Space-time Discontinuous Galerkin Methods for Compressible Flows

Jaap van der Vegt

Numerical Analysis and Computational Mechanics Group
Department of Applied Mathematics
University of Twente

Joint Work with

Christiaan Klaij (UT), Daniel Köster (UT), Harmen van der Ven (NLR)
and Marc van Raalte (CWI)

Workshop Méthodes Numériques pour les Fluides

Paris, December 18, 2007

Space-Time Discontinuous Galerkin Finite Element Methods

Motivation of research:

- In many applications one encounters moving and deforming flow domains:
  - Aerodynamics: helicopters, manoeuvering aircraft, wing control surfaces
  - Fluid structure interaction
  - Two-phase and chemically reacting flows with free surfaces
  - Water waves, including wetting and drying of beaches and sand banks
- A key requirement for these applications is to obtain an accurate and conservative discretization on moving and deforming meshes

Motivation of Research

Other requirements

- Improved capturing of vortical structures and flow discontinuities, such as shocks and interfaces, using hp-adaptation.
- Capability to deal with complex geometries.
- Excellent computational efficiency for unsteady flow simulations.

These requirements have been the main motivation to develop a space-time discontinuous Galerkin method.

Overview of Lecture

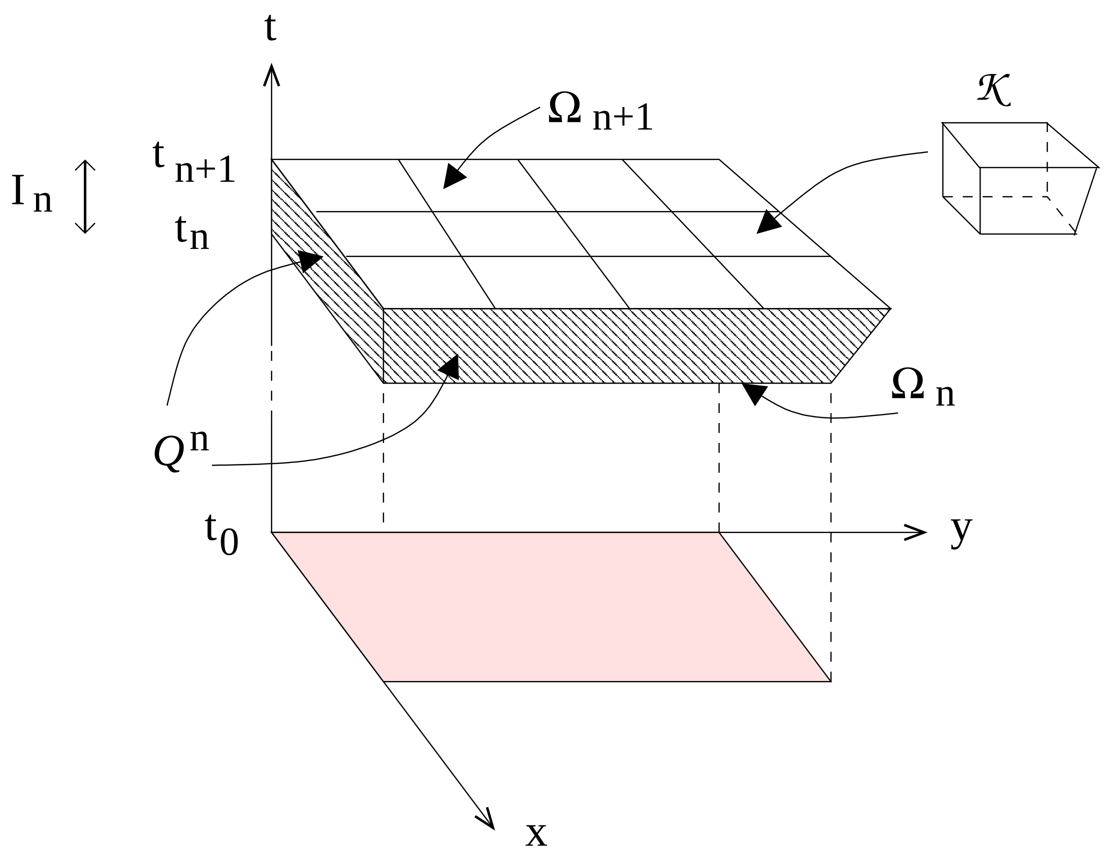
- Space-time discontinuous Galerkin finite element discretization for the compressible Navier-Stokes equations
  - main aspects of discretization
  - efficient solution techniques
- Applications in aerodynamics
- Concluding remarks

Space-Time Approach

Key feature of a space-time discretization

- A time-dependent problem is considered directly in four dimensional space, with time as the fourth dimension

Space-Time Domain



Sketch of a space-time mesh in a space-time domain.

Benefits of Space-Time Approach

A space-time discretization of time-dependent problems has as main benefits

- The problem is transformed into a steady state problem in space-time which makes it easier to deal with time dependent boundaries. No extrapolation or interpolation of (boundary) data
- A conservative numerical discretization is obtained on deforming and locally refined meshes

Compressible Navier-Stokes Equations

- Compressible Navier-Stokes equations in space-time domain $\mathcal{E} \subset \mathbb{R}^4$:

$$\frac{\partial U_i}{\partial x_0} + \frac{\partial F_k^e(U)}{\partial x_k} - \frac{\partial F_k^v(U, \nabla U)}{\partial x_k} = 0$$

- Conservative variables $U \in \mathbb{R}^5$ and inviscid fluxes $F^e \in \mathbb{R}^{5\times 3}$

$$U = \begin{bmatrix} \rho \\ \rho u_j \\ \rho E \end{bmatrix}, \qquad F_k^e = \begin{bmatrix} \rho u_k \\ \rho u_j u_k + p\delta_{jk} \\ \rho h u_k \end{bmatrix}$$

Compressible Navier-Stokes Equations

- Viscous flux $F^v \in \mathbb{R}^{5\times 3}$

$$F_k^v = \begin{bmatrix} 0 \\ \tau_{jk} \\ \tau_{kj}u_j - q_k \end{bmatrix}$$

with the total stress tensor $\tau \in \mathbb{R}^{3\times 3}$ defined as:

$$\tau_{jk} = \lambda \frac{\partial u_i}{\partial x_i}\delta_{jk} + \mu(\frac{\partial u_j}{\partial x_k} + \frac{\partial u_k}{\partial x_j})$$

and the heat flux vector $q \in \mathbb{R}^3$ given by:

$$q_k = -\kappa \frac{\partial T}{\partial x_k}$$

Compressible Navier-Stokes Equations

- The viscous flux F^v is homogeneous with respect to the gradient of the conservative variables ∇U:

$$F^v_{ik}(U, \nabla U) = A_{ikrs}(U)\frac{\partial U_r}{\partial x_s}$$

with the homogeneity tensor $A \in \mathbb{R}^{5\times 3\times 5\times 3}$ defined as:

$$A_{ikrs}(U) := \frac{\partial F^v_{ik}(U, \nabla U)}{\partial(\nabla U)}$$

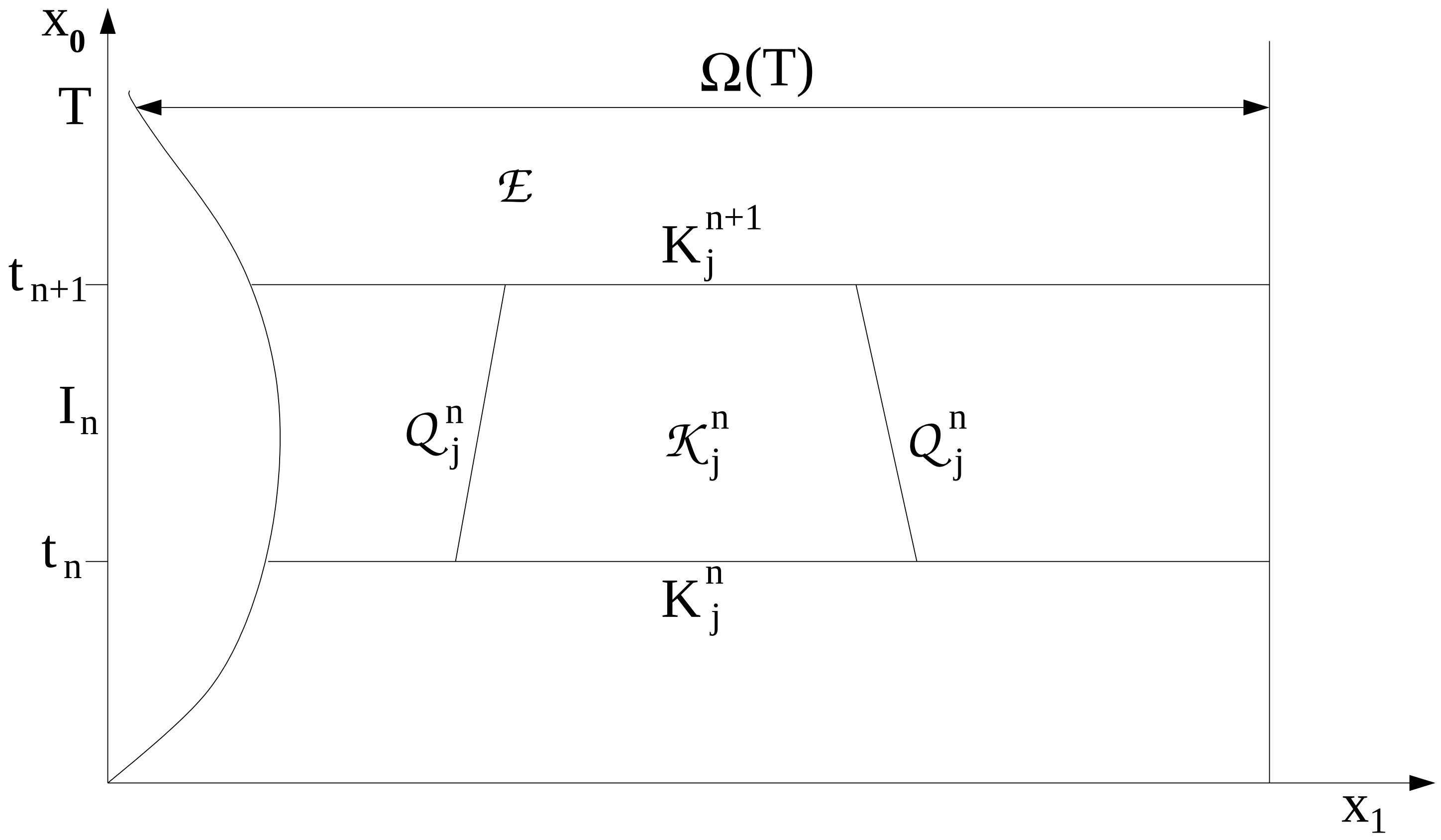
- The system is closed using the equations of state for an ideal gas.

Space-Time Discontinuous Galerkin Discretization

Main features of a space-time DG approximation

- Basis functions are discontinuous in space and time
- Weak coupling through numerical fluxes at element faces
- Discretization results in a coupled set of nonlinear equations for the DG expansion coefficients

Space-Time Slab



Space-time slab with elements in a space-time domain.

Benefits of Space-Time DG Discretization

Main benefits of a space-time DG approximation

- The space-time DG method results in a very local discretization, which is beneficial for:
  - hp-mesh adaptation
  - parallel computing
- The space-time discretization is conservative on moving and deforming meshes and also on locally adapted meshes

Discontinuous Finite Element Approximation

Approximation spaces

- The finite element space associated with the tessellation $\mathcal{T}_h$ is given by:

$$W_h := \left\{ W \in \left(L^2(\mathcal{E}_h)\right)^5 \; : \; W|_{\mathcal{K}} \circ G_{\mathcal{K}} \in (P^k(\hat{\mathcal{K}}))^5, \quad \forall \mathcal{K} \in \mathcal{T}_h \right\}$$

- We will also use the space:

$$V_h := \left\{ V \in \left(L^2(\mathcal{E}_h)\right)^{5\times 3} \; : \; V|_{\mathcal{K}} \circ G_{\mathcal{K}} \in (P^k(\hat{\mathcal{K}}))^{5\times 3}, \quad \forall \mathcal{K} \in \mathcal{T}_h \right\}.$$

- Note the fact that $\nabla_h W_h \subset V_h$ is essential for the discretization.

First Order System

- Rewrite the compressible Navier-Stokes equations as a first-order system using the auxiliary variable Θ:

$$\frac{\partial U_i}{\partial x_0} + \frac{\partial F^e_{ik}(U)}{\partial x_k} - \frac{\partial \Theta_{ik}(U)}{\partial x_k} = 0,$$

$$\Theta_{ik}(U) - A_{ikrs}(U)\frac{\partial U_r}{\partial x_s} = 0.$$

Weak Formulation

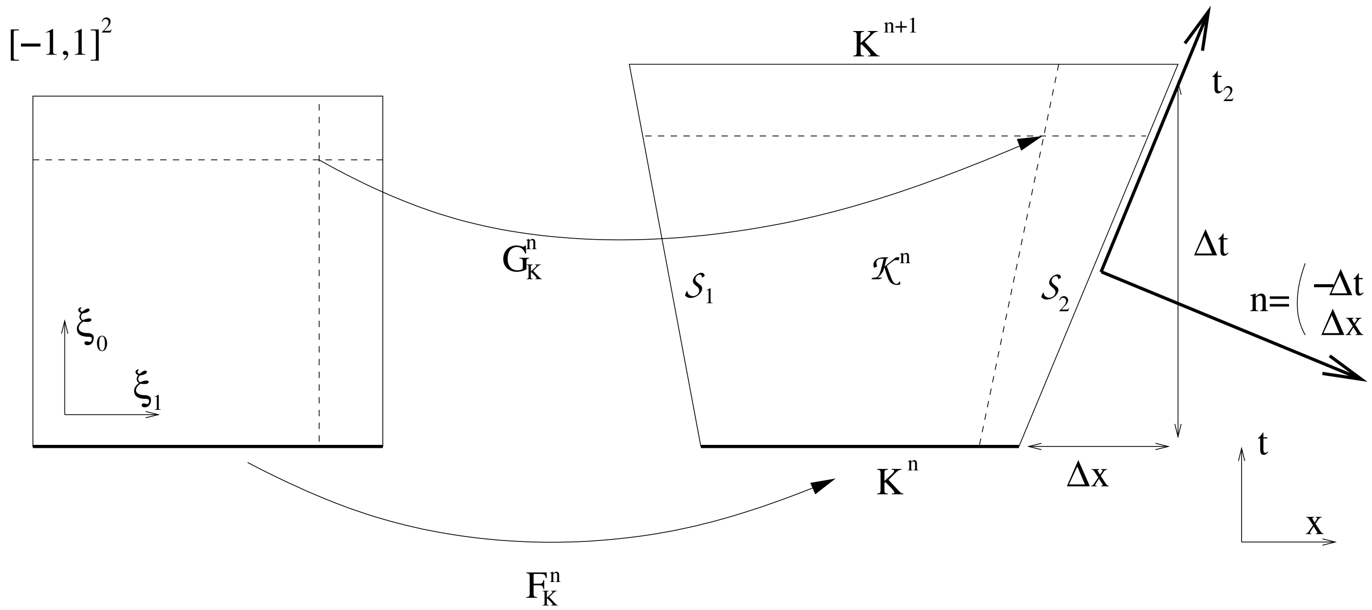
- Weak formulation for the compressible Navier-Stokes equations

  Find a $U \in W_h$, $\Theta \in V_h$, such that for all $W \in W_h$ and $V \in V_h$, the following holds:

$$
\begin{aligned}
-\sum_{\mathcal{K}\in\mathcal{T}_h} \int_{\mathcal{K}} \Big(\frac{\partial W_i}{\partial x_0} U_i + \frac{\partial W_i}{\partial x_k}(F^e_{ik} - \Theta_{ik})\Big)\, d\mathcal{K}& \\
+ \sum_{\mathcal{K}\in\mathcal{T}_h} \int_{\partial\mathcal{K}} W_i^L(\widehat{U}_i + \widehat{F}^e_{ik} - \widehat{\Theta}_{ik}) n_k^L \, d(\partial\mathcal{K}) &= 0,
\end{aligned}
$$

$$
\begin{aligned}
\sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{K}} V_{ik}\Theta_{ik}\, d\mathcal{K} &= \sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{K}} V_{ik} A_{ikrs} \frac{\partial U_r}{\partial x_s}\, d\mathcal{K} \\
&\quad + \sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{Q}} V_{ik}^L A_{ikrs}^L (\widehat{U}_r - U_r^L) \bar{n}_s^L \, d\mathcal{Q}
\end{aligned}
$$

Geometry of Space-Time Element



Geometry of 2D space-time element in both computational and physical space.

Transformation to Arbitrary Lagrangian Eulerian form

- The space-time normal vector on a grid moving with velocity $\vec{v}$ is:

$$n = \begin{cases} (1,0,0,0)^T & \text{at } K(t_{n+1}^-), \\ (-1,0,0,0)^T & \text{at } K(t_n^+), \\ (-v_k \bar{n}_k, \bar{n})^T & \text{at } Q^n. \end{cases}$$

- The boundary integral then transforms into:

$$\sum_{\mathcal{K}\in\mathcal{T}_h} \int_{\partial\mathcal{K}} W_i^L(\widehat{U}_i + \widehat{F}_{ik}^e - \widehat{\Theta}_{ik}) n_k^L \, d(\partial\mathcal{K})$$

$$= \sum_{K\in\mathcal{T}_h} \Big(\int_{K(t_{n+1}^-)} W_i^L \widehat{U}_i \, dK + \int_{K(t_n^+)} W_i^L \widehat{U}_i \, dK \Big)$$

$$+ \sum_{K\in\mathcal{T}_h} \int_{\mathcal{Q}} W_i^L(\widehat{F}_{ik}^e - \widehat{U}_i v_k - \widehat{\Theta}_{ik}) \bar{n}_k^L \, d\mathcal{Q}$$

Numerical Fluxes

- The numerical flux $\widehat{U}$ at $K(t_{n+1}^-)$ and $K(t_n^+)$ is defined as an upwind flux to ensure causality in time:

$$\widehat{U} = \begin{cases} U^L & \text{at } K(t_{n+1}^-), \\ U^R & \text{at } K(t_n^+), \end{cases}$$

- At the space-time faces $\mathcal{Q}$ we introduce the HLLC approximate Riemann solver as numerical flux:

$$\bar{n}_k(\widehat{F}_{ik}^e - \widehat{U}_i v_k)(U^L, U^R) = H_i^{\mathrm{HLLC}}(U^L, U^R, v, \bar{n})$$

ALE Weak Formulation

- The ALE flux formulation of the compressible Navier-Stokes equations transforms now into:

  Find a $U \in W_h$, such that for all $W \in W_h$, the following holds:

$$
\begin{aligned}
&-\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{K}}\Big(\frac{\partial W_i}{\partial x_0}U_i+\frac{\partial W_i}{\partial x_k}(F^e_{ik}-\Theta_{ik})\Big)\,d\mathcal{K}\\
&+\sum_{K\in\mathcal{T}_h^n}\Big(\int_{K(t^-_{n+1})}W_i^L U_i^L\,dK-\int_{K(t^+_n)}W_i^L U_i^R\,dK\Big)\\
&+\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{Q}}W_i^L(H_i^{\mathrm{HLLC}}(U^L,U^R,v,\bar{n})-\widehat{\Theta}_{ik}\bar{n}_k^L)\,d\mathcal{Q}=0.
\end{aligned}
$$

Ensuring Monotonicity of Second and Higher Order DG Discretizations

- For flow discontinuities a stabilization operator is added to the weak formulation

$$\sum_{j=1}^{N_n} \int_{\mathcal{K}_j^n} (\nabla\, W_h)^T \cdot \mathfrak{D}(U_h) : \nabla\, U_h \, d\mathcal{K}$$

The dyadic product is defined as $A : B = A_{ij}B_{ij}$ for $A, B \in R^{n \times m}$.

- Both the jumps at element faces and the element residual are used to define the artificial viscosity (Jaffre, Johnson and Szepessy model).

- A stabilization operator results in a numerical scheme which can converge to steady state. This is not possible with a slope limiter.

Efficient Solution of Nonlinear Algebraic System

- The space-time DG discretization results in a large system of nonlinear algebraic equations:

$$\mathcal{L}(\hat{U}^n; \hat{U}^{n-1}) = 0$$

- This system is solved by marching to steady state using pseudo-time integration and multigrid techniques:

$$\frac{\partial \hat{U}}{\partial \tau} = -\frac{1}{\Delta t} \mathcal{L}(\hat{U}; \hat{U}^{n-1})$$

Benefits of Coupled Pseudo-Time and Multigrid Approach

- The locality of the DG discretization is preserved, which is beneficial for parallel computing and hp-adaptation.
- In comparison with a Newton method the memory overhead is considerably smaller
- The algorithm has good stability and convergence properties and is not sensitive to initial conditions

EXI Runge-Kutta Scheme

- **Explicit Runge-Kutta method for inviscid flow with Melson correction.**

  1. Initialize $\hat{V}^0 = \hat{U}$.
  2. For all stages $s = 1$ to 5 compute $\hat{V}^s$ as:

  $$\left(I + \alpha_s \lambda I\right) \hat{V}^s = \hat{V}^0 + \alpha_s \lambda \left(\hat{V}^{s-1} - \mathcal{L}(\hat{V}^{s-1}; \hat{U}^{n-1})\right).$$

  3. Return $\hat{U} = \hat{V}^5$.

- Runge-Kutta coefficients: $\alpha_1 = 0.0791451$, $\alpha_2 = 0.163551$, $\alpha_3 = 0.283663$, $\alpha_4 = 0.5$ and $\alpha_5 = 1.0$.

- The factor λ is the ratio between the pseudo- and physical-time step: $\lambda = \Delta\tau / \Delta t$.

EXV Runge-Kutta Scheme

- **Explicit Runge-Kutta method for viscous flows.**

  1. Initialize $\hat{V}^0 = \hat{U}$.
  2. For all stages $s = 1$ to 4 compute $\hat{V}^s$ as:

  $$\hat{V}^s = \hat{V}^0 - \alpha_s \lambda \mathcal{L}(\hat{V}^{s-1}; \hat{U}^{n-1}).$$

  3. Return $\hat{U} = \hat{V}^4$.

- Runge-Kutta coefficients: $\alpha_1 = 0.0178571$, $\alpha_2 = 0.0568106$, $\alpha_3 = 0.174513$ and $\alpha_4 = 1.0$.

Combined EXI-EXV Runge-Kutta Scheme

- Time accuracy is not important in pseudo-time, we apply therefore local pseudo-time stepping and deploy whichever scheme gives the mildest stability constraint.
- The EXI scheme has the mildest stability constraint for relatively high cell Reynolds numbers and the EXV scheme for relatively low cell Reynolds numbers.
- The pseudo-time Runge-Kutta schemes act as smoother in a multigrid algorithm.

Stability Analysis

- Stability analysis is conducted for the linear advection-diffusion equation with periodic boundary conditions

$$u_t + au_x = du_{xx}, \quad t \in (0, T), \quad x \in \mathbb{R},$$

  with $a > 0$ and $d > 0$ constant.

- The domain is divided into uniform rectangular elements Δt by Δx.

- The discretization depends on the CFL number

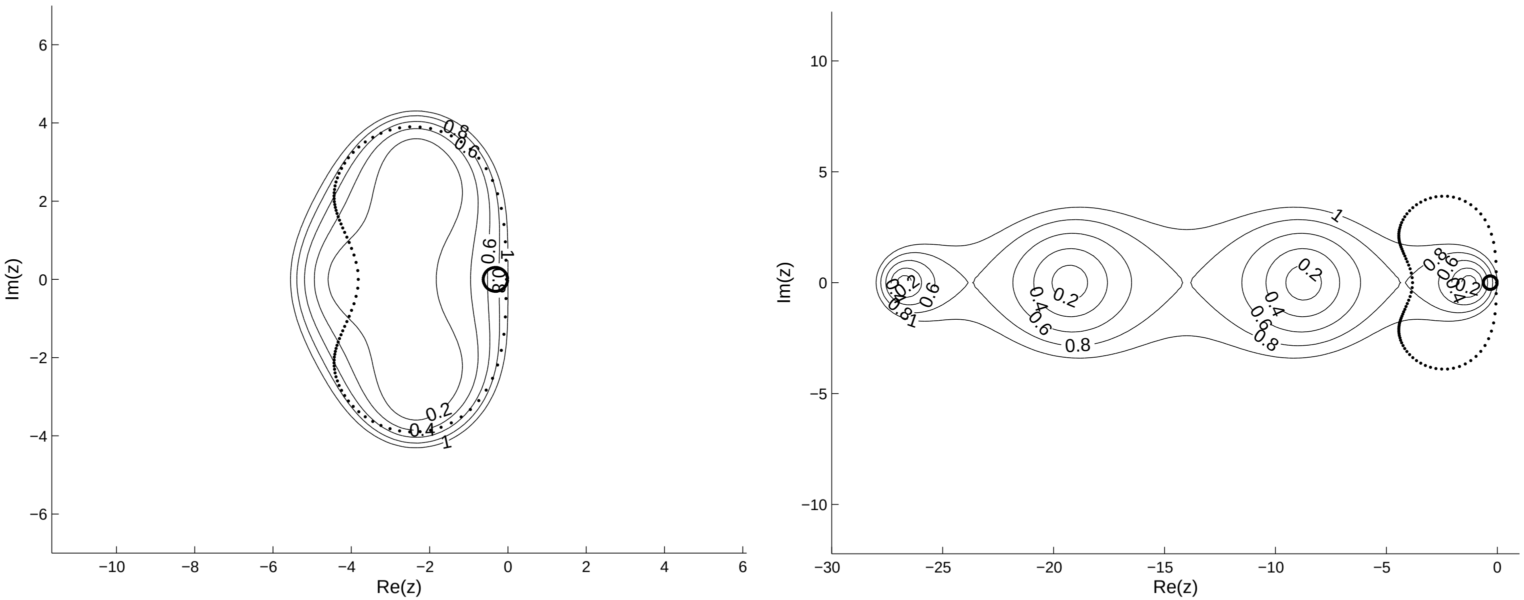
$$CFL_{\triangle t} = a\triangle t/\triangle x$$

  the diffusion number

$$\beta = d\triangle t/(\triangle x)^2$$

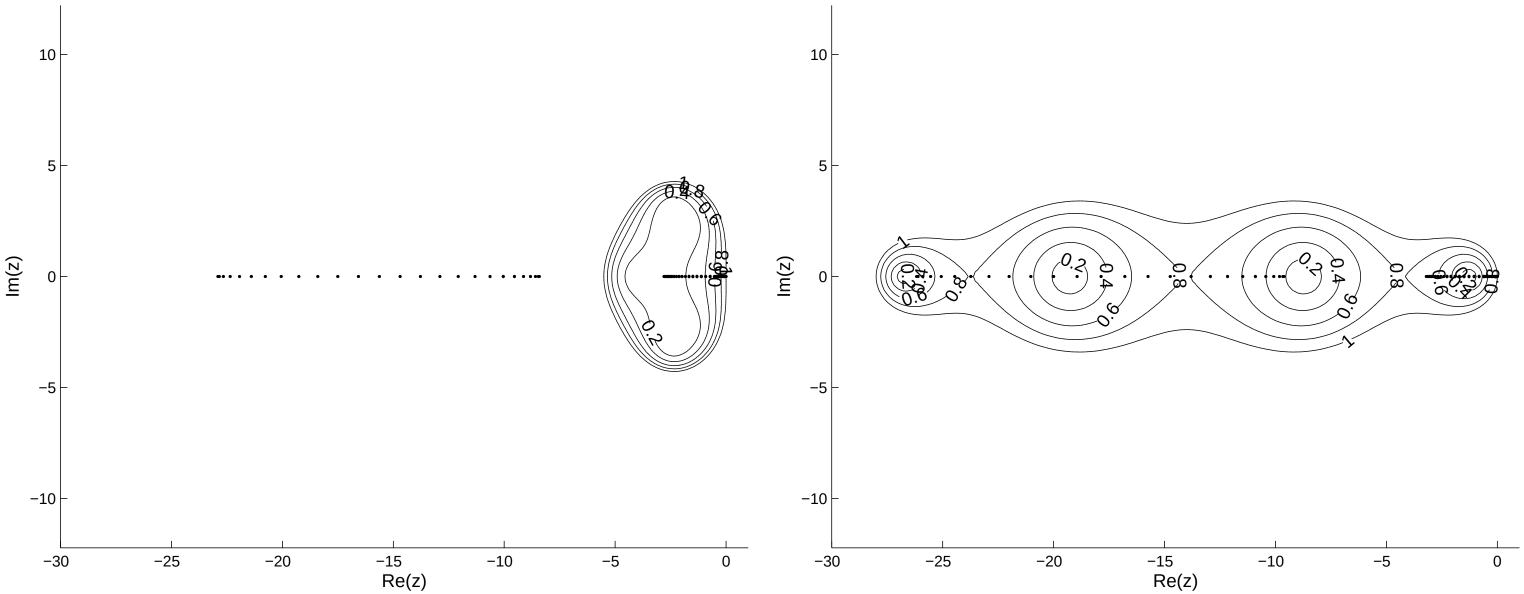
  and the stabilization coefficient η.

Stability Analysis for Steady State Inviscid Problems



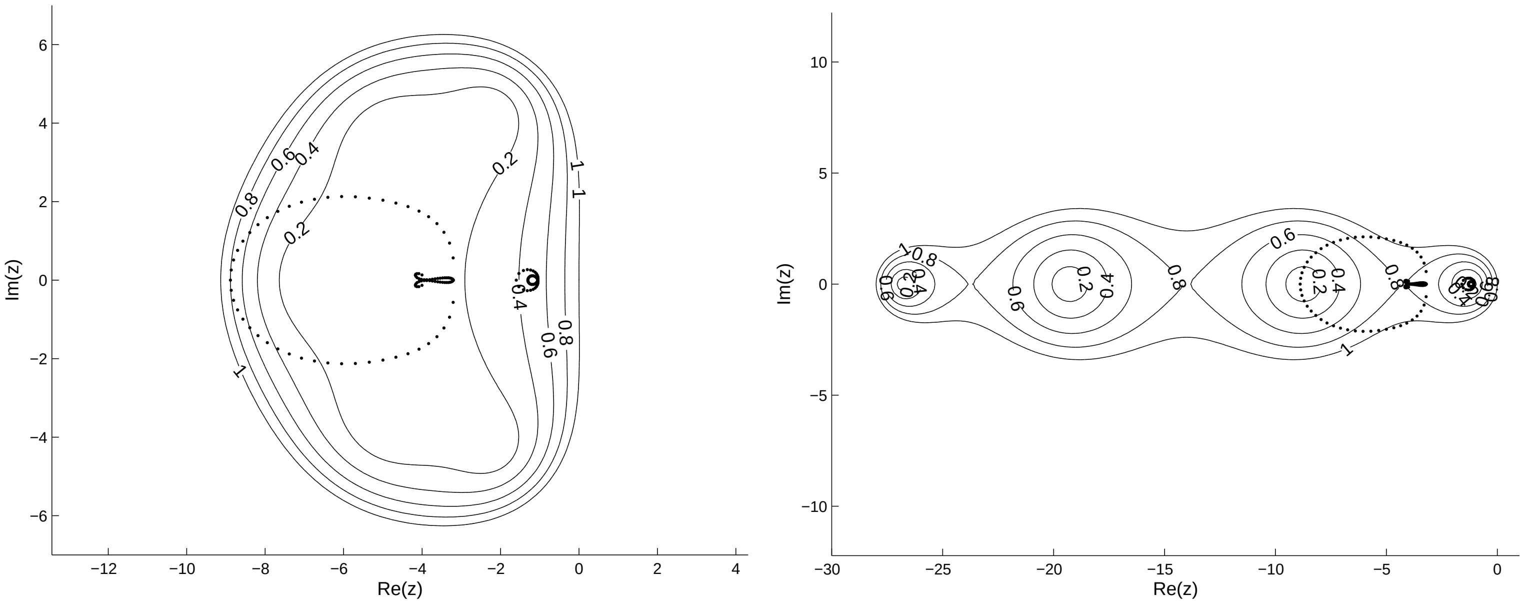
Eigenvalues and stability domain for the EXI method (L) and EXV method (R) in the steady-state inviscid flow regime ($\lambda = 1.8 \cdot 10^{-2}, CFL_{\triangle t} = 1.8$).

Stability Analysis for Steady State Viscous Problems



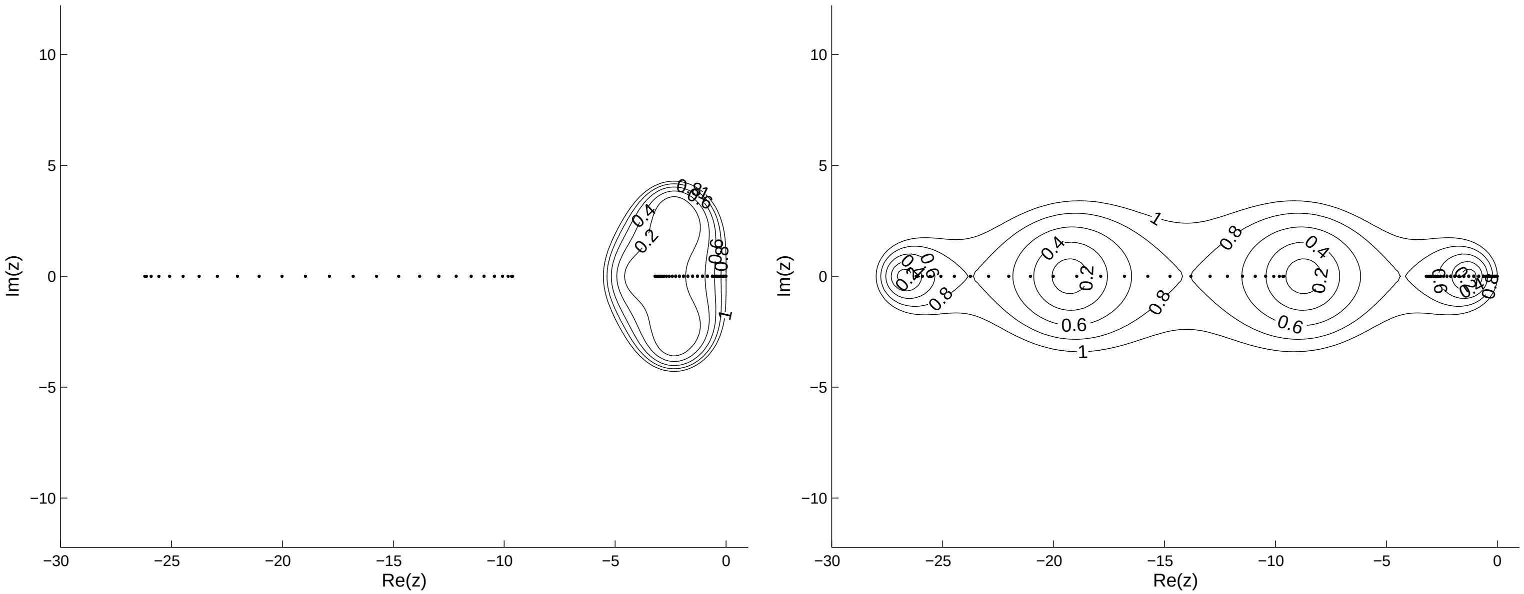
Eigenvalues and stability domain for the EXI method (L) and EXV method (R) in the steady-state viscous flow regime ($\lambda = 8 \cdot 10^{-5}, \beta = 0.8$).

Stability Analysis for Time-Dependent Inviscid Problems



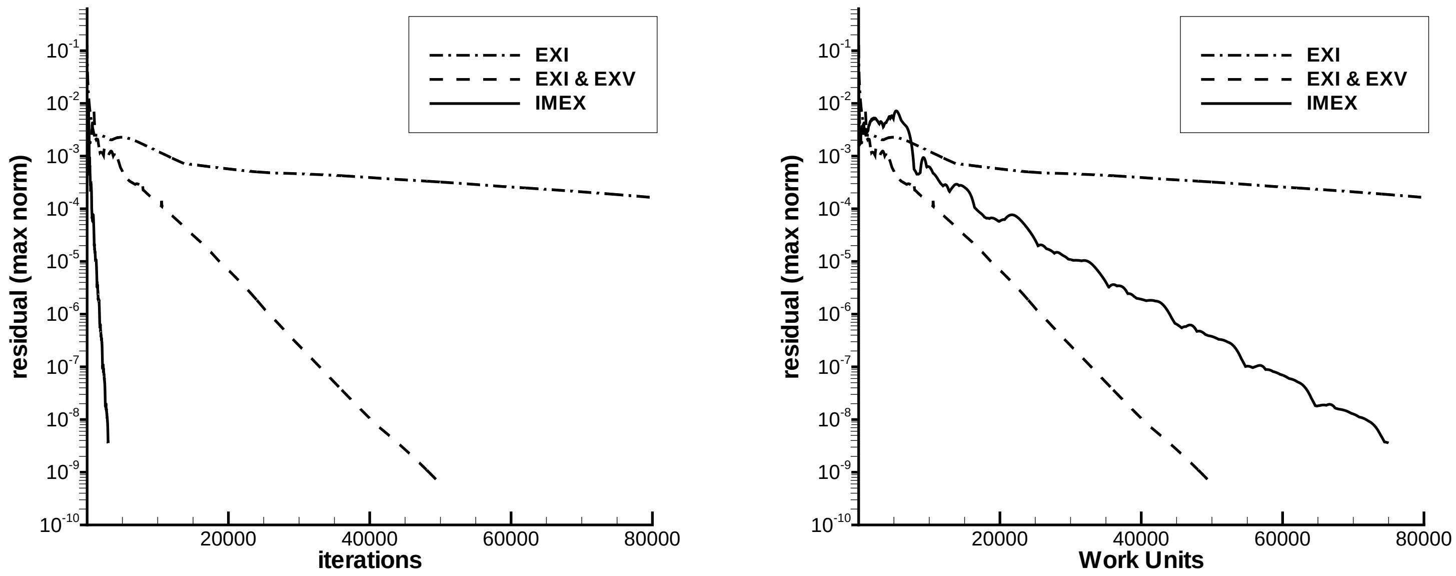
Eigenvalues and stability domain for the EXI method (L) and EXV method (R) in the time-dependent inviscid flow regime ($\lambda = 1.6, CFL_{\triangle t} = 1.6$).

Stability Analysis for Time-Dependent Viscous Problems



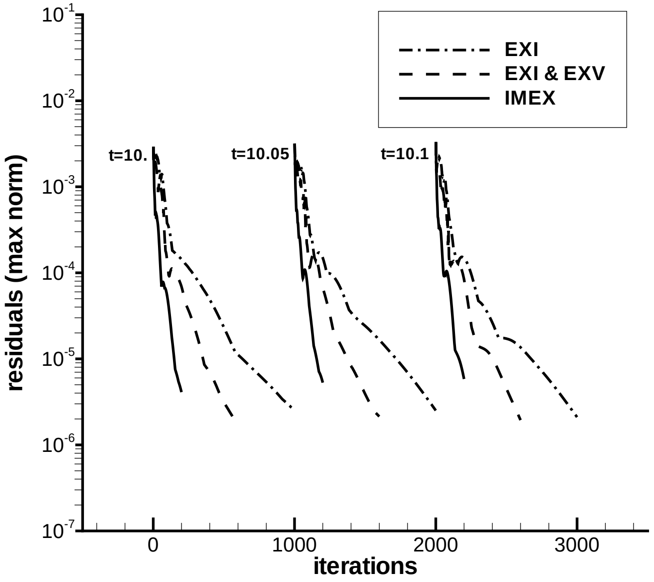
Eigenvalues and stability domain for the EXI method (L) and EXV method (R) in the time-dependent viscous flow regime ($\lambda = 8 \cdot 10^{-3}, \beta = 0.8$).

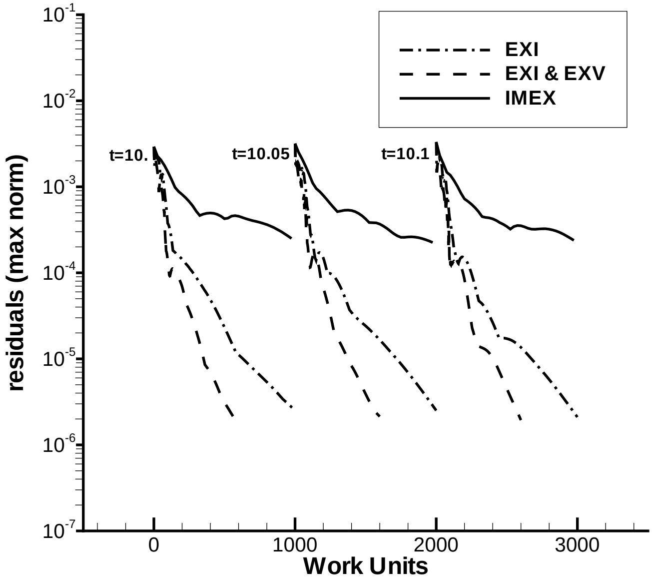
Performance of Pseudo-Time Integration Schemes



Convergence to steady state for the GAMM A1 case ($\mathrm{M}_\infty = 0.8$, $\mathrm{Re}_\infty = 73$, $\alpha = 12^\circ$, 112×38 grid).

Performance of Time Integration Schemes





Convergence in pseudo-time for three physical time steps in the GAMM A7 case ($M_\infty = 0.85$, $Re_\infty = 10^4$, $\alpha = 0^\circ$, 224×76 grid).

Two-Level h-Multigrid Algorithm

- At the core of any multigrid method is the two-level algorithm.
- Subscripts $(\cdot)_h$ and $(\cdot)_H$ denote a quantity $(\cdot)$ on the fine and coarse grid.
- Define:
  - $\hat{U}$ an approximation of the solution $\hat{U}^n$
  - R the restriction operator for the solution
  - $\bar{R}$ the restriction operator for the residuals
  - P the prolongation operator
- The h-multigrid algorithm is applied only in space, hence the time-step is equal on both levels; but multi-time multi-space multigrid is also feasible.

Two-level h-Multigrid Algorithm

Two-level algorithm.

1. Take one pseudo-time step on the fine grid with the combined EXI and EXV methods, this gives the approximation $\hat{U}_h$.
2. Restrict this approximation to the coarse grid: $\hat{U}_H = R(\hat{U}_h)$.
3. Compute the forcing:

$$F_H \equiv \mathcal{L}(\hat{U}_H; \hat{U}_H^{n-1}) - \bar{R}\big(\mathcal{L}(\hat{U}_h; \hat{U}_h^{n-1})\big).$$

4. Solve the coarse grid problem for the unknown $\hat{U}_H^*$:

$$\mathcal{L}(\hat{U}_H^*; \hat{U}_H^{n-1}) - F_H = 0,$$

5. Compute the coarse grid error $E_H = \hat{U}_H^* - \hat{U}_H$ and correct the fine grid approximation: $\hat{U}_h \leftarrow \hat{U}_h + P(E_H)$.

Two-level h-Multigrid Algorithm

- Solving the coarse grid problem at stage four of the multigrid algorithm can again be done with the two-level algorithm.
- This recursively defines the V-cycle multi-level algorithm in terms of the two-level algorithm.
- It is common practice to take ν_1 pseudo-time pre-relaxation steps at stage one and another ν_2 post-relaxation pseudo-time steps after stage five.
- The exact solution of the problem on the coarsest grid is not always feasible; instead one simply takes $\nu_1 + \nu_2$ relaxation steps.

Inter-Grid Transfer Operators

- The inter-grid transfer operators stem from the L_2-projection of the coarse grid solution U_H in an element $\mathcal{K}_H$ on the corresponding set of fine elements $\{\mathcal{K}_h\}$.
- The solution U_h in element $\mathcal{K}_h$ can be found by solving:

$$\int_{\mathcal{K}_h} W_i U_i^h \, d\mathcal{K} = \int_{\mathcal{K}_h} W_i U_i^H \, d\mathcal{K}, \quad \forall W \in W_h.$$

- This relation supposes the embedding of spaces, i.e. $W_H \subset W_h$, to ensure that U_H is defined on $\mathcal{K}_h$.

Prolongation Algorithm

- Introducing the polynomial expansions of the test and trial functions, we obtain the prolongation operator $P : U^H \to U^h$:

$$\hat{U}^h_{im} = (M_h^{-1})_{ml}\Big(\int_{\mathcal{K}_h} \psi_l^h \psi_n^H \, d\mathcal{K} \Big) \hat{U}^H_{in}.$$

  with the mass matrix M_h of element $\mathcal{K}_h$

- The restriction operator for the residuals is defined as the transpose of the prolongation operator: $\bar{R} = P^T$.

- The restriction operator R for the solution is defined as $R = P^{-1}$, such that the property $U_H = R(P(U_H))$ holds, meaning that the inter-grid transfer does not modify the solution.

Error Amplification Operator

- The error amplification operator of the two-level algorithm M_h^{TLA}, is given by:

$$M_h^{\mathrm{TLA}} = M_h^{\mathrm{CGC}} M_h^{\mathrm{REL}},$$

  with M_h^{REL} the error amplification operator associated with either the EXI or EXV scheme.

- The coarse grid correction (CGC) of the multigrid algorithm is given by:

$$M^{\mathrm{CGC}} = I - P\mathcal{L}_H^{-1}\bar{R}\mathcal{L}_h.$$

- The convergence behaviour of the two-level algorithm for the space-time DG discretization is given by the spectral radius of the error amplification operator$\rho(M_h^{\mathrm{TLA}})$.

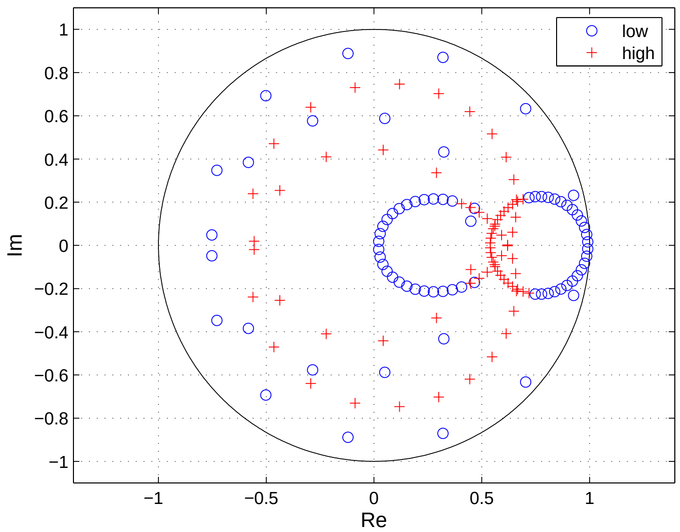
Stability Parameters

- The space-time DG discretization is implicit in time and unconditionally stable.
- The Runge-Kutta methods are explicit in pseudo time and their stability depends on the ratio λ between the pseudo time step and the physical time step $\lambda = \Delta\tau/\Delta t$.
- The stability condition is expressed in terms of the pseudo-time CFL number $\sigma_{\Delta\tau}$ and the pseudo-time diffusive Von Neumann condition $\delta_{\Delta\tau}$:

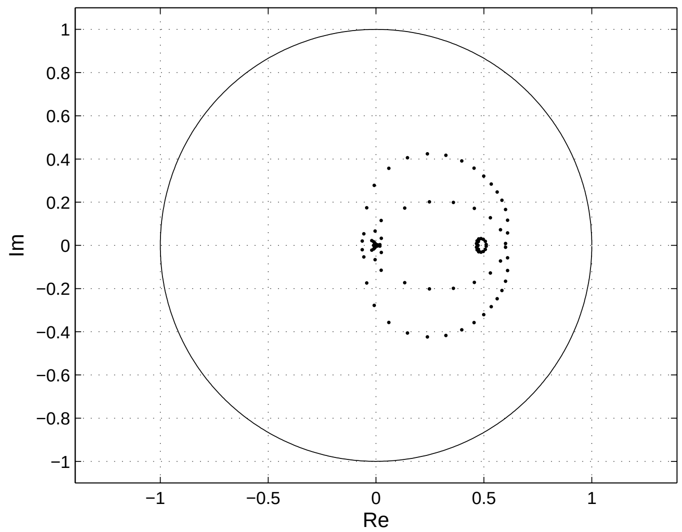
$$\Delta\tau \leq \Delta\tau^a \equiv \frac{\sigma_{\Delta\tau} h}{a} \quad \text{and} \quad \Delta\tau \leq \Delta\tau^d \equiv \frac{\delta_{\Delta\tau} h^2}{d}.$$

The pseudo-time CFL number is given by $\sigma_{\Delta\tau} = \lambda\sigma$ and the pseudo-time diffusive Von Neumann number by $\delta_{\Delta\tau} = \lambda\sigma/\mathrm{Re}_h$

Eigenvalue Spectra Two-Level Algorithm with EXI Smoother (Steady Case)



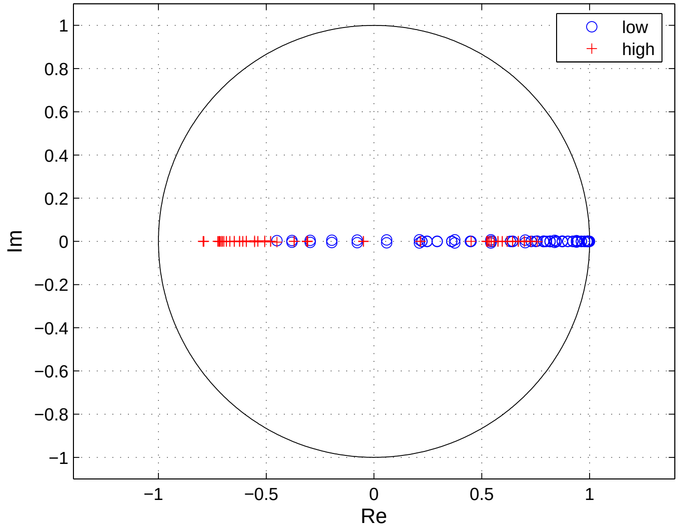
(a) EXI



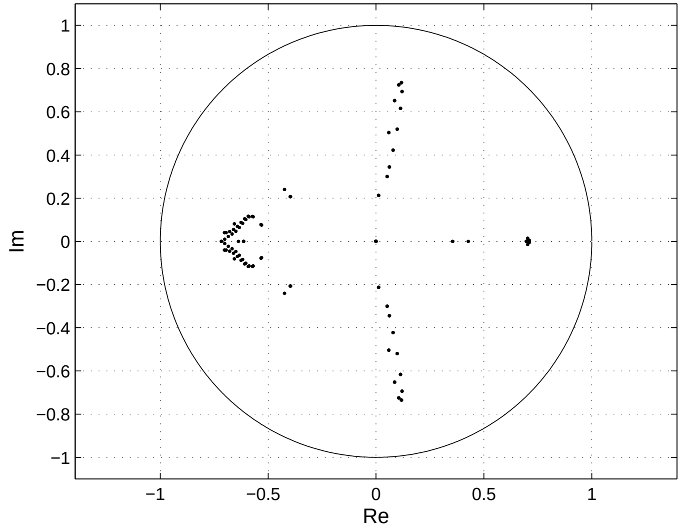
(b) TLA with EXI

Eigenvalue spectra of the EXI smoother and two-level algorithm in the steady advection dominated case ($\sigma = 100$ and $\mathrm{Re}_h = 100$).

Eigenvalue Spectra Two-Level Algorithm with EXV Smoother (Steady Case)



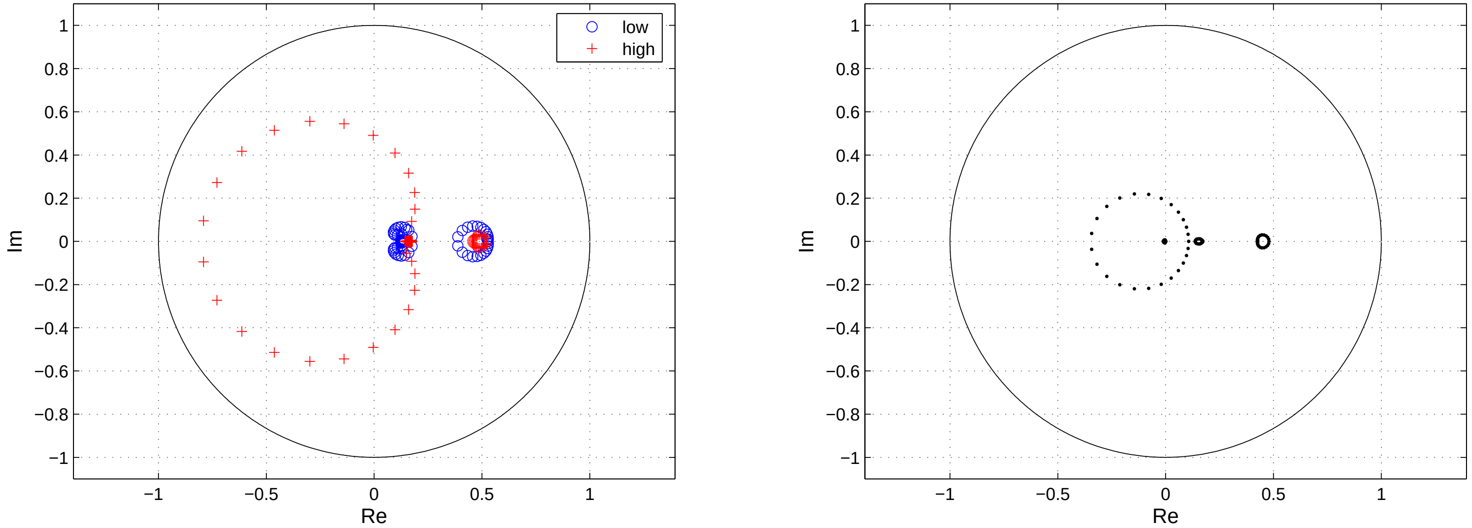
(c) EXI



(d) TLA with EXI

Eigenvalue spectra of the EXV smoother and two-level algorithm in the steady diffusion dominated case ($\sigma = 100$ and $\mathrm{Re}_h = 0.01$).

Eigenvalue Spectra Two-Level Algorithm with EXI Smoother (unsteady case)

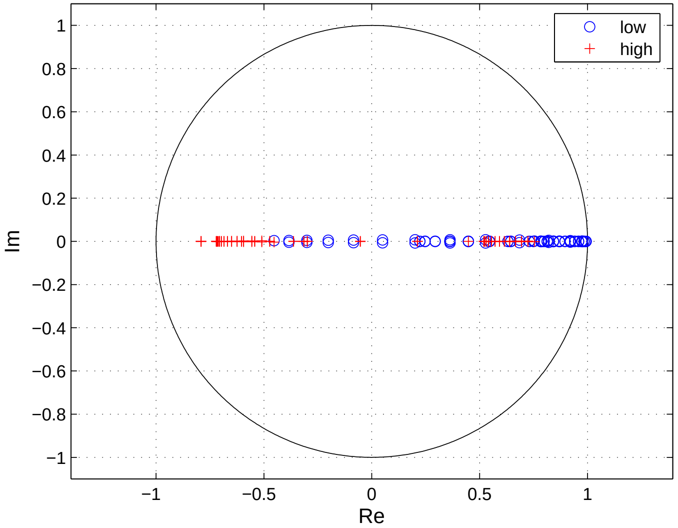


(e) EXI

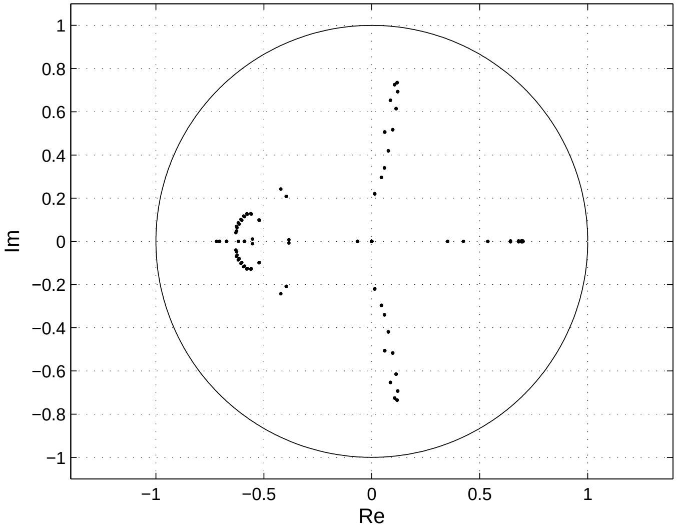
(f) TLA with EXI

Eigenvalue spectra of the EXI smoother and two-level algorithm in the unsteady advection dominated case ($\sigma = 1$ and $\mathrm{Re}_h = 100$).

Eigenvalue Spectra Two-Level Algorithm with EXV Smoother (unsteady case)



(g) EXI



(h) TLA with EXI

Eigenvalue spectra of the EXV smoother and two-level algorithm in the unsteady diffusion dominated case ($\sigma = 1$ and $\mathrm{Re}_h = 0.01$).

Spectral Radii of Two-Level Algorithm for Steady Problems ($\sigma = 100$)

EXI smoother

Re_h	$\Delta\tau/\Delta t$	$\rho\left(M_h^{\mathrm{EXI}}\right)$	$\rho\left(M_h^{\mathrm{TLA}}\right)$
100	1.8e-02	0.991	0.622
10	8.0e-03	0.996	0.716
1	1.4e-03	0.999	0.906

EXV smoother

Re_h	$\Delta\tau/\Delta t$	$\rho\left(M_h^{\mathrm{EXV}}\right)$	$\rho\left(M_h^{\mathrm{TLA}}\right)$
100	2.0e-03	0.999	0.914
10	3.0e-03	0.998	0.871
1	7.0e-03	0.996	0.697

Spectral Radii of Two-Level Algorithm for Unsteady Problems ($\sigma = 1$)

EXI smoother

Re_h	$\Delta\tau/\Delta t$	$\rho\left(M_h^{\mathrm{EXI}}\right)$	$\rho\left(M_h^{\mathrm{TLA}}\right)$
100	1.6e-00	0.796	0.479
10	8.0e-01	0.918	0.599
1	1.4e-01	0.904	0.837

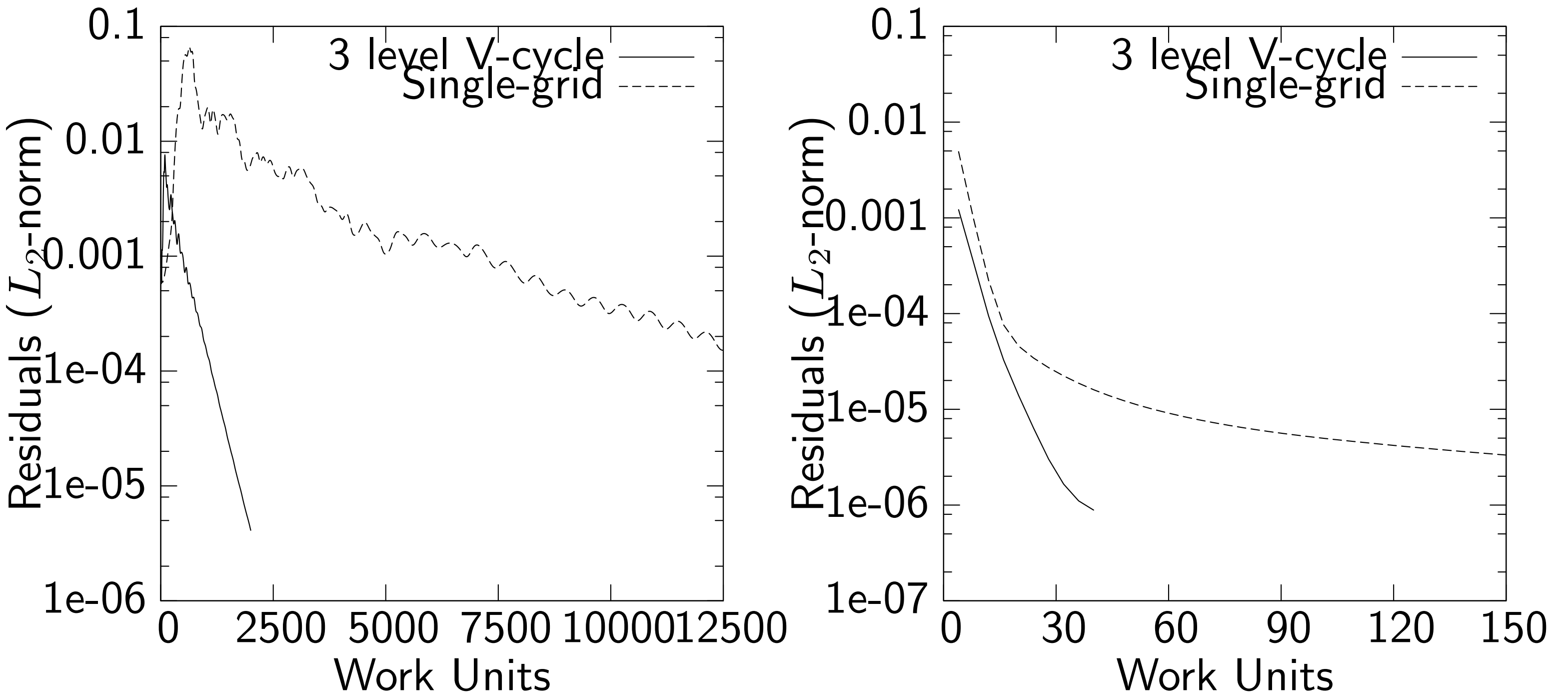
EXV smoother

Re_h	$\Delta\tau/\Delta t$	$\rho\left(M_h^{\mathrm{EXV}}\right)$	$\rho\left(M_h^{\mathrm{TLA}}\right)$
100	1.0e-00	0.924	0.660
10	7.0e-01	0.812	0.704
1	7.0e-01	0.805	0.719

Numerical Simulations

- Definition of work units:
  - One work unit corresponds to one Runge-Kutta step on the fine grid.
  - The work on a one times coarsened mesh is $\frac{1}{8}$ of the work on the fine grid ($\frac{1}{4}$ in 2D).

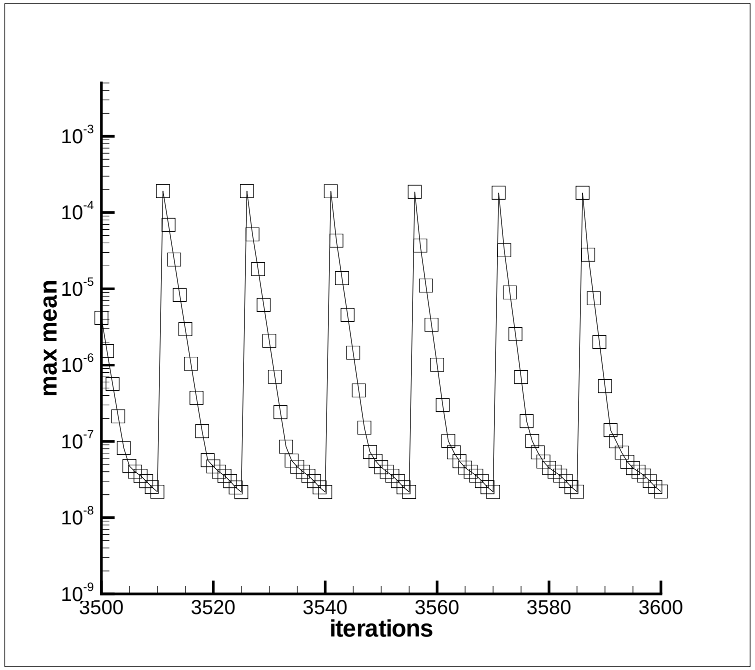
Convergence Rate for Flow about a Circular Cylinder



Steady-state (L) and time-accurate (R)

$\mathrm{M}_\infty = 0.3$, $\mathrm{Re}_\infty = 40$, 64×64 mesh (L) and 80×84 mesh (R)

Convergence Rate for Unsteady Flow about Circular Cylinder



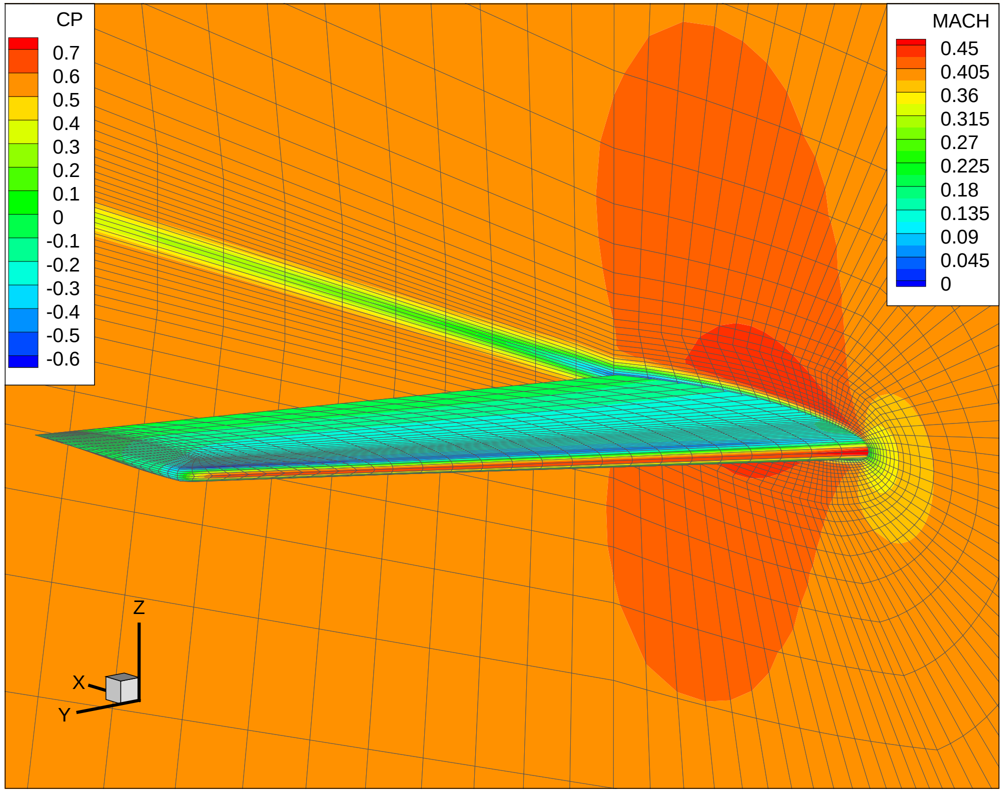
$\mathrm{M}_\infty = 0.3$, $\mathrm{Re}_\infty = 1000$ on a 128×128 mesh

Multigrid: 3 level V-cycle, 4 relaxation steps on each level.

Flow about ONERA M6 Wing

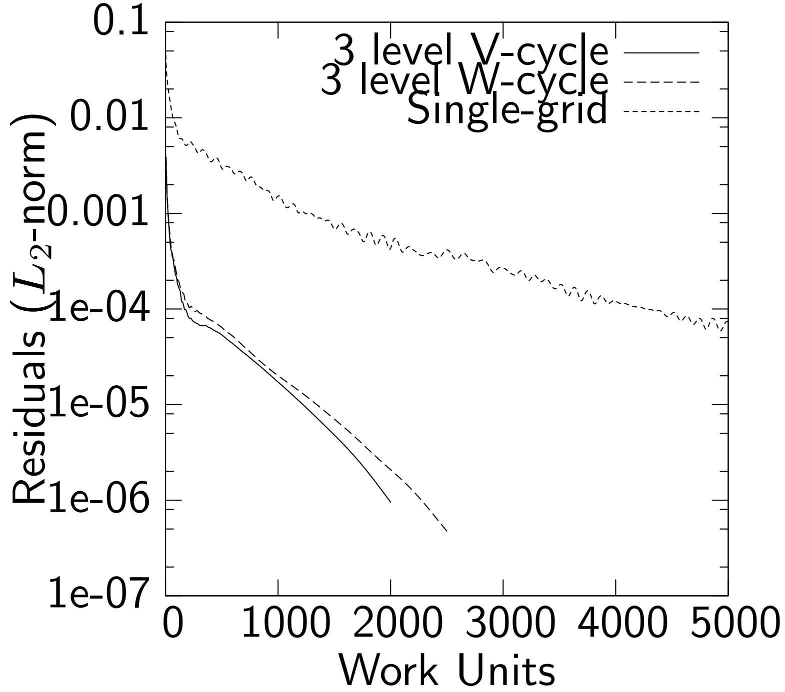
- Steady laminar flow about the ONERA M6 wing at $\mathrm{M}_\infty = 0.4$, $\mathrm{Re}_\infty = 10^4$ and angle of attack $\alpha = 1^\circ$.
- Fine grid consists of $125\,000$ hexahedral elements.
- Multigrid iteration consisting of 3 level V- or W-cycles.
- The V-cycle has a total of 4 relaxations on each grid level, while the W-cycle has 4 relaxations on the fine grid and 8 on the medium and coarse grid.

Grid and Flow about ONERA M6 Wing



Mach number isolines and the pressure coefficient C_p on the ONERA M6 wing $\mathrm{M}_\infty = 0.4$, $\mathrm{Re}_\infty = 10^4$ and $\alpha = 1^\circ$.

Convergence Rate for ONERA M6 Wing



Convergence in pseudo-time for the ONERA M6 wing

$\mathrm{M}_\infty = 0.4$, $\mathrm{Re}_\infty = 10^4$, $\alpha = 1^\circ$.

Summary of Computational Effort for Different Cases

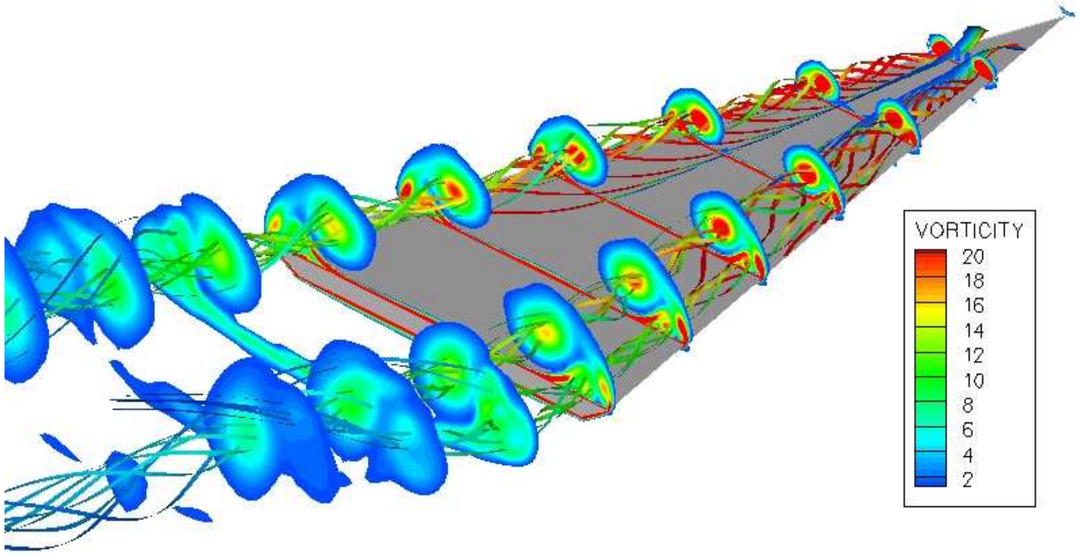
Case	Single-grid performance	Multigrid performance	Cost reduction
cylinder (steady)	2 orders in 12 500 WU	3 orders in 2000 WU	9.4
cylinder (unsteady)	3 orders in 150 WU	3 orders in 30 WU	5.0
ONERA M6	2 orders in 5000 WU	3 orders in 2000 WU	3.7

Summary of computational effort for cylinder and ONERA M6 wing.

Delta Wing Simulations

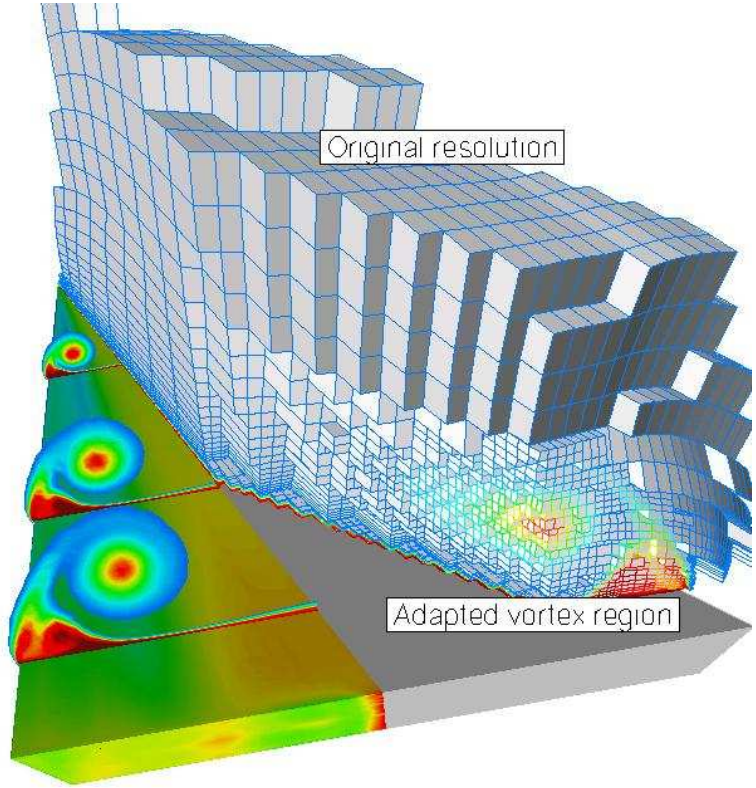
- Simulations of viscous flow about a delta wing with 85° sweep angle.
- Conditions
  - Angle of attack $\alpha = 12.5^\circ$.
  - Mach number $M = 0.3$
  - Reynolds numbers $Re = 40.000$ and $Re = 100.000$ (LES)
  - Unadapted fine grid mesh $1.600.000$ elements, $40.000.000$ degrees of freedom
  - Adapted mesh for LES with 1.919.489 elements, 47.987.225 degrees of freedom

Delta Wing Simulations



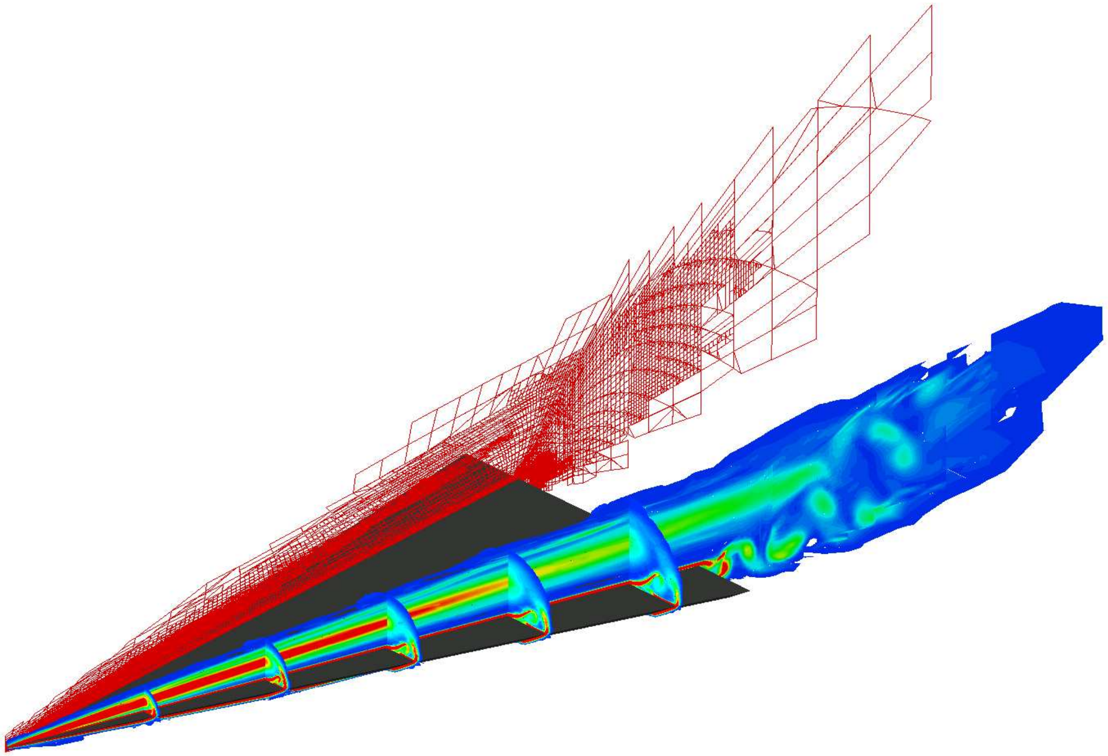
Streaklines and vorticity contours in various cross-sections

Delta Wing Simulations



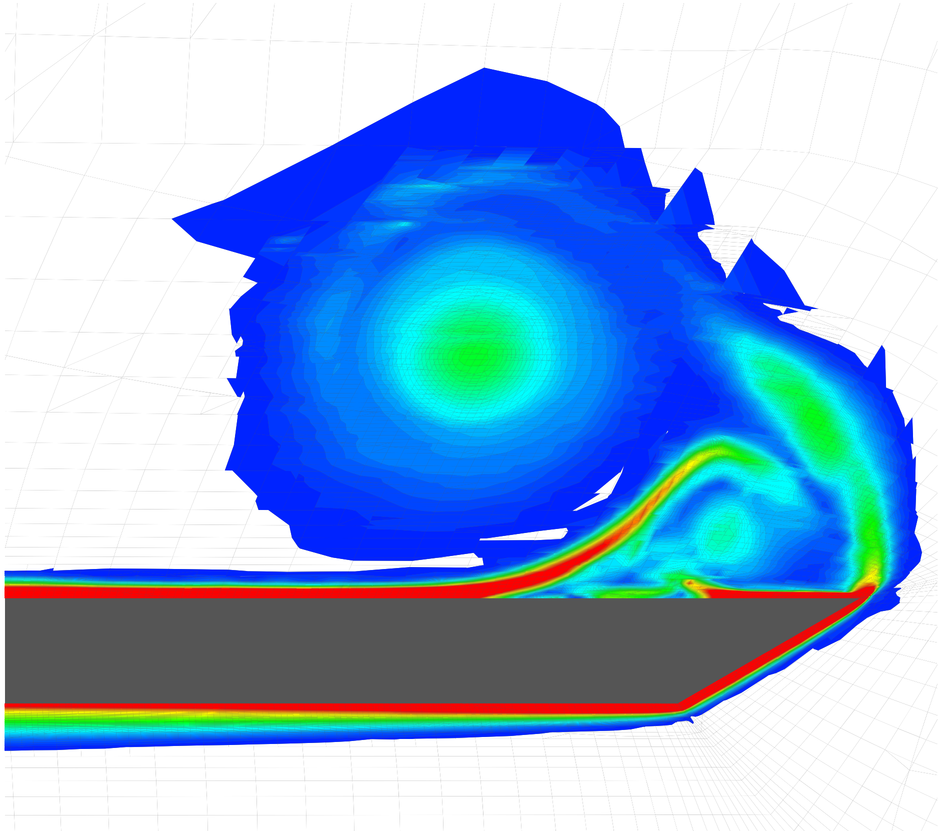
Impression of the vorticity based mesh adaptation

Delta Wing Simulations



Adapted mesh and vorticity field in primary vortex and cross-sections of a delta wing ($Re_c = 100.000$, $Ma = 0.3$, $\alpha = 12.5$ degrees).

Delta Wing Simulations



Vorticity field near leading edge of delta wing at $x = 0.9c$
($Re_c = 100.000$, $Ma = 0.3$, $\alpha = 12.5$ degrees)

NACA0012 Airfoil in Laminar Dynamic Stall

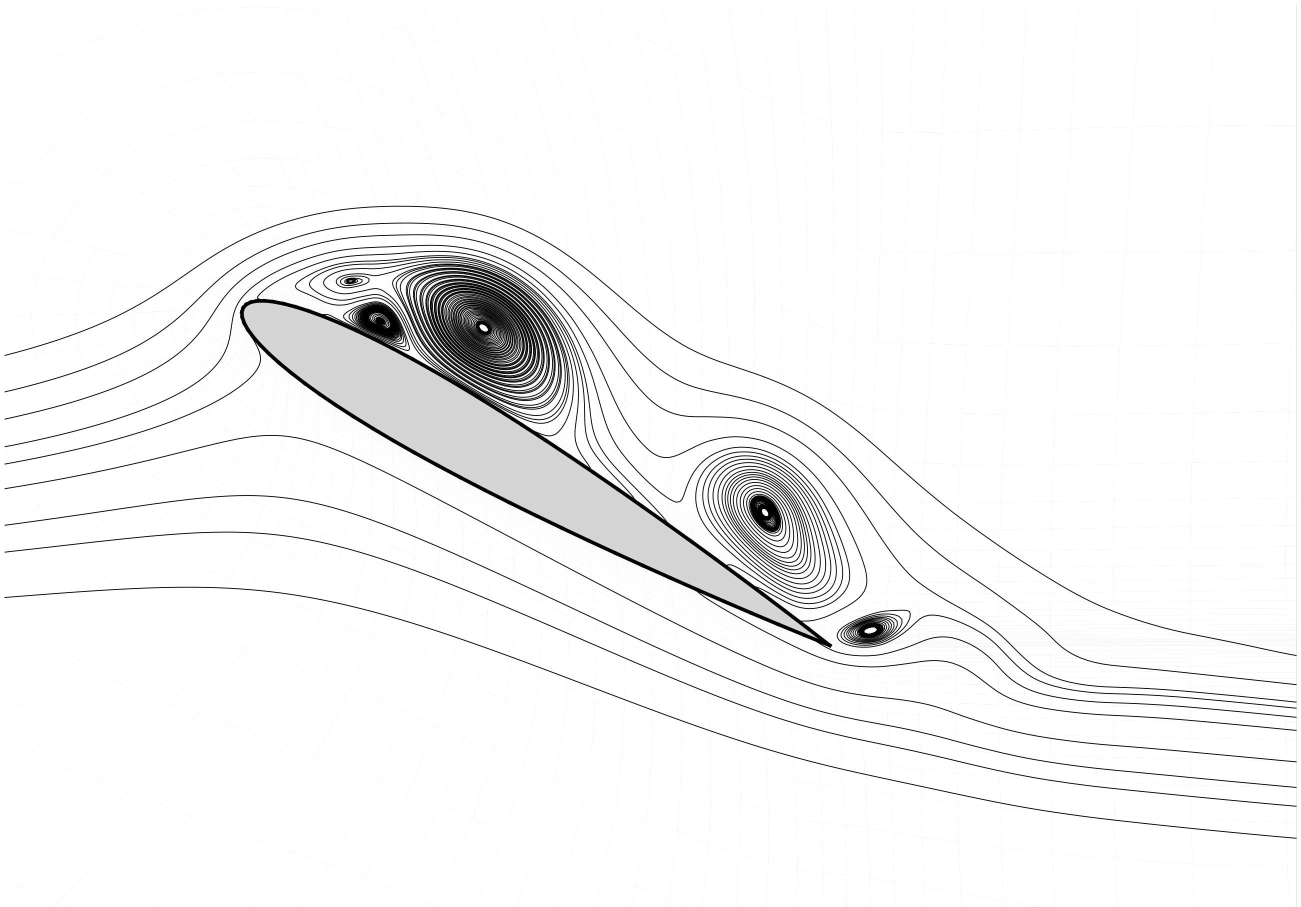
Conditions:

- Free stream Mach number $M_\infty = 0.2$
- Reynolds number 10000
- Pitch axis is situated at 25% from the leading edge
- Angle of attack α evolves as:

$$\alpha(t) = a + bt - a\exp(-ct),$$

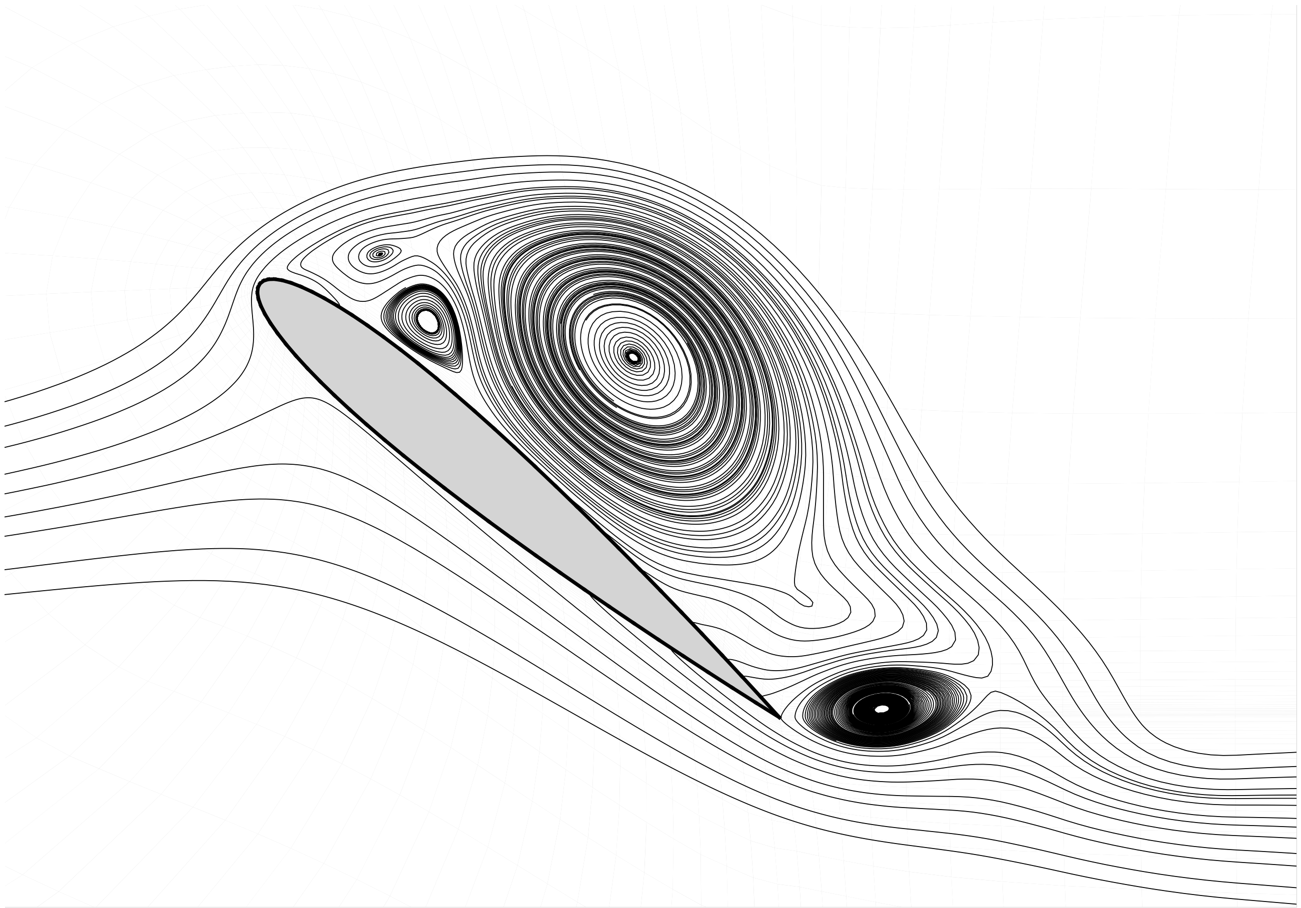
  with coefficients $a = -1.2455604$, $b = 2.2918312$, $c = 1.84$ and time $t \in [0, 25]$.
- Time step $\Delta t = 0.005$
- C-type mesh with 112×38 elements with 14 elements in the boundary layer

NACA0012 Airfoil in Laminar Dynamic Stall



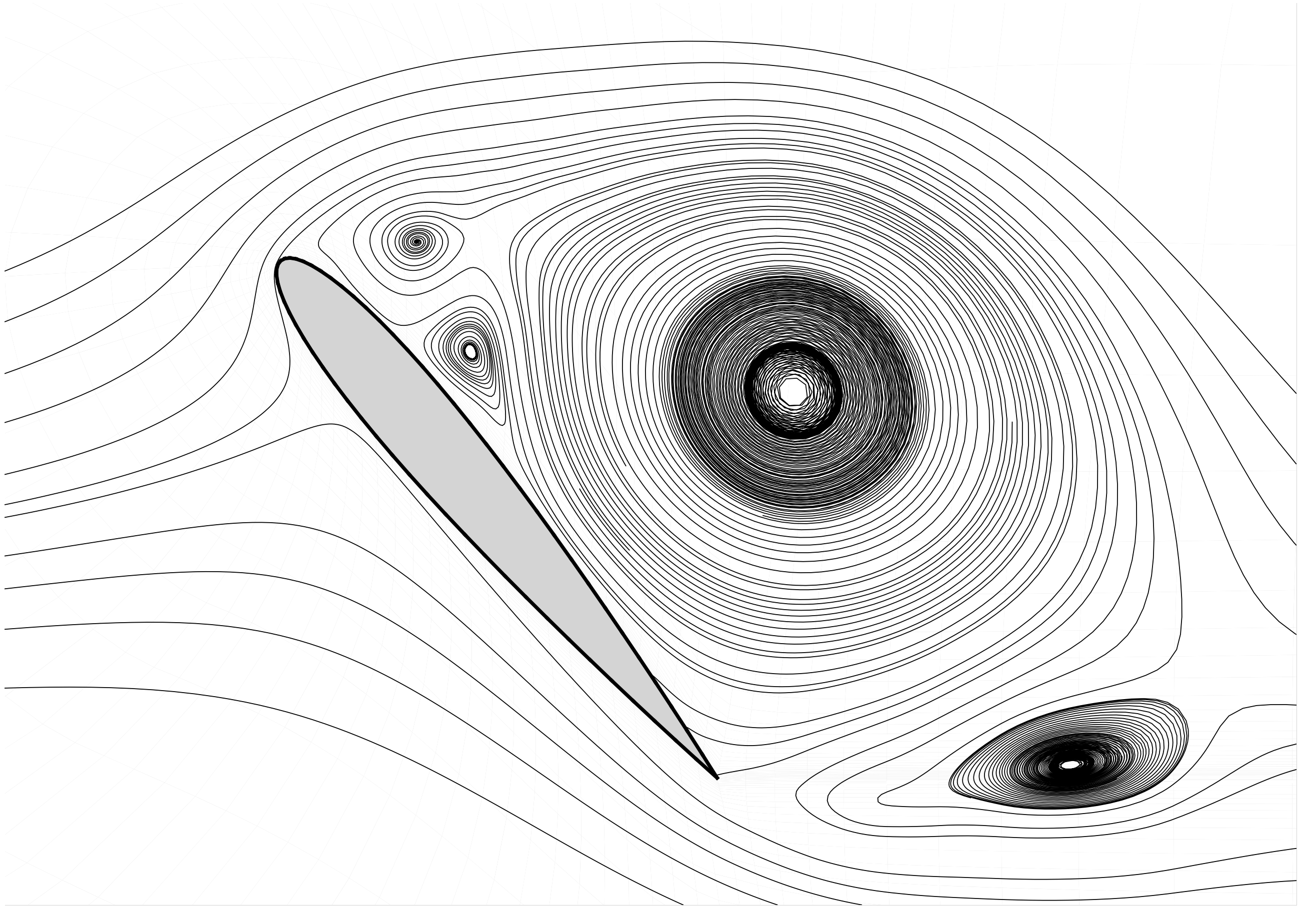
Streamlines around NACA 0012 airfoil in dynamic stall at $\alpha = 30^\circ$.

NACA0012 Airfoil in Laminar Dynamic Stall



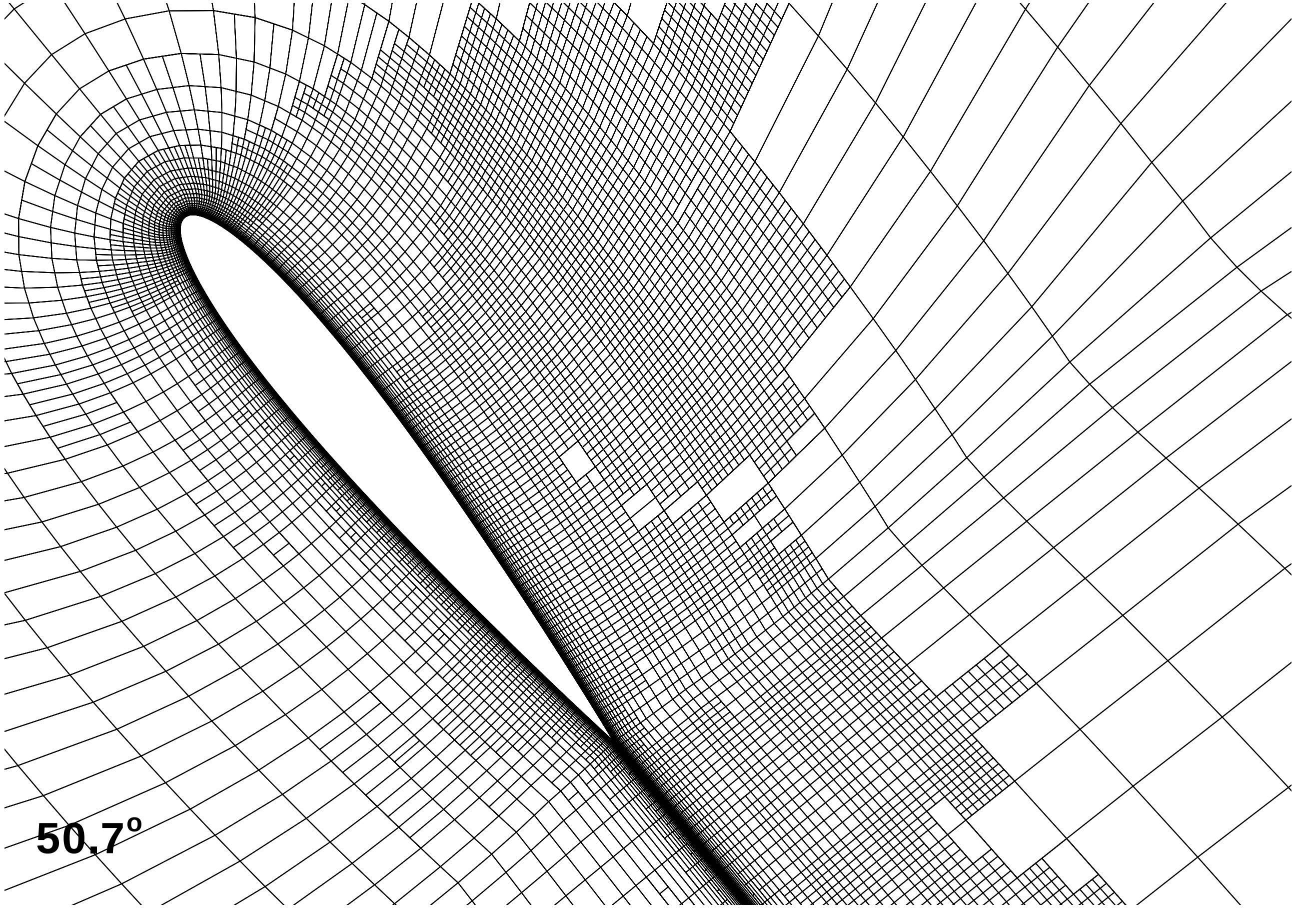
Streamlines around NACA 0012 airfoil in dynamic stall at $\alpha = 40^\circ$.

NACA0012 Airfoil in Laminar Dynamic Stall



Streamlines around NACA 0012 airfoil in dynamic stall at $\alpha = 50^{\circ}$.

NACA0012 Airfoil in Laminar Dynamic Stall



Adapted mesh around NACA 0012 airfoil in dynamic stall at $\alpha = 50.7^\circ$.

Conclusions

The space-time discontinuous Galerkin method has the following interesting properties:

- Accurate, unconditionally stable scheme for the compressible Navier-Stokes equations.
- Conservative discretization on moving and deforming meshes which satisfies the geometric conservation law.
- Local, element based discretization suitable for h-(p) mesh adaptation.
- Optimal accuracy proven for advection-diffusion equation.

Conclusions

- Runge-Kutta pseudo-time integration methods in combination with multigrid are an efficient technique to solve the nonlinear algebraic equations originating from the space-time DG method.
- Two-level Fourier analysis of the space-time DG discretization for the scalar advection-diffusion equation shows good convergence factors.
- The construction of intergrid transfer operators is based on the L_2 projection of the coarse grid solution on the fine grid and assumes embedding of spaces.

More information on: `wwwhome.math.utwente.nl/~vegtjjw/`