & | & & | \\
5,6 & - & 1,2 & - & 7,8 \\
| & & | & & | \\
15,16 & - & 9,10 & - & 17,18
\end{array}
$$

With respect to the ordering of dofs within an agglomerate, the agglomerate matrices take the 2×2 block form

$$A_a = \begin{pmatrix} A_{a,11} & A_{a,12} \\ A_{a,21} & A_{a,22} \end{pmatrix},$$

where $A_{a,11}$ and $A_{a,22}$ are the blocks corresponding to fine dofs and coarse dofs (coarse-element dofs), respectively.

This allows us to compute the local Schur complements

$$S_a = A_{a,22} - A_{a,21}(A_{a,11})^{-1}A_{a,12}$$

that serve as element matrices on the next coarser level, i.e.,

$$A_e^{(k+1)} := S_a^{(k)}$$

for all agglomerates a at level k.

As a global Schur complement approximation we now take the globally assembled local Schur complements (coarse-element matrices), i.e.,

$$\tilde{S}^{(k)} = \sum_a S_a^{(k)} = \sum_e A_e^{(k+1)}.$$

That is, the approximation matrix $\tilde{S}^{(k)}$ acts as coarse-grid operator, i.e., $A^{(k+1)} := \tilde{S}^{(k)}$.

We will use the assembled Schur complement approximation later on.

Our next goal will be to assemble an incomplete factorization of the pivot-matrix (that yields the exact Schur complement).

Thus, all components of an algebraic multilevel iteration will be computed locally from agglomerate matrices!

A straight-forward approach can be used to define a new type of incomplete factorization method:

At levels $0 \leq k < l$, let

$$A_a^{(k)} = \begin{pmatrix} A_{a,11}^{(k)} & A_{a,12}^{(k)} \\ A_{a,21}^{(k)} & A_{a,22}^{(k)} \end{pmatrix},$$

and $A_{a,11}^{(k)} = L_a^{(k)} U_a^{(k)}$, where $\mathrm{diag}(L_a^{(k)}) = I$.

Further on, let $A^{(l)} = \tilde{S}^{(l-1)} = L^{(l)} U^{(l)}$, $\mathrm{diag}(L^{(l)}) = I$.

Then, since

$$A_a^{(k)} = \begin{pmatrix} L_a^{(k)} & 0 \\ A_{a,21}^{(k)} (U_a^{(k)})^{-1} & I \end{pmatrix} \cdot \begin{pmatrix} U_a^{(k)} & (L_a^{(k)})^{-1} A_{a,12}^{(k)} \\ 0 & S_a^{(k)} \end{pmatrix}$$

we define

$$\tilde{U}^{(l)} := \sum_{k=0}^{l-1} \sum_a \left[U_a^{(k)}, (L_a^{(k)})^{-1} A_{a,12}^{(k)} \right]$$

and, finally,

$$U := \tilde{U}^{(l)} + U^{(l)}, \quad \text{and} \quad L = U^T (\mathrm{diag}(U))^{-1}.$$

Now, let us use this idea in order to compute an incomplete LU factorization of the pivot matrix $A_{11}^{(k)}$ at every level k, $0 \leq k < l$:

The pivot block $A_{11}^{(k)}$ of

$$A^{(k)} = \begin{pmatrix} A_{11}^{(k)} & A_{12}^{(k)} \\ A_{21}^{(k)} & A_{22}^{(k)} \end{pmatrix}$$

is given by the assembly of the local pivot blocks, i.e.,

$$A_{11}^{(k)} = \sum_a A_{a,11}^{(k)}.$$

Thus,

$$\tilde{U}^{(k)} := \sum_a U_a^{(k)}$$

gives an approximation of $U^{(k)}$, where

$$A_{a,11}^{(k)} = L_a^{(k)} U_a^{(k)}, \quad A_{11}^{(k)} = L^{(k)} U^{(k)},$$

and

$$\tilde{L}^{(k)} = (\tilde{U}^{(k)})^T (\mathrm{diag}(\tilde{U}^{(k)}))^{-1}.$$

As numerical experiments show, a simple modification helps to improve the approximations $\tilde{U}^{(k)}$ and $\tilde{L}^{(k)}$.

That is, we adjust the diagonal entries of $\tilde{U}^{(k)}$ such that

$$\mathrm{diag}(\tilde{A}_{11}^{(k)}) = \mathrm{diag}(A_{11}^{(k)})$$

where

$$A_{11}^{(k)} = L^{(k)} U^{(k)}$$

and

$$\tilde{A}_{11}^{(k)} = \bar{L}^{(k)} \bar{U}^{(k)}.$$

This is done by replacing the main diagonal of $\tilde{U}^{(k)}$ by $(\bar{u}_{ii})_{i=1}^{n}$, where

$$\begin{aligned} \bar{u}_{ii} &= a_{ii} - \sum_{j=1}^{i-1} \frac{\tilde{u}_{ji}^2}{\bar{u}_{jj}} \\ \bar{u}_{ij} &= \tilde{u}_{ij} \quad \text{for} \quad j > i. \end{aligned}$$

$\lambda_{max}((\tilde{A}_{11}^{(0)})^{-1}A_{11}^{(0)})/\lambda_{min}((\tilde{A}_{11}^{(0)})^{-1}A_{11}^{(0)})$:

Problem 1

α	4 aggs	16 aggs	64 aggs
0	1.06955	1.07786	1.08069
0.5	1.07408	1.07447	1.07428
0.9	1.10782	1.11126	1.11164
0.99	1.11729	1.12165	1.12206

Problem 2

ϵ	4 aggs	16 aggs	64 aggs
0.5	1.16665	1.19919	1.21136
0.25	1.10414	1.13528	1.14592
0.1	1.02519	1.03400	1.04152
0.01	1.00215	1.00225	1.00228

Problem 3

μ	4 aggs	16 aggs	64 aggs
0.1	1.43393	1.46719	1.47636
0.25	1.53208	1.55038	1.55821
$0.\dot{3}$	1.59849	1.60504	1.61168
0.5	1.76591	1.74664	1.75159

Linear System: $A^{(0)}\mathbf{x}^{(0)} = \mathbf{b}$

initialize: **for** $k = 0$ **to** l set $\mathbf{x}^{(k)} := \mathbf{0}$
$k := 0$
$\mathbf{d}^{(0)} := \mathbf{b}$

forward: $\sigma_k := \sigma_k + 1$
if $(\sigma_k = 1$ && $k \neq 0)$
$\mathbf{x}^{(k)} := \mathbf{0}$
$\mathbf{r}^{(k)} := \mathbf{d}^{(k)}$
end
pcgsolve: $\mathbf{p_1}_{(\sigma_k)}^{(k)} \cong (A_{1,1}^{(k)})^{-1}\mathbf{r_1}^{(k)}$

$\mathbf{d}^{(k+1)} := \mathbf{r_2}^{(k)} - A_{2,1}^{(k)}\mathbf{p_1}_{(\sigma_k)}^{(k)}$
$k := k + 1$
if $k < l$ **goto forward**
solve: $L^{(l)}U^{(l)}\mathbf{x}^{(l)} = \mathbf{d}^{(l)}$

backward: $\sigma_k = 0$; $k := k - 1$
$\mathbf{p_2}_{(\sigma_k)}^{(k)} := \mathbf{x}^{(k+1)}$
solve: $\tilde{L}^{(k)}\tilde{U}^{(k)}\mathbf{y}^{(k)} = A_{1,2}^{(k)}\mathbf{p_2}_{(\sigma_k)}^{(k)}$
$\mathbf{p_1}_{(\sigma_k)}^{(k)} := \mathbf{p_1}_{(\sigma_k)}^{(k)} - \mathbf{y}^{(k)}$
if $\nu_k = 0$
$\mathbf{x}^{(k)} := \mathbf{p}_{(\sigma_k)}^{(k)}$; **goto backward**
end

$\mathbf{q}^{(k)}_{(\sigma_k)} := A^{(k)}\mathbf{p}^{(k)}_{(\sigma_k)}$
for $j = 1$ **to** $\sigma_k - 1$
  $\beta = (\mathbf{q}^{(k)}_{(\sigma_k)}, \mathbf{p}^{(k)}_{(j)})/\gamma^{(k)}_{(j)}$
  $\mathbf{p}^{(k)}_{(\sigma_k)} := \mathbf{p}^{(k)}_{(\sigma_k)} - \beta\mathbf{p}^{(k)}_{(j)}$
  $\mathbf{q}^{(k)}_{(\sigma_k)} := \mathbf{q}^{(k)}_{(\sigma_k)} - \beta\mathbf{q}^{(k)}_{(j)}$
end
$\gamma^{(k)}_{(\sigma_k)} = (\mathbf{q}^{(k)}_{(\sigma_k)}, \mathbf{p}^{(k)}_{(\sigma_k)})$
$\alpha = (\mathbf{r}^{(k)}, \mathbf{p}^{(k)}_{(\sigma_k)})/\gamma^{(k)}_{(\sigma_k)}$
$\mathbf{x}^{(k)} := \mathbf{x}^{(k)} + \alpha\mathbf{p}^{(k)}_{(\sigma_k)}$
$\mathbf{r}^{(k)} := \mathbf{r}^{(k)} - \alpha\mathbf{q}^{(k)}_{(\sigma_k)}$
if $(k = 0)$
  if (termination) **goto finish**
  if $(\sigma_0 = \nu_0)$ $\sigma_0 = 0$
end
if $(\sigma_k < \nu_k)$ **goto forward**
goto backward

finish:

The above algorithm performs a nonlinear AMLI with inner GCG and PCG iterations, where ν_k is the number of inner GCG iterations at level k; restart at the fine-grid level after every ν_0 iterations.

The number of dofs at level $k+1$ is approximately one fourth of the number of dofs at level k. Hence, a visit at level k must induce less than four visits at level $k+1$ for optimal order of computational complexity of each outer iteration.

The PCG-solve in the forward loop is not critical in this respect if a fixed number of iterations is not exceeded, but it may increase the work per outer iteration significantly.

In our tests we used 3 inner PCG iterations in the forward loop and 2 inner GCG iterations at levels $1, 3, 5, \ldots$. The tables summarize the number of outer iterations that reduced the L_2-norm of the initial residual by a factor 10^{-6}.

Number of outer iterations for 3 to 8 levels (64 to 65536 elts):

Problem 1

$\alpha \backslash l$	3	4	5	6	7	8
0	4	5	5	5	6	6
0.5	4	5	5	5	5	6
0.9	4	5	5	6	6	6
0.99	5	5	6	6	6	7

Problem 2

$\epsilon \backslash l$	3	4	5	6	7	8
0.5	7	8	8	9	9	9
0.25	6	9	9	10	10	10
0.1	5	7	9	9	10	10
0.01	2	2	3	3	4	4

Problem 3

$\mu \backslash l$	3	4	5	6	7	8
0.1	5	6	7	8	8	8
0.25	5	6	8	8	8	8
$0.\dot{3}$	5	7	8	8	8	9
0.5	5	7	8	9	9	10

Future investigations will deal with:

- numerical tests for 3D Problems and unstructured FE meshes
- more sophisticated agglomeration techniques (and the optimal size of agglomerates)
- different types of elements and problems
- a generalization of the method to unsymmetric problems
- a theoretical convergence analysis (if possible)