Efficient multigrid algorithms for higher order Discontinuous Galerkin discretizations of turbulent flows

Marcel Wallraff, Ralf Hartmann, Tobias Leicht
DLR Braunschweig (AS - C²A²S²E)
Multigrid
DG discretization

Basis functions

- non-parametric ortho-normal basis functions
- directly formulated in physical space
- also referred to as Taylor-DG
- need to be evaluated for each mesh element

RANS-$k\omega$ equations

- $k\omega$ turbulence model
- second scheme of Bassi and Rebay (BR2) for the viscous terms
- Roe flux as a convective flux
Non-linear multigrid method

nested hierarchy of linear spaces

\[ \mathbf{V}_{l_{\min}} \subset \mathbf{V}_{l_{\min}+1} \subset \cdots \subset \mathbf{V}_{l_{\max}-1} \subset \mathbf{V}_{l_{\max}} \]

\[ \mathbb{R}^{n_{l_{\min}}} \subset \mathbb{R}^{n_{l_{\min}+1}} \subset \cdots \subset \mathbb{R}^{n_{l_{\max}-1}} \subset \mathbb{R}^{n_{l_{\max}}} \]

intergrid transfer operators:

- prolongation: natural injection \( I_{l_{-1}}^{l} : \mathbb{R}^{n_{l_{-1}}} \rightarrow \mathbb{R}^{n_{l}} \)

- canonical restriction operator \( I_{l_{-1}}^{l-1} := \left( I_{l_{-1}}^{l} \right)^{\top} \)

non-linear multigrid algorithm also requires:

- restricted non-linear state vector:
  orthogonal \( L^2 \)-projection \( \hat{I}_{l_{-1}}^{l-1} \) on the space \( \mathbf{V}_{l_{-1}} \)
Non-linear multigrid method

Let the non-linear problem to be solved on the fine level $l_{\text{max}}$ be given by

$$L_{l_{\text{max}}} (u_{l_{\text{max}}}) = f_{l_{\text{max}}}.$$ 

- restrict solution approximation $u_{l-1} := \hat{1}_{l}^{-1} u_l$
- compute forcing function for the coarse level:

$$f_{l-1} \leftarrow f_{l-1} + \hat{1}_{l}^{-1} (f_l - L_l(u_l)) - \left( f_{l-1} - L_{l-1}(u_{l-1}^0) \right)$$

- Galerkin-transfer for the Jacobian: $R_{l-1} = \hat{1}_{l}^{-1} R_l \hat{1}_{l-1}$
Non-linear smoother / solver

- linearized Backward-Euler
- Solve \[ (\alpha_i \Delta t)^{-1} M + R_l \] \( (u_{l,i} - u_{l,i-1}) = \left[ f_l - L_l(u_{l,i-1}) \right] \), where \( R_l \) is the fully implicit Jacobian matrix and \( M \) is the mass matrix. In addition to that \( u_{l,j} \) is a state vector, with \( u_{l,j} \in V_l \ \forall \ j \in \mathbb{N} \).
- local pseudo-time steps, adaptive CFL number
Linear smoother / solver

- Krylov method as linear solver (GMRES method)
- line-Jacobi as preconditioner / smoother
  - let $L_{l,k}(u_{l,k}) = f_{l,k}$ the underlying linear problem on line $k$,
  - solve $\delta u_{l,k,i} := u_{l,k,i} - u_{l,k,i-1} = R_{l,k}^{-1}(f_{l,k} - L_{l,k}u_{l,k,i-1})$
  - set $u_{l,k,i} := u_{l,k,i-1} + \delta u_{l,k,i}$
  - where $R_{l,k}^{-1}$ is the inverse of the Jacobian matrix computed one line $k$ in the mesh
Relaxation scheme

Jacobian / system matrix structure

- element diagonal
- line neighbor
- off-line off-diagonal

matrix blocks
Numerical algorithms

possible solver choices

- single grid Backward-Euler
- start up strategy in mesh or order sequencing for improved initial conditions
- linear MG as preconditioner
- non-linear MG to accelerate process in pseudo-time
- non-linear MG with linear MG on each level
Numerical parameters for the non-linear problems

**non-linear multigrid**

- only V-cycles will be presented
- one pre- and post-smoothing iteration on each level
- one smoothing iteration on the lowest level
- a linearized Backward-Euler scheme as smoother
- using an SER time stepping scheme for the Backward-Euler
- Galerkin-transfer to obtain the Jacobian on the lower levels
Numerical parameters for the linear problems

parameters for solving the resulting linear problems on every level from the Backward-Euler linearization

- GMRES method with a fixed number of max steps on every level
- linear multigrid as a preconditioner for the GMRES method
- four smoothing iterations
- line-Jacobi scheme as smoother
- Galerkin-transfer to obtain the lower level matrices
L1T2 high-lift configuration

- Mach: 0.197
- Reynolds number: 3,520,000
- $\alpha = 20.18^\circ$
- Testcase from EC funded ADIGMA project.
L1T2 high-lift configuration

run time comparison: $p$-MG

---

density residual

normalized CPU time

- SG
- $p$-sequence SG
- $p$-NMG
- $p$-LMG
- $p$-NMG + LMG
L1T2 high-lift configuration

run time comparison: $h$-MG

![Graph showing normalized CPU time and density residual for different configurations.](image-url)
VFE-2 Delta-Wing with rounded leading edge

- Mach: 0.4
- Reynolds number: 3,000,000
- $\alpha = 13.3^\circ$, $\beta = 0^\circ$
- Testcase from EC funded IDIHOM project.

- Top: geometry/mesh
- Left: Surface pressure plot of a $\rho = 2$ solution on an adjoint-based refined mesh with 23877 elements.
VFE2 Delta-Wing with rounded leading edge

run time comparison: $p$-MG

![Graph showing normalized CPU time vs. density residual for different p-sequence methods.](image-url)
VFE2 Delta-Wing with rounded leading edge

run time comparison: $h$-MG

![Graph showing run time comparison for different $h$-MG methods.]

- $h$-sequence SG
- $h$-NMG
- $h$-NMG+LMG
- $h$-NMG+LMG (CFL ↑)

The graph compares the density residual over normalized CPU time for different methods, showing the efficiency and convergence of each method.
VFE2 Delta-Wing with rounded leading edge

top level time comparison

![Graph showing normalized CPU time vs density residual for different methods.]

- $h$-NMG+LMG (1e-12)
- $h$-NMG+LMG
- $h$-NMG+LMG (CFL ↑)
- $p$-NMG+LMG (1e-12)
- $p$-NMG+LMG
- $p$-NMG+LMG (CFL ↑)
VFE-2 Delta-Wing with rounded leading edge

The blue computation on the mesh with 13816 elements and the magenta computation on an adjoint refined mesh with 23877 elements.
Conclusion

- Development of a nonlinear p-multigrid algorithm for turbulent flows (as proposed for DGHPOPT) was successful
- Additionally, we implemented
  - a linear p-multigrid as preconditioner of a Backward-Euler smoothing step,
  - and the corresponding nonlinear and linear h-multigrid algorithms based on agglomeration