STAB Workshop, Göttingen, 5th November 2019

LU-SGS preconditioned Newton-Krylov solver applied to industrial relevant test cases

Arpit Aggarwal, Ralf Hartmann
Overview

• Objective
• Description of solution methods
• Numerical results
• Outlook
DLR-TAU Code

- Finite volume method with node centered scheme on unstructured meshes.
- Available solvers/accelerators: single grid (SG), multigrid (FAS), full multigrid (FMG), grid sequencing (GSeq).
- Primarily used solver/smoothing is LU-SGS.
- TAU is used by various research institutions, universities, and industry (Airbus, …).

Objective of this work

Improvement of the robustness and the efficiency of the TAU code for stationary RANS simulations of various aircraft configurations.

- Improvement of the stability of the multigrid method.
- Improvement of the nonlinear multigrid smoother.
- Implementation of the developments in the central TAU version.
- Demonstration of the efficiency and robustness improvements of the TAU code on specific test cases.
Backward-Euler method and LUSGS iteration

Discretized flow equations:
\[ M \frac{dW}{dt} + R(W) = 0, \quad M = \text{diag}(vol(\Omega_i)) \]

Backward-Euler method:
\[ \left( \frac{M}{\Delta t} + J \right) \Delta W = -R(W), \quad W = W + \Delta W \]
with \( A = \frac{M}{\Delta t} + J \) and Jacobian matrix \( J = \frac{dR}{dW} \). Recovers Newton’s method for \( \Delta t \to \infty \).

LUSGS iteration:
Replace exact Jacobian \( J \) by approximate \( \tilde{J} \) including flux differences and eigenvalues:
\[ \tilde{A} = \frac{M}{\Delta t} + \tilde{J} = L + D + U \]
• Forward sweep: \( (D + L)\Delta W^* = -R(W) \)
• Backward sweep: \( (D + U)\Delta W = D\Delta W^* \)
Backward-Euler with LUSGS preconditioned GMRes

\[
\left( \frac{M}{\Delta t} + J \right) \Delta W = -R(W)
\]

- Solve \( Ax = b \) for \( x \) using the GMRes method.
- Use LUSGS (or an iterated version of it) as (linear) preconditioner of GMRes.

Matrix-free approximation to the Jacobian matrix:
Replace matrix-vector multiplication \( Jx \) by finite difference (FD) approximation:

\[
Jx \approx \frac{R(W + hx) - R(W)}{h}
\]

Advantages:
- (Hand-differentiated) Jacobian matrix not required
  - Significantly simpler implementation and lower memory requirements
- Same formula/code for all turbulence models

(Possible) disadvantage:
- FD gives an approximation only to \( J \). Is it sufficiently good?
- Influence of step size \( h \)
FD for matrix-free Jacobian matrix-vector multiplication

\[ Ax = \left( \frac{M}{\Delta t} + J \right) x \]

Approximation by forward difference:

\[ Ax \approx \frac{M}{\Delta t} x + \frac{R(W + hx) - R(W)}{h} \]

\[ h = \epsilon \cdot \frac{||W||_2}{||x||_2} \]

\[ \epsilon = 10^{-8} \]
Switch-Evolution Relaxation (SER)

\[ CFL = \min \left\{ CFL_{\text{max}}, CFL_{\text{init}} \times CFL_{\text{factor}} \times \left( \frac{\|R^n,\text{ref}\|_2}{\|R^n\|_2} \right)^\alpha \right\} \]

where

\[ \alpha = \begin{cases} 
1, & \text{2D laminar flows} \\
0.6, & \text{2D turbulent flows} \\
0.4, & \text{3D turbulent flows} 
\end{cases} \]

- \( CFL_{\text{max}} \) is the maximum allowed CFL value
- \( CFL_{\text{init}} \) is the initial CFL
- \( CFL_{\text{factor}} \) has default value = 1, might be reduced to \( \frac{1}{2} \), \( \frac{1}{4} \), ... in case of solver recovery
- \( \|R^n\|_2 \) is the \( l_2 \)-norm of the residual computed after the \( n^{th} \) nonlinear iteration
- \( \|R^n,\text{ref}\|_2 = \|R^\infty\|_2 \) is the \( l_2 \)-norm of the freestream residual on the (currently finest) grid
Residual Smoothing [Mavripilis]

- Smooth transition
  - from an explicit iteration scheme (known to provide good initial convergence)
  - to an increasingly implicit scheme (Backward Euler / Newton method).
- Ensure smooth residual field, thus producing smooth updates to the solution vector.

\[
\left( \frac{M}{\Delta t} + \frac{\partial R(W^n)}{\partial W} \right) \Delta W^n = -R(W^n)
\]

\[
\left( \frac{M}{\Delta t} + \frac{\partial R(W^n)}{\partial W} \right) \Delta W^n = -R(W^n) - P^{-1} \frac{M}{\Delta t} R(W^n)
\]

\[
\left( \frac{M}{\Delta t} + \frac{\partial R(W^n)}{\partial W} \right) \Delta W^n = - \left[ I + P^{-1} \frac{M}{\Delta t} \right] R(W^n)
\]

- Left hand side Jacobian of the systems remains identical.
  - implementation is relatively straight-forward.
- Choice of preconditioners (explicit or slightly implicit iteration schemes)
  - Runge-Kutta, **LU-SGS, SGS**, nonlinear multigrid.

Backward-Euler as SG solver or FAS smoother

- **Single grid, SG**
- **Grid sequencing, GSeq**
- **Nonlinear multigrid**
  - Full approximation scheme, FAS
- **Full multigrid, FMG (= GSeq x FAS)**
Complex geometry, SA-neg: Grid 2 (~63e6 points) mesh
Complex geometry, SA-neg: Case description

<table>
<thead>
<tr>
<th>Flow equations</th>
<th>RANS</th>
<th>CFL ramping</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>CFL_max</td>
</tr>
<tr>
<td>Turbulence model</td>
<td>$SA - neg$</td>
<td>α</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flow Conditions</th>
<th>Residual tolerances</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>fine grid</td>
</tr>
<tr>
<td></td>
<td>$10^{-11}$ (Grid 1)</td>
</tr>
<tr>
<td></td>
<td>$10^{-8}$ (Grid 2)</td>
</tr>
<tr>
<td>Reynolds Number (Re)</td>
<td>$3.4 \cdot 10^6$</td>
</tr>
<tr>
<td></td>
<td>coarse grids (in multigrid)</td>
</tr>
<tr>
<td></td>
<td>$10^{-3}$</td>
</tr>
<tr>
<td>Angle of Attack (AoA)</td>
<td>$20^\circ$</td>
</tr>
<tr>
<td></td>
<td>linear solver (GMRes)</td>
</tr>
<tr>
<td></td>
<td>$10^{-8}$</td>
</tr>
<tr>
<td>Mach number (M)</td>
<td>0.15</td>
</tr>
<tr>
<td></td>
<td>LUSGS, SGS</td>
</tr>
<tr>
<td>Preconditioner</td>
<td></td>
</tr>
</tbody>
</table>
Complex geometry, SA-neg:

<table>
<thead>
<tr>
<th>Grid 1 (~23e6 points)</th>
<th>Grid 2 (~63e6 points)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{CFL}_{\text{init}} = 1.2 )</td>
<td>( \text{CFL}_{\text{init}} = 1.2 )</td>
</tr>
<tr>
<td>( \text{FMGV5} )</td>
<td>( \text{FMGV5} )</td>
</tr>
<tr>
<td><strong>Baseline TAU</strong></td>
<td><strong>Baseline TAU</strong></td>
</tr>
<tr>
<td><strong>Iterations till convergence</strong></td>
<td><strong>Iterations till convergence</strong></td>
</tr>
<tr>
<td>Baseline TAU</td>
<td>Stalled (1e-4)</td>
</tr>
<tr>
<td>20, 80</td>
<td>Stalled (1e-4)</td>
</tr>
<tr>
<td>50, 200</td>
<td>519 (SGS 3)</td>
</tr>
<tr>
<td>100, 400</td>
<td>487</td>
</tr>
<tr>
<td>200, 800</td>
<td>447</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Complex geometry, SA-neg: Grid 1 (~23e6 points)
Complex geometry, SA-neg: Grid 1 (~23e6 points)
Complex geometry, SA-neg: Grid 2 (~63e6 points)
Complex geometry, SA-neg: Grid 2 (~63e6 points)
Complex geometry, SA-neg: Grid 2 (~63e6 points)
VFE-2, SA-neg: Grid 4 (~10e6 points): mesh
# VFE-2, SA-neg: Case description

<table>
<thead>
<tr>
<th>Flow equations</th>
<th>Reynolds Number (Re)</th>
<th>Angle of Attack (AoA)</th>
<th>Mach number (M)</th>
<th>CFL ramping</th>
<th>Residual tolerances</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>RANS</em></td>
<td>59.5 ( \cdot 10^6 )</td>
<td>24.7°</td>
<td>0.869</td>
<td><em>CFL</em> (_{max})</td>
<td><em>fine grid</em> (10^{-10})</td>
</tr>
<tr>
<td>Turbulence model</td>
<td>SA – neg</td>
<td></td>
<td></td>
<td>(\alpha) 0.4</td>
<td><em>coarse grids (in multigrid)</em> (10^{-3})</td>
</tr>
</tbody>
</table>

**Preconditioner** 
*LU*SGS
<table>
<thead>
<tr>
<th>Grid 1 (~208K points) FMGV3</th>
<th>Grid 2 (~802K points) FMGV3</th>
<th>Grid 3 (~4.6e6 points) FMGV5</th>
<th>Grid 4 (~10e6 points) FMGV5</th>
<th>Grid 5 (~30.9e6 points) FMGV5</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFL_{init}=1</td>
<td>CFL_{init}=1</td>
<td>CFL_{init}=1</td>
<td>CFL_{init}=1</td>
<td>CFL_{init}=1</td>
</tr>
<tr>
<td>50 Krylov vectors, 200 GMRes iterations</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Baseline TAU</th>
<th>No of iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>No smoothing</td>
<td>393</td>
</tr>
<tr>
<td>LUSGS smoother</td>
<td>278</td>
</tr>
<tr>
<td>SGS 3 smoother</td>
<td>263</td>
</tr>
<tr>
<td>SGS 6 smoother</td>
<td>182 (diverged)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No of iterations</th>
<th>647</th>
<th>1854</th>
<th>19594</th>
<th>45670</th>
</tr>
</thead>
<tbody>
<tr>
<td>586</td>
<td>438</td>
<td>568</td>
<td>1819</td>
<td>11140</td>
</tr>
<tr>
<td>677</td>
<td>532</td>
<td>10207</td>
<td>4964</td>
<td></td>
</tr>
<tr>
<td>1232</td>
<td>729</td>
<td>13133</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Residual smoothing helps in all the computations.
- Smoother best practice: unclear.
VFE-2, SA-neg: Grid 1 (~208k points): Residual smoothing
VFE-2, SA-neg: Grid 2 (~802k points): Residual smoothing
VFE-2, SA-neg: Grid 3 (~4.6e6 points): Residual smoothing
VFE-2, SA-neg: Grid 4 (~10e6 points): Residual smoothing
VFE-2, SA-neg: Grid 5 (~30.9e6 points): Residual smoothing

![Graph 1](image1)

![Graph 2](image2)

- **Baseline**
- **No smoothing**
- **LUSGS**
- **SGS3**
- **SGS6**

**X-axis:** Nonlinear iterations

**Y-axis:** Density residual

**X-axis:** Wall-clock time (min)

**Y-axis:** Density residual
Outlook

• Integration into the TAU central version.

• Application to further test cases.

• Extension to the $k - \omega$ and RSM turbulence models.
Thank You
Questions ?