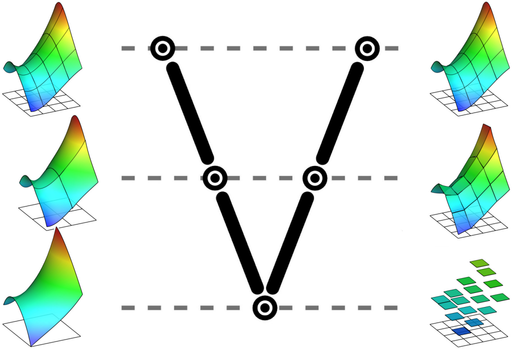
Higher Order Multigrid Algorithms for a 2D and 3D RANS-$k\omega$ DG-Solver

Marcel Wallraff, Tobias Leicht
DLR Braunschweig (AS - C²A²S²E)

DLR

Multigrid algorithms



DG discretization

Basis functions

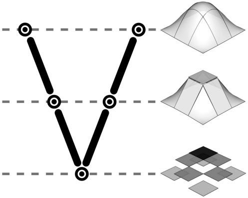
- non-parametric ortho-normal basis functions
- directly formulated in physical space
- also referred to as Taylor-DG
- need to be evaluated for each mesh element

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, based on an eigen-decomposition of the full jacobian

DLR

Non-linear multigrid method



nested hierarchy of linear spaces

$$\underset{\mathbb{R}^{n_{l_{\min}}}}{\mathbf{V}_{l_{\min}}} \subset \underset{\mathbb{R}^{n_{l_{\min}+1}}}{\mathbf{V}_{l_{\min}+1}} \subset \underset{\cdots}{\cdots} \subset \underset{\mathbb{R}^{n_{l_{\max}-1}}}{\mathbf{V}_{l_{\max}-1}} \subset \underset{\mathbb{R}^{n_{l_{\max}}}}{\mathbf{V}_{l_{\max}}}$$

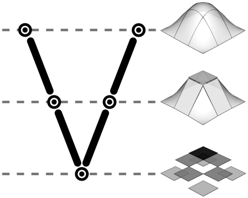
intergrid transfer operators:

- prolongation: natural injection $I_{l-1}^{l} : \mathbb{R}^{n_{l-1}} \to \mathbb{R}^{n_l}$
- canonical restriction operator $I_{l}^{l-1} := \left(I_{l-1}^{l}\right)^{\top}$

non-linear multigrid algorithm also requires:

- restricted nonlinear state vector:
  orthogonal L^2-projection $\hat{I}_{l}^{l-1}$ on the space $\mathbf{V}_{l-1}$

Non-linear multigrid method



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

$$\mathbf{L}_{l_{\max}}(\mathbf{u}_{l_{\max}}) = \mathbf{f}_{l_{\max}}.$$

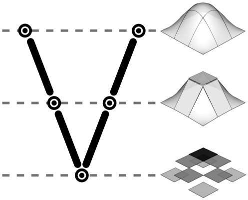
- restrict solution approximation $\mathbf{u}_{l-1} := \hat{I}_l^{l-1}\mathbf{u}_l$
- compute forcing function for the coarse level:

$$\mathbf{f}_{l-1} \leftarrow \mathbf{f}_{l-1} + I_l^{l-1}\left(\mathbf{f}_l - \mathbf{L}_l(\mathbf{u}_l)\right) - \left(\mathbf{f}_{l-1} - \mathbf{L}_{l-1}(\mathbf{u}_{l-1}^0)\right)$$

- Galerkin-transfer for the Jacobian: $\underline{R}_{l-1} = I_l^{l-1}\ \underline{R}_l\ I_{l-1}^l$

DLR

Non-linear smoother / solver



smoother / solver

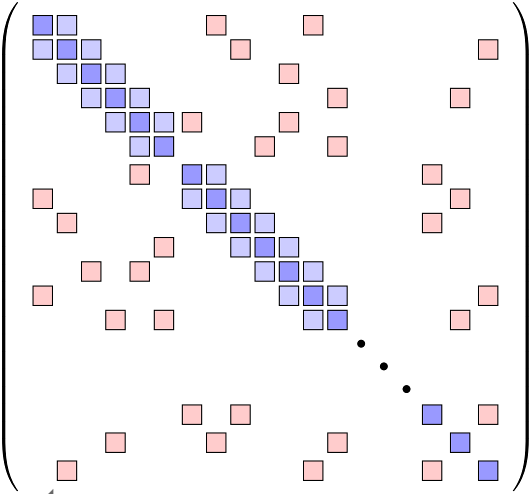
- linearized Backward-Euler
  - Solve $\left[(\alpha_i \Delta t)^{-1}\underline{\mathrm{M}} + \underline{\mathrm{R}}_l\right](\mathbf{u}_{l,i} - \mathbf{u}_{l,i-1}) = \left[\mathbf{f}_l - \mathbf{L}_l(\mathbf{u}_{l,i-1})\right]$,
  - where $\underline{\mathrm{R}}_l$ is the fully implicit Jacobian matrix and $\underline{\mathrm{M}}$ is the mass matrix. In addition to that $\mathbf{u}_{l,j}$ is a state vector, with $\mathbf{u}_{l,j} \in \mathbf{V}_l \; \forall \, j \in \mathbb{N}$.
- local pseudo-time steps, adaptive CFL number

Linear smoother / solver

- Krylov method as linear solver (GMRES method)
- line-Jacobi as preconditioner / smoother
  - let $\mathcal{L}_{l,k}(\mathfrak{u}_{l,k}) = \mathfrak{f}_{l,k}$ the underlying linear problem on line k,
  - solve $\delta\mathfrak{u}_{l,k,i} := \mathfrak{u}_{l,k,i} - \mathfrak{u}_{l,k,i-1} = \mathrm{R}_{l,k}^{-1}(\mathfrak{f}_{l,k} - \mathcal{L}_{l,k}\mathfrak{u}_{l,k,i-1})$
  - set $\mathfrak{u}_{l,k,i} := \mathfrak{u}_{l,k,i-1} + \delta\mathfrak{u}_{l,k,i}$,
  - where $\mathrm{R}_{l,k}^{-1}$ is the inverse of the Jacobian matrix computed one line k in the mesh

Relaxation scheme

Jacobian / system matrix structure



matrix blocks

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

DLR

Numerical algorithms

possible solver choices

- single grid Backward-Euler
- start up streategy 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 steping scheme for the Backward-Euler
- Galerkin-transfer to obtain the Jacobian on the lower levels

Numerical parameters for the linear problems

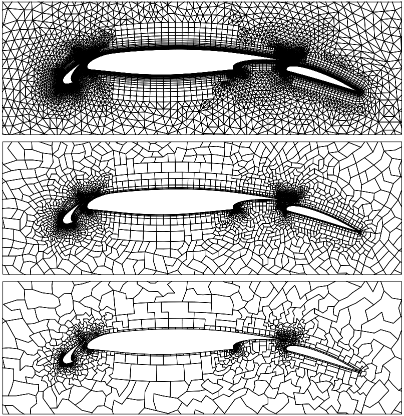
parameters for solving the resulting linear problems on every level from the Backward-Euler linearization

- GMRES method with a fixed number of max steps on every level
- linear multigrid as a preconditioner for the GMRES method
- one V-cycle is done for preconditioning
- no pre-smoothing iterations
- four post-smoothing iterations
- four smoothing iterations on the lowest level
- line-Jacobi scheme as smoother
- Galerkin-transfer to obtain the lower level matrices

L1T2 high-lift configuration

- flow conditions:
  - Mach: 0.197
  - Reynolds number: 3,520,000
  - $\alpha = 20.18^\circ$
- testcase from EC funded ADIGMA project
- computations will be shown on an unstructured mesh with 23824 mixed elements
- $p = 2$ solution is desired for all computations

L1T2 high-lift configuration

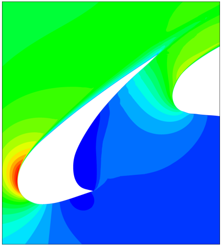


mixed-element mesh (CENAERO via GMSH)

third order

142 944 DoF

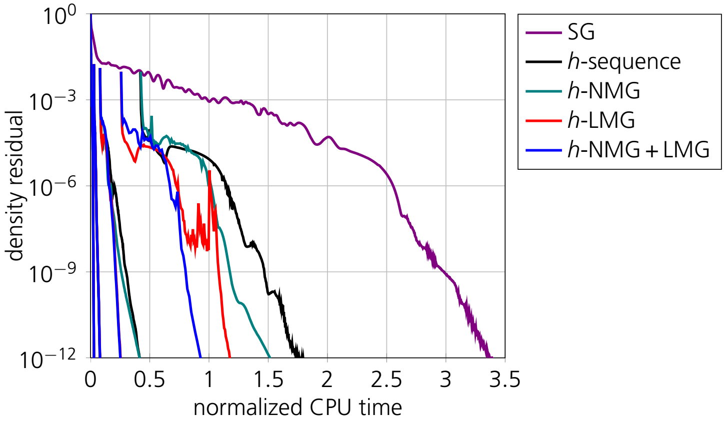
three levels



DLR

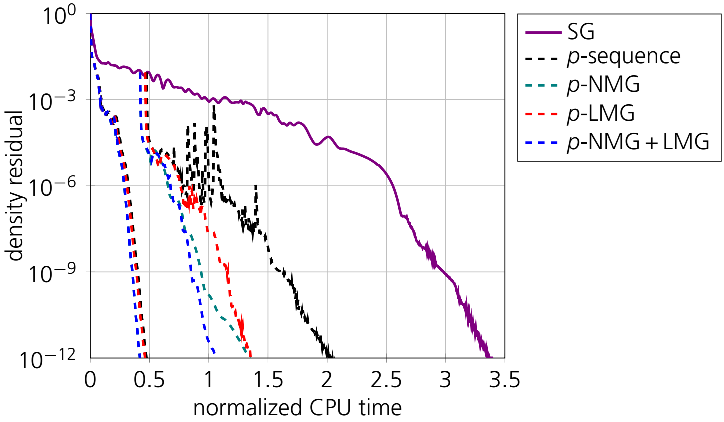
L1T2 high-lift configuration

run time comparison: *h*-MG



L1T2 high-lift configuration

run time comparison: *p*-MG

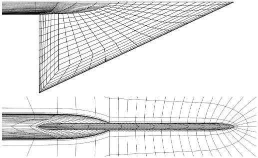


VFE-2 Delta-Wing with rounded leading edge

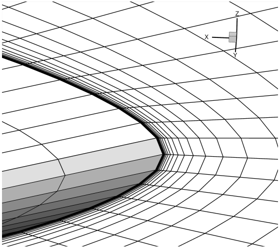
- flow conditions:
  - Mach: 0.4
  - Reynolds number: 3,000,000
  - $\alpha = 13.3°,\ \beta = 0°$
- testcase from EC funded IDIHOM project
- computations will be shown on a structured mesh with 13816 elements
- $p = 2$ solution is desired for all computations

VFE-2 Delta-Wing with rounded leading edge

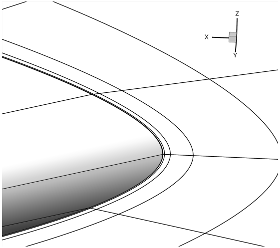
- original FV mesh with 884 224 elements was two times agglomerated
- resulting in a higher order mesh with 13 816 curved elements



geometry

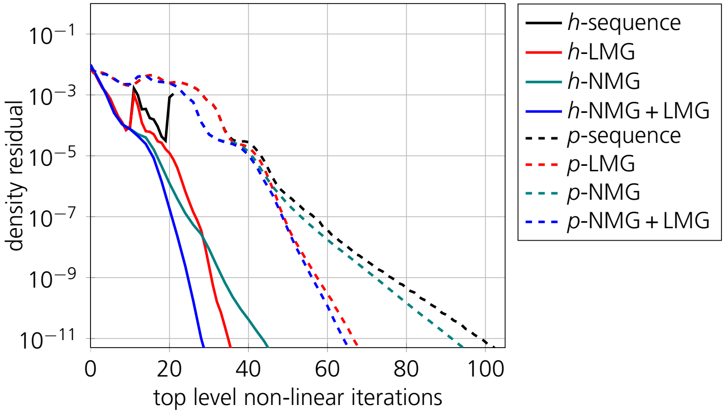


original mesh
with straight faces

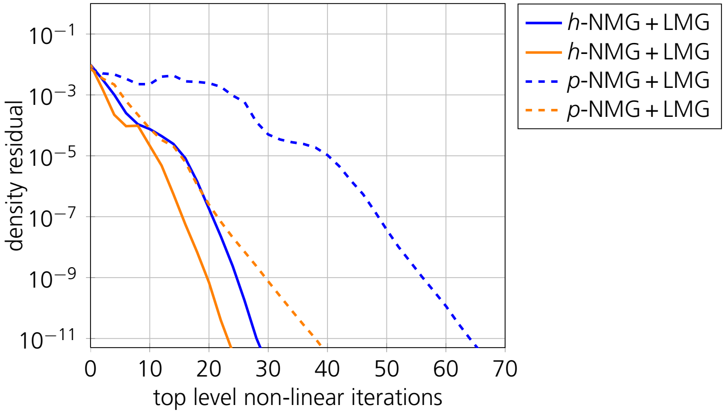


curved mesh with faces
represented by polynomials of order 4

VFE-2 Delta-Wing with rounded leading edge

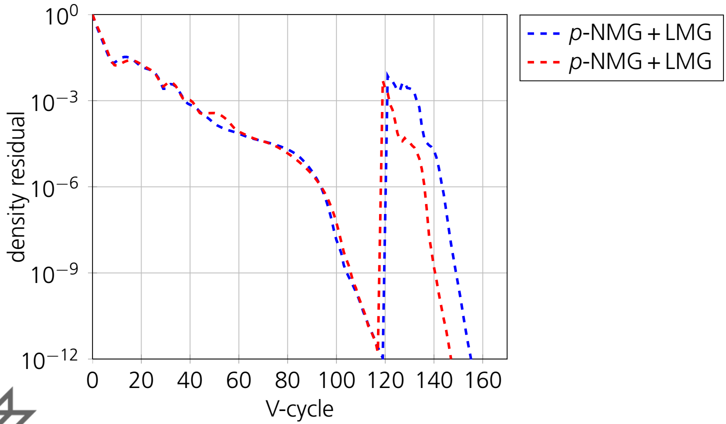


VFE-2 Delta-Wing with rounded leading edge



VFE-2 Delta-Wing with rounded leading edge

The *blue* computation on the mesh with 13816 elements and the *red* computation on an adjoint refined mesh with 23877 elemtents.



Thanks for your attention!