Multigrid Tutorial

Multigrid (MG) and Local Refinement for Elliptic Partial Differential Equations

Klaus Stüben

FhG-SCAI
Schloss Birlinghoven
53754 St. Augustin, Germany
e-mail: Klaus.Stueben@scai.fraunhofer.de

Overview

• Why multigrid?
• Basic multigrid principles
• Full multigrid (FMG)
• Nonlinear multigrid (FAS)
• Eigenproblems
• Local Refinements
Why Multigrid?

Large problems require optimal solvers.

Optimal solvers require hierarchical algorithms:
- Multigrid
- Multilevel
- Multiscale

The diagram shows the computational work per variable as a function of the logarithm of the problem size (Log(N)).

- One-level solvers: $O(N)$, $O(N^{1.2})$, $O(N^{1.5})$, $O(N^2)$, $O(N^3)$.
- Optimal solvers: $O(N \log(N))$. 

The diagram illustrates the efficiency of different algorithms in solving large problems.
### Model Problem: 2D Poisson Equation

\[ Lu = -\Delta u \quad + \text{Dirichlet boundary conditions} \]

\[ Lu(x, y) = f(x, y) \quad ((x, y) \in \Omega) \]

\[ L_h u_h(x, y) = f_h(x, y) \quad ((x, y) \in \Omega_h) \]

\[ L_h = \frac{1}{h^2} \begin{bmatrix} -1 & 4 & -1 \\ -1 & 4 & -1 \\ -1 & 4 & -1 \end{bmatrix}_h \]

\[ h = \frac{1}{n}, \quad N = (n-1)^2 \]

\[ \rightarrow A_h u_h = f_h \]

---

### Model Problem: Solution Methods

<table>
<thead>
<tr>
<th>Method:</th>
<th>Complexity (accuracy (\varepsilon)):</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gauss Elimination</td>
<td>(O(N^2))</td>
</tr>
<tr>
<td>Jacobi iteration</td>
<td>(O(N^2)) log((\varepsilon))</td>
</tr>
<tr>
<td>Gauss-Seidel iteration</td>
<td>(O(N^2)) log((\varepsilon))</td>
</tr>
<tr>
<td>SOR</td>
<td>(O(N^{3/2})) log((\varepsilon))</td>
</tr>
<tr>
<td>Conjugate Gradient</td>
<td>(O(N^{3/2})) log((\varepsilon))</td>
</tr>
<tr>
<td>Nested Dissection</td>
<td>(O(N^{3/2}))</td>
</tr>
<tr>
<td>ICCG</td>
<td>(O(N^{5/4})) log((\varepsilon))</td>
</tr>
<tr>
<td>ADI</td>
<td>(O(N \log(N))) log((\varepsilon))</td>
</tr>
<tr>
<td>FFT</td>
<td>(O(N \log(N)))</td>
</tr>
<tr>
<td>Buneman</td>
<td>(O(N \log(N)))</td>
</tr>
<tr>
<td>Total Reduction</td>
<td>(O(N))</td>
</tr>
<tr>
<td>Multigrid (iterative)</td>
<td>(O(N)) log((\varepsilon))</td>
</tr>
<tr>
<td>Multigrid (FMG)</td>
<td>(O(N))</td>
</tr>
</tbody>
</table>
Quantitative Example: Molecular Dynamics

Analysis of dynamical behavior of biological systems

\[
\begin{align*}
\text{Time integration:} & \quad \frac{d^2 x_i}{dt^2} = \frac{\vec{F}_i}{q_i} \\
\text{Computation of energy/forces:} & \quad \vec{F}_i = -\nabla_\vec{r}(V)
\end{align*}
\]

Lipid-double layer membrane

\[E \approx V = \sum_b k_b (r_b - r_{eq}^b)^2 + \sum_v k_v (\theta_v - \theta_{eq}^v)^2 + \sum_d k_d (1 - \cos(n_d \varphi_d - \delta_d)) + \sum_{i,j} \frac{q_i q_j}{4\pi \epsilon_{r_{ij}}} + \sum_{i,j} \frac{A_{ij}}{r_{ij}^6} - \frac{B_{ij}}{r_{ij}^8}\]

\(E \approx V\) represents the potential energy of the system, which is a sum of bond energies, valence angles, torsion angles, effective charges, and van der Waals interactions.

MOLMEC: straightforward, \(O(N^2)\)
MEGADYN: FMM approach, \(O(N)\)

FMM (Fast Multipole)
Greengard, Rokhlin
Separate short & long range forces:
- Short-range forces are updated in each time step
- Long-range forces are treated on "coarser scales"

<table>
<thead>
<tr>
<th>P</th>
<th>MOLMEC 7,000 atoms</th>
<th>MEGADYN 550,000 atoms</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8152 sec</td>
<td>6305 sec</td>
</tr>
<tr>
<td>2</td>
<td>4481 sec</td>
<td>3295 sec</td>
</tr>
<tr>
<td>3</td>
<td>3056 sec</td>
<td>1840 sec</td>
</tr>
<tr>
<td>4</td>
<td>2427 sec</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1769 sec</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1769 sec</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1769 sec</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1769 sec</td>
<td></td>
</tr>
</tbody>
</table>

Estim. time for 550,000 atoms: 1.5 years!
Basic Multigrid Principles

Smoothing &
Coarse-grid correction

Model Problem: Gauss-Seidel Relaxation

One Gauss-Seidel step:
\[ u \rightarrow \bar{u}: \quad -\bar{u}_{i-1,j} - \bar{u}_{i,j-1} + 4\bar{u}_{i,j} - u_{i+1,j} - u_{i,j+1} = h^2 f_{i,j} \]

or, in terms of the error:
\[ v \rightarrow \bar{v}: \quad -\bar{v}_{i-1,j} - \bar{v}_{i,j-1} + 4\bar{v}_{i,j} - v_{i+1,j} - v_{i,j+1} = 0 \]

lexicographic order:

Very slow convergence: \[ \rho = 1 - O(h^2) \]

Asymptotic convergence factor

Very fast „smoothing“ of the error:
\[ \frac{1}{4} \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} v_{ij}^h \approx v_{ij}^h \]

The smoother the error, the less efficient a further error reduction becomes!

Averaging of error
Principle of Error Smoothing

Relaxation methods converge slowly but smooth the error quickly!

Error = superposition of low and high frequencies

low frequencies (slow reduction)

high frequencies (fast reduction)

Infinite Grid Smoothing Analysis

Fourier components on infinite grid

\[ e^{i\Theta x / h} \quad (|\Theta| \leq \pi) \]

\[ |\Theta| := \max(|\Theta_1|, |\Theta_2|) \]

\[ \Theta x := \Theta_1 x_1 + \Theta_2 x_2 \]

Amplification factor per relaxation step

\[ e^{i\Theta x / h} \rightarrow \mu(\Theta)e^{i\Theta x / h} \]

Distinguish low & high frequencies

\[ e^{i\Theta x / h} \begin{cases} 
|\Theta| < \pi / 2 \quad \text{low (smooth) frequencies} \\
|\Theta| \geq \pi / 2 \quad \text{high (non-smooth) frequencies}
\end{cases} \]

Model problem, Gauss-Seidel

\[ \mu^* = 0.5, \quad h\text{-independent!} \]
Model Problem: Error Smoothing

Smoothing by Gauss-Seidel relaxation

Coarse-Grid Correction

Defect equation:
\[ d_h^m = f_h - A_h u_h^m \Rightarrow A_h v_h = d_h^m \Rightarrow u_h^* = u_h^m + v_h \]

Coarse-grid correction:
\[ d_h^m = f_h - A_h u_h^m \]
\[ d_h^m : = I_H^H d_h^m \] restriction
\[ A_H v_H = d_H^m \]
\[ u_{h+1}^m = u_h^m + \hat{v}_h \]
\[ \hat{v}_h : = I_H^h v_H \] interpolation

\[ M_{h,H} = I_h - I_H^h A_H^{-1} I_H^h A_h \]
\[ \rho(M_{h,H}) \geq 1 \] not full rank
Two-Grid Cycle

\[ v_1 \text{ smoothing steps} \]
\[ d_h^m = f_h - A_h u_h^m \]
\[ d_H^m := I_h^H d_h^m \quad \text{restriction} \]
\[ A_H v_H = d_H^m \]
\[ v_2 \text{ smoothing steps} \]
\[ u_h^{m+1} = u_h^m + \hat{v}_h \]
\[ \hat{v}_h := I_H^h v_H \quad \text{interpolation} \]

\[ M_{h,H} = S_h^{\nu_2} (I_h - I_H^h A_H^{-1} I_h^H A_h) S_h^{\nu_1} \]

\[ \rho(M_{h,H}) << 1 \quad \text{independent of } h! \]

---

Example: Model Problem

Standard coarsening: \( \Omega_h \rightarrow \Omega_{2h} \)

Smoothing: Gauss-Seidel relaxation

\[ L_h = \frac{1}{h^2} \begin{bmatrix} -1 & 1 & -1 \\ -1 & 4 & -1 \\ -1 & 1 & -1 \end{bmatrix}_h \]

\[ L_{2h} = \frac{1}{4h^2} \begin{bmatrix} -1 & 1 & -1 \\ -1 & 4 & -1 \\ -1 & 1 & -1 \end{bmatrix}_{2h} \]

Restriction: „full weighting“

\[ I_h^2 = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h \]

Interpolation: bilinear

\[ I_{2h}^h = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h \]
Multigrid Cycle

Recursive extension of two-grid cycle

Approximate solution by $\gamma$ two-grid cycles using still coarser grids, ..... 

V-Cycle $(\gamma=1)$

W-Cycle $(\gamma=2)$

MG Performance for Model Problem

Time per variable (msec)

Residual reduction: $\varepsilon = 10^{-12}$
General Remarks

What is multigrid not?
A particular solver

What is multigrid?
A general strategy for constructing hierarchical solvers

MG Components

- Smoothing process (type, number of steps, ...)
- Coarsening process (type of hierarchy, speed of coarsening, ...)
- Intergrid transfer processes (interpolation, restriction)
- Coarser-level operators
- Coarsest-level solver
- More advanced techniques
- ......
Advanced MG Techniques/Applications

Advanced techniques

• Full multigrid (FMG)
• Nonlinear multigrid (FAS)
• Local refinements
• Multigrid for eigenproblems
• Parallel multigrid
• Algebraic multigrid (AMG)

Applications

• General domains and BCs
• Variable coefficients
• Singular perturbed problems
• Discontinuous coefficients
• Systems of PDEs
• Non-elliptic PDEs
• Algebraic problems
• .........

Unstructured Grids
Full Multigrid (FMG)

Nested Iteration + MG = Full Multigrid (FMG)

\[
II = \text{FMG interpolation order} \geq \text{discretization order, model problem: cubic}
\]

\[
E_h = O(h^2)
\]

On each level, a fixed number of cycles, $\kappa$, is sufficient (typically, $\kappa = 1$ or $\kappa = 2$) $\implies O(N)$ complexity!
Nonlinear Multigrid (FAS)

\[ L[u](\bar{x}) = f(\bar{x}) \]

discrete:

\[
\begin{align*}
L_h[u_h](\bar{x}) &= f_h(\bar{x}) \\
A_h[u_h] &= f_h
\end{align*}
\]

Nonlinear Problems: Two Approaches

Straightforward approach

1. Global linearization (outer iteration)
2. Linear multigrid (inner iteration)

- Global Jacobian needs to be computed (and stored!)
- Inner and outer iterations have to be matched

Full approximation scheme (FAS)

1. Nonlinear relaxation (local linearization)
2. Nonlinear defect equation

- No global linearization (except for coarsest level?)
- Same cycle structure as in the linear case
- No matching of different iterations required
- Advantages also for linear problems
Nonlinear Relaxation (Local Linearization)

\[ i = 1, \ldots, N : \begin{cases} a_i[u_1, \ldots, u_{i-1}, \bar{u}_i, u_{i+1}, \ldots, u_N] = f_i & \text{Jacobi} \\ u_i \rightarrow \bar{u}_i \end{cases} \quad \begin{cases} a_i[\bar{u}_1, \ldots, \bar{u}_{i-1}, \bar{u}_i, u_{i+1}, \ldots, u_N] = f_i & \text{Gauss-Seidel (GS)} \end{cases} \]

Example for nonlinear GS

\[ L[u] = -\Delta u + g(x, y, u) = f \quad (g_u(x, y, u) \geq 0) \]

\[-u_{i-1,j} - u_{i,j-1} + 4u_{i,j} - u_{i+1,j} - u_{i,j+1} + h^2g(x_i, y_j, \bar{u}_{i,j}) = h^2f_{i,j} \]

GS-Picard: \[ g(x_i, y_j, u_{i,j}) \] (reasonable if \( h^2g_u < 1 \))

GS-Newton: \[ g(x_i, y_j, u_{i,j}) + g_u(x_i, y_j, u_{i,j})(\bar{u}_{i,j} - u_{i,j}) \]

Nonlinear Defect Equation

Linear defect equation: \[ A_h v_h = d_h^m \quad (= f_h - A_h u_h^m) \]

Nonlinear analog: \[ A_h [v_h + u_h^m] - A_h [u_h^m] = d_h^m \]

Coarse-grid approximation

\[ A_H [v_H + \overline{I_h}^H u_h^m] - A_H [\overline{I_h}^H u_h^m] = I_h^H d_h^m \]

\[ \overline{I_h}^H \] and \( I_h^H \) not necessarily the same!

Standard coarsening

Typical choice: \( \overline{I_h}^{2h} = \text{straight injection} \)
Full Approximation Scheme (FAS)

\[ d_h^m = f_h - A_h[u_h^m] \]

\[ d_H^m := I_h^H d_h^m \]

\[ f_H^m := d_h^m + A_h[T_h^H u_h^m] \]

\[ A_h[v_H + T_h^H u_h^m] = d_h^m + A_h[T_h^H u_h^m] \]

\[ \Rightarrow u_H = :f_H^m \]

\[ := u_{h} \]

\[ := u_{h+1} \]

\[ \nu_1 \text{ smoothing steps} \]

\[ \nu_2 \text{ smoothing steps} \]

\[ u_h^{m+1} = u_h^m + \hat{v}_h \]

\[ \text{Standard coarsening} \]

\[ \overline{I}_h^{2h} = \text{straight injection} \]

\[ u_h^m \rightarrow u_h^* \quad u_H \rightarrow T_h^H u_h^* \]

\[ u_{2h} \rightarrow u_h^* \]
Full Approximation Scheme (FAS)

Remarks

- If A linear:
  - FAS identical to linear cycle ("correction scheme")
  - however: different point of view
- Recursive extension to multigrid as in the linear case
- In general, continuation required
  - approach the range of attraction
  - most natural: combination with FMG
- Natural for solving eigenvalue problems

Example

\[-\Delta u + e^u = f (\Omega)\]
\[u = g (\Gamma)\]

<table>
<thead>
<tr>
<th>(m)</th>
<th>Method I</th>
<th>Method II</th>
<th>FAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.18(+2)</td>
<td>.18(+2)</td>
<td>.14(+2)</td>
</tr>
<tr>
<td>2</td>
<td>.29</td>
<td>.20</td>
<td>.20</td>
</tr>
<tr>
<td>3</td>
<td>.86(-2)</td>
<td>.55(-2)</td>
<td>.54(-2)</td>
</tr>
<tr>
<td>4</td>
<td>.14(-3)</td>
<td>.14(-3)</td>
<td>.14(-3)</td>
</tr>
<tr>
<td>5</td>
<td>.43(-5)</td>
<td>.42(-5)</td>
<td>.42(-5)</td>
</tr>
<tr>
<td>6</td>
<td>.13(-6)</td>
<td>.13(-6)</td>
<td>.13(-6)</td>
</tr>
<tr>
<td>7</td>
<td>.47(-8)</td>
<td>.39(-8)</td>
<td>.38(-8)</td>
</tr>
<tr>
<td>8</td>
<td>.13(-9)</td>
<td>.12(-9)</td>
<td>.12(-9)</td>
</tr>
<tr>
<td>9</td>
<td>.42(-11)</td>
<td>.40(-11)</td>
<td>.39(-11)</td>
</tr>
</tbody>
</table>

\[ \| \| u_h^* - u_h^m \|_2 \] as a function of \(m\)

**Method I:** Global Newton linearization, number of MG cycles doubled

**Method II:** Global Newton linearization, one MG cycle per Newton step
Continuation Methods

Bratu’s problem (bifurcation)

\[-Δu - λ e^u = 0 \quad (Ω = (0,1)^2)\]
\[u = 0 \quad (Γ)\]

Bifurcation diagram

Continuation via \(λ\):
gives only lower branch solutions (up to \(λ \sim 6.5\))

Other solutions require more sophisticated continuation techniques!

Continuation Methods

Augmented system: continuation parameter „s“

\[L[u,λ] = 0 \quad (Ω)\]
\[u = 0 \quad (Γ)\]

\[L[u(s),λ(s)] = 0 \quad (Ω)\]
\[u(s) = 0 \quad (Γ)\]
\[C[u(s),λ(s),s] = 0 \quad \text{(constraint)}\]

The constraint has also to be transferred to coarser grids in the sense of FAS

For example: „arclength continuation“

\[C[u,λ,s] = ||du/ds||_2^2 + |dλ/ds|^2 - 1 = 0\]

Simpler choices for Bratu:

\[C[u,λ,s] = ||u|| - s\]
\[C[u,λ,s] = u(0.5,0.5) - s\]
Eigenproblems

Here: computation of smallest EV/EF for elliptic s.p.d. problems

Eigenproblems + FAS

Problem: Find smallest EV/EF of

\[ L_h u_h - \lambda_h u_h = 0 \quad (=: f_h) \]

\[ \eta_h(u_h) = \sigma_h \]

One step of a relaxation method (inefficient as solver!):

1. GS relaxation step w.r.t. \( u_h \) (\( \lambda_h \) fixed): \( L_h u_h - \lambda_h u_h = f_h \)
2. Scale \( u_h \): \( \eta_h(u_h) = \sigma_h \)
3. Update of \( \lambda_h \) (\( u_h \) fixed): \( (L_h u_h, u_h) - \lambda_h(u_h, u_h) = (f_h, u_h) \)

Two multigrid approaches

(a) Replace (1) by linear MG cycle \( \Rightarrow \) inefficient

(b) Use (1-3) as smoother in (non-linear) FAS process (can be simplified, see later)
**Eigenproblems + FAS**

\[ L_H u_H - \lambda_H u_H = f_H \]

\[ \eta_H (u_H) = \sigma_H \]

**\( \nu_1 \) smoothing steps**

\[ a_h^{(L)} = f_h - (L_h u_h - \lambda_h u_h) \]

\[ d_h^{(n)} = \sigma_h - \eta_h (u_h) \]

**\( \nu_2 \) smoothing steps**

\[ u_h \leftarrow u_h + \hat{v}_h \]

\[ \lambda_h \leftarrow \lambda_H \]

\[ \hat{v}_h := I_H^h v_H \]

\[ v_H := u_H - I_h^H u_h \]

At convergence:

\[ u_H = I_h^H u_h, \quad \lambda_H = \lambda_h \]

If coarse enough, e.g.,

solution by relaxation (1-3)

**Remarks**

- Straightforward: recursive extension to more levels
- Solution on coarsest level: e.g. by relaxation (1-3)
- Generally sufficient:
  - \( \lambda \)-update only on coarsest level
  - Normalization only on coarsest level
- Natural combination with FMG
- No more expensive than MG for regular problems
- Generalization to several (many) eigenvalues
Local Refinement  
(MLAT)

Adaptivity

Adaptivity
grid resolution,
order of discretization,
type of discretization.

Adaptive grids
pre-defined (static),
self-adapting (dynamic).

Applications
locally non-smooth solutions
(boundary or interior layers, shocks, turbulence, ....),
non-smooth domains,
singularities / discontinuities in the differential problem.
Boundary-Driven Singularity

\[ -\Delta u = f \quad (\Omega) \]
\[ u = 0 \quad (\Gamma) \]

Singular behavior near \( P \):
\[ u_s(r,\phi) = r^{2/3} \sin\left(\frac{2}{3}\phi\right) \]
\( r,\phi = \) polar coordinates

Discretization error:
- \( O(h^{4/3}) \) at fixed distance from \( P \)
- \( O(h^{2/3}) \) at distance \( O(h) \) from \( P \)

Remedies:
1. \( u^* = \tilde{u} + u_s \), \( \tilde{u} \) smooth
2. local refinements

Global grid vs. adaptive grid

Global fine grid
- 49,665 points
- \( \| u^* - u_h \|_\infty = 3.3(-3) \)

Adaptive grid
- 657 points
- \( \| u^* - u_h \|_\infty = 3.8(-3) \)
Local Refinement

Natural integration into the MG approach

Composite grid ➔ Multigrid hierarchy of grids

Important aspects:

Dynamic local refinement
- Combination with FMG process

New aspects in MG cycling
- Treatment of interface boundaries
- Grids covering only subdomains ➔ FAS, even for linear problems!

Local Refinement + FAS

Recall: FAS correction equation

\[
L_{2h}[u_{2h}] = I_{2h}^{2h} d_m + L_{2h}^{2h} [I_{2h}^{2h} u_h^m]
\]

\[
= I_{2h}^{2h} f_h + L_{2h}^{2h} [I_{2h}^{2h} u_h^m] - I_{2h}^{2h} L_h[u_h^m]
\]

\[
= I_{2h}^{2h} f_h + \tau_{2h}^h
\]

(h,2h)-relative truncation error

\[
I_{2h}^{2h} = \text{straight injection}:
\]

\[
u_h^m \rightarrow u_h^* \quad \rightarrow \quad u_{2h} \rightarrow I_{2h}^{2h} u_h^* = u_h^*
\]
Local Refinement + FAS

\[ \text{Smoothing:} \]
Relaxation on grid $\Omega_h$: 
Interface variables are treated as Dirichlet points:

\[ \begin{align*}
\text{FAS correction:} \\
\text{At all points:} & \quad \bullet \quad \triangleleft \quad \triangle \quad \square \\
\end{align*} \]

\[ L_{2h} u_{2h} = \begin{cases} 
I_{2h} f_h + \tau_{2h} \triangleleft \\
            f_h \quad \triangle \quad \triangle
\end{cases} \]
Local Refinement

Modifications

- conservative interpolation
- conservative discretization along interface

Modified interface treatment

Conservative FV discretization along interface

Smoothing by relaxation now includes interface variables

Local Refinement

Requirements on automatic refinement criteria

- Reliably detect local low accuracy
- Terminate automatically (!)

A simple grid refinement criterion

\[
L_{2h} u_{2h} = \left\{ \begin{array}{l}
I_{h}^{2h} f_{h} + \tau_{h}^{2h} \\
\tau_{h}^{2h}
\end{array} \right.
\]

\(\tau_{h}^{2h}\): measures to which extent the fine grid solution is different from the coarse grid one.

refine where \( h^d \tau_{h}^{2h} \geq \varepsilon \) (\(d = \text{dimension}\)
Example: Euler Equations

Euler Equations
\[ \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0 \]
\[
\begin{bmatrix}
\rho u \\
\rho u^2 + p \\
\rho uv \\
(E + p)u
\end{bmatrix}
\quad \text{and} \quad
\begin{bmatrix}
\rho v \\
\rho uv \\
\rho v^2 + p \\
(E + p)v
\end{bmatrix}
\]
\[ p = (\gamma - 1)(E - \frac{1}{2}\rho(u^2 + v^2)) \]

Example: Euler Equations

Global block-structured grid
(subdivided for parallel processing)

NACA0012 airfoil:

\[ M_\infty = 0.85, \ \alpha = 1.0^\circ \quad M_\infty = 0.80, \ \alpha = 1.25^\circ \]
Example: Euler Equations

Refinement criterion (FE)

Refine near $Q$ if $r^h(Q) \geq \varepsilon$

$$r^h(Q) = \frac{2}{\sum_{i=1}^{4} \sum_{j=1}^{\Delta_i} |r_{ij}| dx}$$

assuming linear shape functions

Example: Euler Equations
Example: Euler Equations

Comparison: global grid vs. adaptive grid (obtained in parallel on 16 processors)

<table>
<thead>
<tr>
<th></th>
<th>Global fine grid</th>
<th>Adaptive grid</th>
<th>Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td># points</td>
<td>197,632</td>
<td>13,866</td>
<td>14</td>
</tr>
<tr>
<td>Time</td>
<td>6,115 sec</td>
<td>590 sec</td>
<td>10</td>
</tr>
</tbody>
</table>

Literature
**Historical Papers**

- **Discovery of multigrid (theoretical)**

- **Beginning of multigrid**

- **Re-discovery of multigrid**

**Text Books**

- **Classical**

- **Theory**

- **Tutorial-level**

- **Engineers**

- **Engineers and Practitioners**