Space-Time Discontinuous Galerkin Finite Element Methods

Part II Compressible Navier-Stokes Equations

Jaap van der Vegt

Numerical Analysis and Computational Science Group
Department of Applied Mathematics
Universiteit Twente

Enschede, The Netherlands

Joint work with Christiaan Klaij and Harmen van der Ven (NLR)

USTC Summer School, August 31 – September 4, 2020

Motivation of research

- In many applications one encounters moving and deforming time-dependent flow domains:
  - Aerodynamics: helicopters, maneuvering aircraft, wing control, surfaces
  - Fluid structure interaction,
  - Multi-Fluid flows,
  - Free surface problems,
  - Local time-stepping (not discussed).
- 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 II

- Space-time discontinuous Galerkin finite element discretization for compressible Navier-Stokes equations
- Applications in aerodynamics
- Concluding remarks

Geometry of Space-Time Domain for Three-Dimensional Time-Dependent Problems

- Consider an open domain $\mathcal{E} \subset \mathbb{R}^4$.
- A point $x \in \mathcal{E}$ has coordinates $x = (x_0, \bar{x})$ with $x_0 = t$, t time, and $\bar{x}$ the spatial coordinates.
- The flow domain $\Omega(t)$ at time t is defined as

$$\Omega(t) := \{x \in \mathcal{E} \mid x_0 = t,\ t_0 < t < T\}.$$

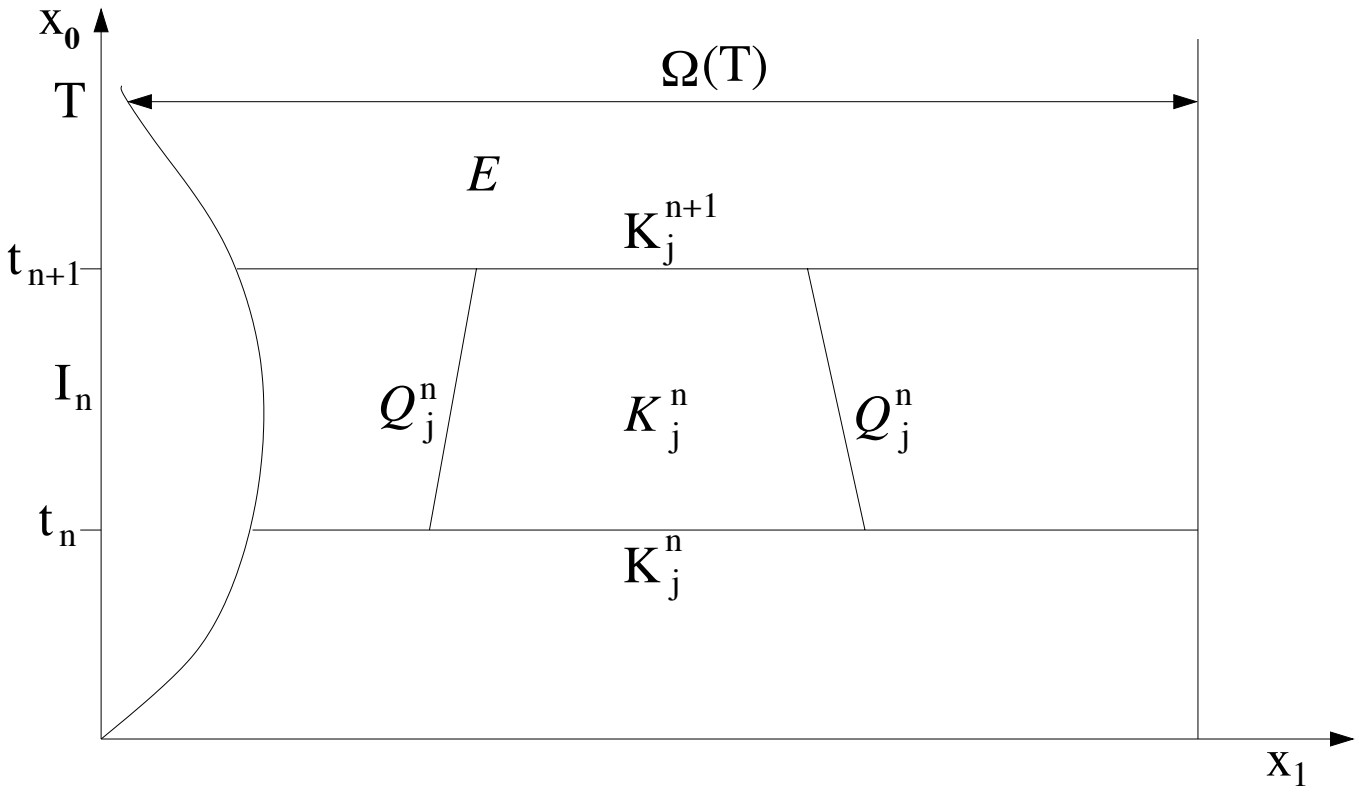
- The space-time domain boundary $\partial\mathcal{E}$ consists of the hypersurfaces

$$\begin{aligned}
\Omega(t_0) :=& \{x \in \partial\mathcal{E} \mid x_0 = t_0\},\\
\Omega(T) :=& \{x \in \partial\mathcal{E} \mid x_0 = T\},\\
\mathcal{Q} :=& \{x \in \partial\mathcal{E} \mid t_0 < x_0 < T\}.
\end{aligned}$$

Definition of Space-Time Slab

- Consider a partitioning of the time interval (t_0, T): $\{t_n\}_{n=0}^{N}$, and set $I_n = (t_n, t_{n+1})$.
- Define a space-time slab as $\mathcal{I}_n := \{x \in \mathcal{E} \mid x_0 \in I_n\}$.
- Split the space-time slab into non-overlapping elements $\mathcal{K}_j^n$.
- We will also use the notation $K_j^n = \mathcal{K}_j^n \cap \{t_n\}$ and $K_j^{n+1} = \mathcal{K}_j^n \cap \{t_{n+1}\}$ and $\mathcal{Q}_j^n = \partial\mathcal{K}_j^n \setminus (K_j^n \cup K_j^{n+1})$.

Space-Time Slab



Space-time slab in the space-time domain $\mathcal{E}$.

Compressible Navier-Stokes Equations

- Compressible Navier-Stokes equations in space-time domain $\mathcal{E}$

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

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

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

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 τ is defined as

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

- The dynamic viscosity coefficient μ given by Sutherland's law

$$\frac{\mu}{\mu_\infty} = \frac{T_\infty + T_S}{T + T_S} \left(\frac{T}{T_\infty} \right)^{3/2},$$

where T is the temperature, T_S a constant and $(\cdot)_\infty$ denotes free-stream values.

- The second viscosity coefficient λ is related to μ following the Stokes hypothesis $3\lambda + 2\mu = 0$.

Compressible Navier-Stokes Equations

- The heat flux vector q is defined as

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

with κ the thermal conductivity coefficient.

- The system is closed using the equations of state for a calorically perfect gas.

$$p = (\gamma - 1)(\rho E - \frac{1}{2} u_i u_i), \quad T = \frac{1}{c_v}(E - \frac{1}{2} u_i u_i),$$

with $\gamma = c_p / c_v$.

Compressible Navier-Stokes Equations

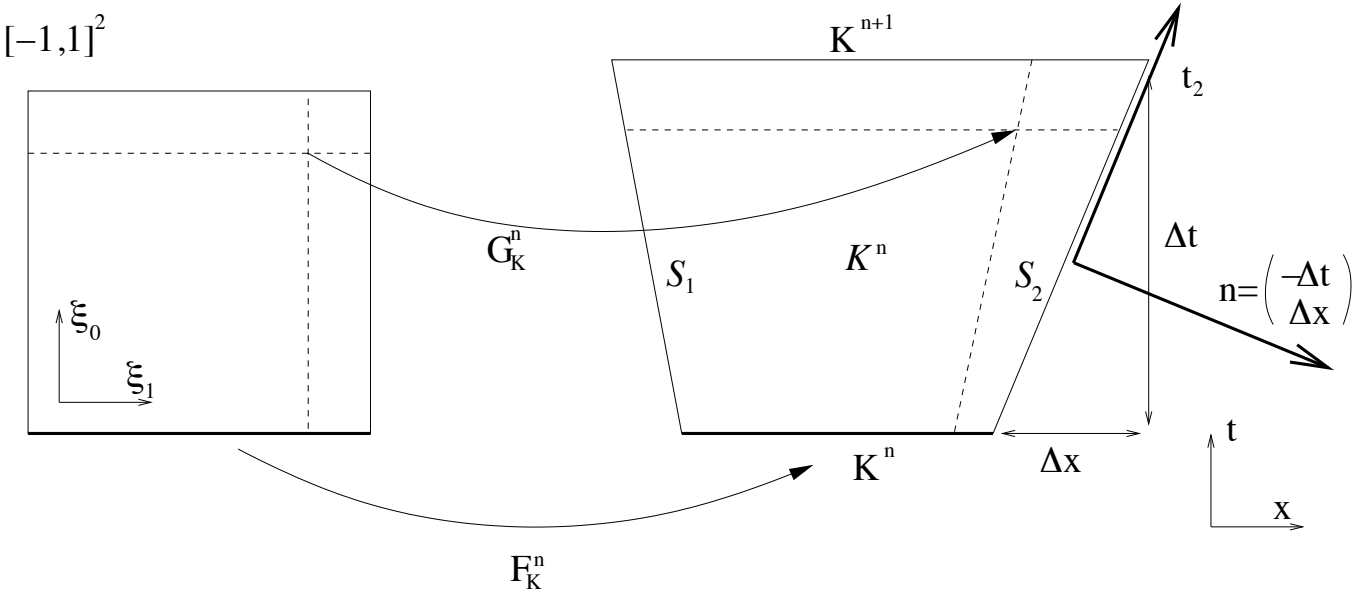
- The viscous flux F^v is homogeneous with respect to the gradient of the conservative variables ∇U

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

with the homogeneity tensor $A \in \mathbb{R}^{5\times3\times5\times3}$ defined as

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

Geometry of Space-Time Element



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

Space-Time Element Definition

- Definition of the mapping G_K^n, which the connects the space-time element $\mathcal{K}^n$ to the reference element $\hat{\mathcal{K}} = [-1,1]^4$

$$G_K^n : [-1,1]^4 \to \mathcal{K}^n : \xi \longmapsto x,$$

with

$$(x_0, \bar{x}) = \Big(\tfrac{1}{2}(t_n + t_{n+1}) + \tfrac{1}{2}(t_n - t_{n+1})\xi_0,$$

$$\tfrac{1}{2}(1-\xi_0)F_K^n(\bar{\xi}) + \tfrac{1}{2}(1+\xi_0)F_K^{n+1}(\bar{\xi}) \Big).$$

- Here $F_K^n : [-1,1]^3 \to K^n$, $F_K^{n+1} : [-1,1]^3 \to K^{n+1}$ are the mappings for the space elements, with

$$F_K^n : \hat{K} \to K^n : \bar{\xi} \mapsto \bar{x} = \sum_{i=1}^{8} x_i(K^n)\chi_i(\bar{\xi}),$$

with $x_i(K^n) \in \mathbb{R}^3$, $1 \le i \le 8$, the spatial coordinates of the vertices of the hexahedron K^n at time t_n.

- For F_K^{n+1} we have a similar expression using the vertices at $t = t_{n+1}$.

Space-Time Element Definition

- The space-time tessellation is now defined as

$$\mathcal{T}_h^n := \{\mathcal{K} = G_k^n(\hat{\mathcal{K}}) \mid K \in \bar{\mathcal{T}}_h^n\},$$

  with $\bar{\mathcal{T}}_h^n$ the tessellation of $\Omega(t_n)$.

- The space-time normal vector at an element boundary point moving with velocity v is given by

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

Approximation Spaces

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

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

- We will also use the space

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

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

Trace Operators

- The jump of f in the Cartesian coordinate direction k is defined at internal faces as

$$[\![f]\!]_k = f^L n_k^L + f^R n_k^R.$$

- The average of f is defined at internal faces as

$$\{\!\{f\}\!\} = \tfrac{1}{2}(f^L + f^R).$$

- The jump operator satisfies the following product rule at internal faces

$$[\![g_i f_{ik}]\!]_k = \{\!\{g_i\}\!\}[\![f_{ik}]\!]_k + [\![g_i]\!]_k \{\!\{f_{ik}\}\!\},$$

- Relation between element boundary and face integrals

$$\sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{Q}} g_i^L f_{ik}^L \bar{n}_k^L \, d\mathcal{Q} = \sum_{\mathcal{S}\in\mathcal{S}_I^n} \int_{\mathcal{S}} [\![g_i f_{ik}]\!]_k \, d\mathcal{S} + \sum_{\mathcal{S}\in\mathcal{S}_B^n} \int_{\mathcal{S}} g_i^L f_{ik}^L \bar{n}_k^L \, d\mathcal{S}.$$

Space-Time Formulation Compressible Navier-Stokes Equations

- The compressible Navier-Stokes equations in the domain $\mathcal{E} \subset \mathbb{R}^4$ can be expressed as

$$\begin{cases} U_{i,0} + \frac{\partial F^e_{ik}}{\partial x_k} - \frac{\partial}{\partial x_k}\left(A_{ikrs}\frac{\partial U_r}{\partial x_s}\right) = 0 & \text{on } \mathcal{E}, \\ U = U_0 & \text{on } \Omega(t_0), \\ U = \mathcal{B}(U, U^b) & \text{on } \mathcal{Q}, \end{cases}$$

  for $i, r = 1, \ldots, 5$ and $k, s = 1, \ldots, 3$.

- The initial flow field is denoted by $U_0 : \Omega(t_0) \to \mathbb{R}^5$, with U_0 the initial condition.

- The boundary operator is denoted by $\mathcal{B} : \mathbb{R}^{5\times 5} \to \mathbb{R}^5$ and is a function of the internal data U and the boundary data U^b derived from the boundary conditions.

- At the far-field boundary, suitable in- and out-flow conditions can be derived using local characteristics.

First Order System

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

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

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

- Note, this results in 5×3 additional equations for auxiliary variables Θ, which will be eliminated later using a lifting operator.

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

$$-\sum_{\mathcal{K}\in\mathcal{T}_h}\int_{\mathcal{K}}\left(\frac{\partial W_i}{\partial x_0}U_i+\frac{\partial W_i}{\partial x_k}(F^e_{ik}-\Theta_{ik})\right)d\mathcal{K}$$
$$+\sum_{\mathcal{K}\in\mathcal{T}_h}\int_{\partial\mathcal{K}}W^L_i(\widehat{U}_i+\widehat{F}^e_{ik}-\widehat{\Theta}_{ik})n^L_k\,d(\partial\mathcal{K})=0,$$

$$\sum_{\mathcal{K}\in\mathcal{T}^n_h}\int_{\mathcal{K}}V_{ik}\Theta_{ik}\,d\mathcal{K}=\sum_{\mathcal{K}\in\mathcal{T}^n_h}\int_{\mathcal{K}}V_{ik}A_{ikrs}\frac{\partial U_r}{\partial x_s}\,d\mathcal{K}$$
$$+\sum_{\mathcal{K}\in\mathcal{T}^n_h}\int_{\mathcal{Q}}V^L_{ik}A^L_{ikrs}(\widehat{U}_r-U^L_r)\bar{n}^L_s\,d\mathcal{Q}.$$

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

$$\begin{aligned} \sum_{\mathcal{K}\in\mathcal{T}_h} \int_{\partial\mathcal{K}} W_i^L (\widehat{U}_i + \widehat{F}_{ik}^e - \widehat{\Theta}_{ik}) n_k^L \, d(\partial\mathcal{K}) \\ = \sum_{K\in\mathcal{T}_h} \Big(\int_{K(t_{n+1}^-)} W_i^L \widehat{U}_i \, dK + \int_{K(t_n^+)} W_i^L \widehat{U}_i \, dK \Big) \\ + \sum_{K\in\mathcal{T}_h} \int_{\mathcal{Q}} W_i^L (\widehat{F}_{ik}^e - \widehat{U}_i v_k - \widehat{\Theta}_{ik}) \bar{n}_k^L \, d\mathcal{Q}. \end{aligned}$$

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

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

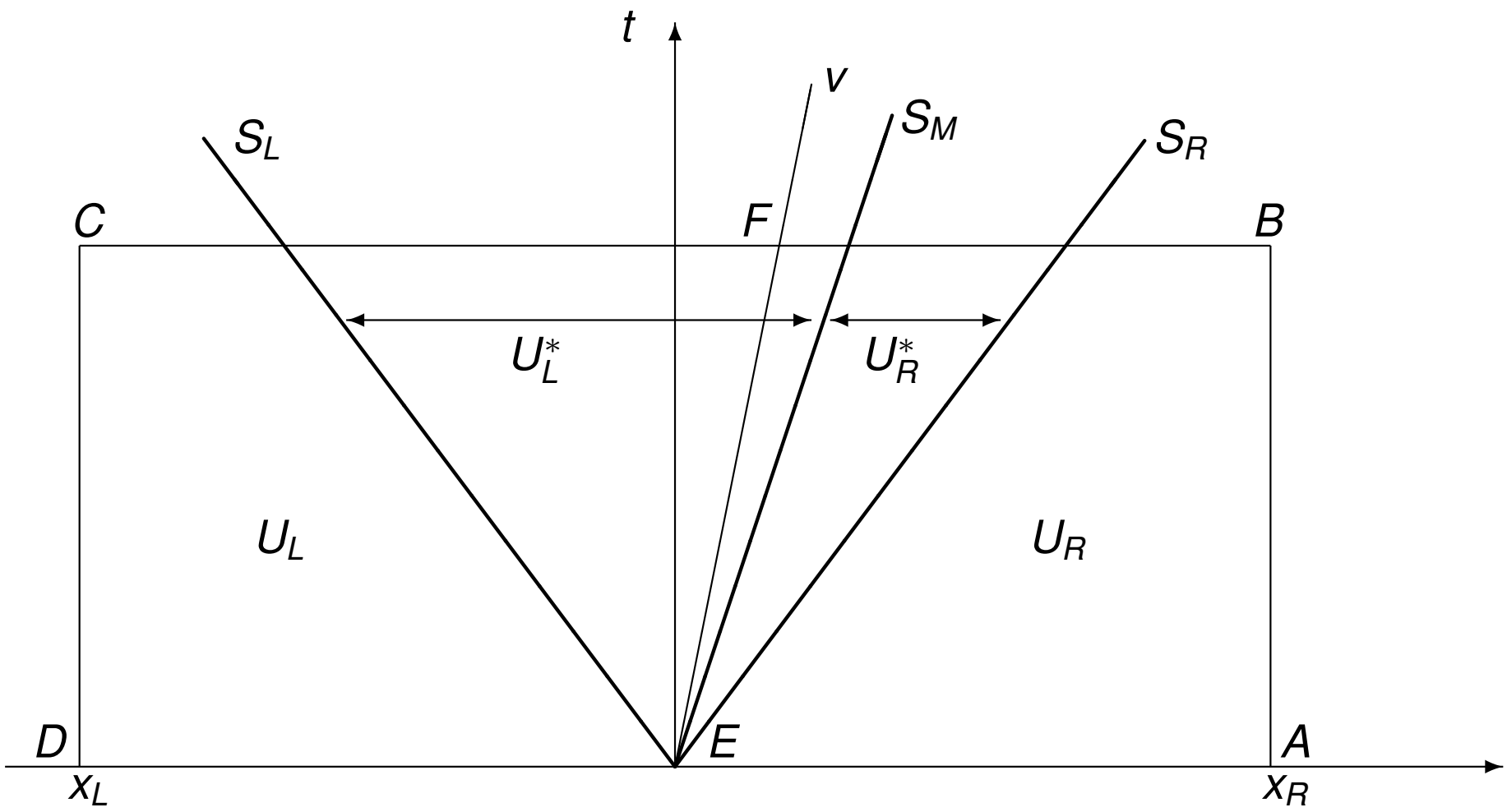
- At the space-time faces $\mathcal{Q}$ we introduce the HLLC approximate Riemann solver as numerical flux

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

HLLC Flux on Moving Meshes

- The HLLC scheme was proposed by Toro (Shock Waves **4**(1994), 25) and Batten e.a. (SIAM J. Sci. Stat. Comput. **18**(1997), 1553), but only for non-moving meshes.
- The extension to moving meshes is most easily accomplished by considering the structure of the wave pattern in the Riemann problem that is assumed in the HLLC scheme.

HLLC Flux on Moving Meshes



Wave pattern in HLLC-flux

HLLC Flux on Moving Meshes

- The HLLC scheme assumes we have two averaged intermediate states U_L^* and U_R^* in the star region, which is the region bounded by the waves with the slowest and fastest signal speeds S_L and S_R, respectively.

- The star region is divided into two parts by a contact wave, which moves with velocity S_M.

- Outside the star region the solution still is at its initial values at time t_m, which are denoted U_L and U_R and are equal to the traces $U_h^-(t_m)$ and $U_h^+(t_m)$, respectively.

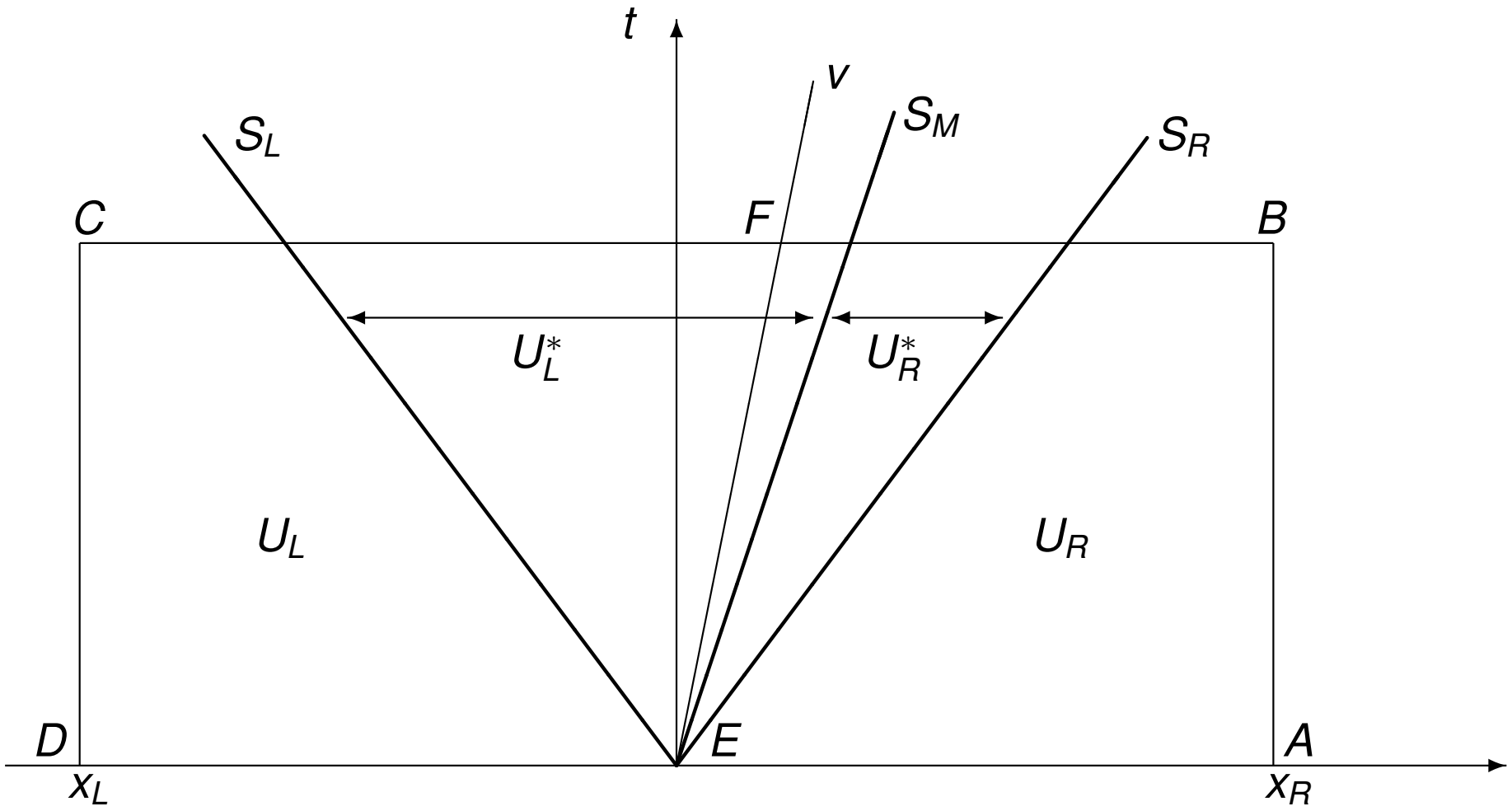
- In the time interval $[t_m, t_m + \triangle t)$ the solution U_{HLLC} at an element face which moves with the velocity v then is equal to

$$U_{HLLC} = \begin{cases} U_L \equiv U_h^-(t_m) & \text{if } S_L > v, \\ U_L^* & \text{if } S_L \leq v < S_M, \\ U_R^* & \text{if } S_M \leq v < S_R, \\ U_R \equiv U_h^+(t_m) & \text{if } S_R \leq v, \end{cases}$$

  where depending on the grid velocity v we have to consider four different cases.

- The time interval $\triangle t$ is chosen such that there is no interaction with waves coming from other Riemann problems.

HLLC Flux on Moving Meshes



Wave pattern in HLLC-flux

HLLC Flux on Moving Meshes

- Assume that $S_L < v$, $S_R > v$, and $S_M \geq v$, then we can calculate the flux $H_{HLLC}(U_L, U_R)$ in the time interval $[t_m, t_m + \triangle t)$ by integrating the Euler equations over the control volumes $\square DEFC$ and $\square EABF$.

  Using Gauss' theorem we obtain for the control volume $\square DEFC$ the relation

$$\int_{x_L}^{S_L \triangle t} U_L \, dx + \int_{S_L \triangle t}^{v \triangle t} U_h(x, t_m + \triangle t) \, dx$$

$$= \int_{x_L}^{0} U_h(x, t_m) \, dx + \int_{t_m}^{t_m + \triangle t} \hat{F}(U_h(x_L, t)) \, dt - \int_{t_m}^{t_m + \triangle t} \hat{F}(U_h^-(vt, t)) \, dt, \quad (2)$$

  and for the control volume $\square EABF$

$$\int_{v \triangle t}^{S_M \triangle t} U_h(x, t_m + \triangle t) \, dx + \int_{S_M \triangle t}^{S_R \triangle t} U_h(x, t_m + \triangle t) \, dx + \int_{S_R \triangle t}^{x_R} U_R \, dx$$

$$= \int_{0}^{x_R} U_h(x, t_m) \, dx + \int_{t_m}^{t_m + \triangle t} \hat{F}(U_h^+(vt, t)) dt - \int_{t_m}^{t_m + \triangle t} \hat{F}(U_h(x_R, t)) \, dt, \quad (3)$$

  with $\hat{F}(U_h) = \bar{n}_{\mathcal{K}} \bar{F}(U_h)$.

HLLC Flux on Moving Meshes

- If we introduce the averaged solutions U_L^* and U_R^*, which are defined as

$$U_L^* = \frac{1}{(S_M - S_L)\triangle t} \int_{S_L \triangle t}^{S_M \triangle t} U_h(x, t_m + \triangle t) dx,$$

$$U_R^* = \frac{1}{(S_R - S_M)\triangle t} \int_{S_M \triangle t}^{S_R \triangle t} U_h(x, t_m + \triangle t) dx.$$

- Use the fact that $U_h^{\pm}$ is constant along the line $x = vt$ in the Riemann problem then we obtain after subtracting (2) from (3) the following expression for the HLLC flux at the interface in the time interval $[t_m, t_m + \triangle t)$

$$H_{HLLC}(U_L, U_R) = \tfrac{1}{2}(\hat{F}(U_L) + \hat{F}(U_R) + ((S_L - v) + (S_M - v))U_L^* +$$

$$((S_R - v) - (S_M - v))U_R^* - S_L U_L - S_R U_R).$$

- For the other three cases: $(S_L < v, S_R > v, S_M \leq v)$, $(S_L < v, S_R < v, S_M < v)$, and $(S_L > v, S_R > v, S_M > v)$ a similar analysis can be made.

HLLC Flux on Moving Meshes

- If we combine the four cases then we obtain the following expression for the HLLC flux at a moving interface in the time interval $[t_m, t_m + \triangle t)$

$$H_{HLLC}(U_L, U_R) = \frac{1}{2}\big(\hat{F}(U_L) + \hat{F}(U_R) - (|S_L - v| - |S_M - v|)U_L^* + (|S_R - v| - |S_M - v|)U_R^* + |S_L - v|U_L - |S_R - v|U_R - v(U_L + U_R)\big).$$

- In order to completely define the HLLC flux we still need to define the star states U_L^* and U_R^*, and the wave speeds S_L, S_R and S_M.

- This can be done in various ways, but since there is no difference with the HLLC scheme for non-moving meshes, we only state the final results.

HLLC Flux on Moving Meshes

- We will follow the approach of Batten e.a. (SIAM J. Sci. Stat. Comput. **18**(1997), 1553) which assumed that

$$S_M = \hat{u}_L^* = \hat{u}_R^* = \hat{u}^*,$$

with $\hat{u}_{L,R} = \bar{n}_{\mathcal{K}} \cdot u_{L,R}$, and $\hat{u}^*$ the normal velocity calculated from the HLL approximation.

- This results in the following expression for S_M

$$S_M = \frac{\rho_R \hat{u}_R (S_R - \hat{u}_R) - \rho_L \hat{u}_L (S_L - \hat{u}_L) + p_L - p_R}{\rho_R (S_R - \hat{u}_R) - \rho_L (S_L - \hat{u}_L)}.$$

- The star states are obtained using the Rankine-Hugoniot relations across the waves moving with the velocities S_L and S_R

$$U_L^* = \frac{S_L - \hat{u}_L}{S_L - S_M} U_L + \frac{1}{S_L - S_M} \begin{pmatrix} 0 \\ (p^* - p_L)\bar{n}_{\mathcal{K}} \\ p^* S_M - p_L \hat{u}_L \end{pmatrix},$$

with an identical relation for U_R^*, only with L replaced with R.

HLLC Flux on Moving Meshes

- The intermediate pressures are equal to

$$p_L^* = \rho_L(S_L - \hat{u}_L)(S_M - \hat{u}_L) + p_L,$$
$$p_R^* = \rho_R(S_R - \hat{u}_R)(S_M - \hat{u}_R) + p_R.$$

- The definition of S_M ensures that $p_L^* = p_R^* = p^*$, as is required for a contact discontinuity.

- The wave speeds S_L and S_R are computed according as

$$S_L = \min(\hat{u}_L - a_L, \hat{u}_R - a_R), \qquad S_R = \max(\hat{u}_L + a_L, \hat{u}_R + a_R),$$

with $a = \sqrt{\gamma p/\rho}$ the speed of sound.

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

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

Auxiliary Equation for Θ

- Recall the auxiliary equation for Θ.

  Find a $\Theta \in V_h$, such that for all $V \in V_h$ the following holds

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

- The following relation holds for the element boundary integrals

$$\sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{Q}} g_i^L f_{ik}^L \bar{n}_k^L \, d\mathcal{Q} = \sum_{\mathcal{S}\in\mathcal{S}_I^n} \int_{\mathcal{S}} [\![g_i f_{ik}]\!]_k \, d\mathcal{S} + \sum_{\mathcal{S}\in\mathcal{S}_B^n} \int_{\mathcal{S}} g_i^L f_{ik}^L \bar{n}_k^L \, d\mathcal{S}.$$

- Transform the element boundary integrals into face integrals in the auxiliary equation

$$\sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{Q}} V_{ik}^L A_{ikrs}^L (\widehat{U}_r - U_r^L) \bar{n}_s^L \, d\mathcal{Q} = \sum_{\mathcal{S}\in\mathcal{S}_I^n} \int_{\mathcal{S}} [\![V_{ik} A_{ikrs} (\widehat{U}_r - U_r)]\!]_s \, d\mathcal{S}$$
$$+ \sum_{\mathcal{S}\in\mathcal{S}_B^n} \int_{\mathcal{S}} V_{ik}^L A_{ikrs}^L (\widehat{U}_r - U_r^L) \bar{n}_s^L \, d\mathcal{S}.$$

Numerical Fluxes in Auxiliary Equation

- Introduce the numerical flux proposed by Bassi and Rebay

$$\widehat{U} = \begin{cases} \{\!\{U\}\!\} & \text{at internal faces,} \\ U^b & \text{at boundary faces.} \end{cases}$$

- Use the relation

$$[\![g_i f_{ik}]\!]_k = \{\!\{g_i\}\!\}[\![f_{ik}]\!]_k + [\![g_i]\!]_k \{\!\{f_{ik}\}\!\},$$

then we obtain

$$[\![V_{ik} A_{ikrs}(\widehat{U}_r - U_r)]\!]_s = -\{\!\{V_{ik} A_{ikrs}\}\!\}[\![U_r]\!]_s.$$

- The weak formulation for the auxiliary variable Θ then becomes

$$\sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{K}} V_{ik}\Theta_{ik}\, d\mathcal{K} = \sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{K}} V_{ik} A_{ikrs} \frac{\partial U_r}{\partial x_s}\, d\mathcal{K} - \sum_{\mathcal{S}\in\mathcal{S}_I^n} \int_{\mathcal{S}} \{\!\{V_{ik} A_{ikrs}\}\!\}[\![U_r]\!]_s\, d\mathcal{S}$$
$$- \sum_{\mathcal{S}\in\mathcal{S}_B^n} \int_{\mathcal{S}} V_{ik}^L A_{ikrs}^L (U_r^L - U_r^b)\bar{n}_s^L\, d\mathcal{S}.$$

Lifting Operator

- Introduce the global lifting operator $\mathcal{R} \in \mathbb{R}^{5\times 3}$, defined in a weak sense as

  Find an $\mathcal{R} \in V_h$, such that for all $V \in V_h$

$$\sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{K}} V_{ik}\mathcal{R}_{ik}\, d\mathcal{K} = \sum_{\mathcal{S}\in\mathcal{S}_I^n} \int_{\mathcal{S}} \{\!\{ V_{ik} A_{ikrs} \}\!\} [\![U_r]\!]_s \, d\mathcal{S}$$
$$+ \sum_{\mathcal{S}\in\mathcal{S}_B^n} \int_{\mathcal{S}} V_{ik}^L A_{ikrs}^L (U_r^L - U_r^b) \bar{n}_s^L \, d\mathcal{S}.$$

- The weak formulation for the auxiliary variable is now transformed into

$$\sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{K}} V_{ik}\Theta_{ik}\, d\mathcal{K} = \sum_{\mathcal{K}\in\mathcal{T}_h^n} \int_{\mathcal{K}} V_{ik}(A_{ikrs}\frac{\partial U_r}{\partial x_s} - \mathcal{R}_{ik})\, d\mathcal{K}, \quad \forall V \in V_h.$$

Θ Equation

- The primal formulation can be obtained by eliminating the auxiliary variable Θ using

$$\Theta_{ik} = A_{ikrs} \frac{\partial U_r}{\partial x_s} - \mathcal{R}_{ik}, \qquad \text{a.e. in } \mathcal{E}_h^n.$$

- Note, this is possible since $\nabla_h W_h \subset V_h$.

ALE Weak Formulation for Primal Variables

- Recall the ALE flux formulation of the compressible Navier-Stokes equations

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

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

Numerical Fluxes for Θ

- The numerical flux $\widehat{\Theta}$ in the primary equation is defined following Brezzi as a central flux $\widehat{\Theta} = \{\!\{\Theta\}\!\}$

$$\widehat{\Theta}_{ik}(U^L, U^R) = \begin{cases} \{\!\{A_{ikrs}\frac{\partial U_r}{\partial x_s} - \eta\mathcal{R}^{\mathcal{S}}_{ik}\}\!\} & \text{for internal faces,} \\ A^b_{ikrs}\frac{\partial U^b_r}{\partial x_s} - \eta\mathcal{R}^{\mathcal{S}}_{ik} & \text{for boundary faces,} \end{cases}$$

- The local lifting operator $\mathcal{R}^{\mathcal{S}} \in \mathbb{R}^{5\times 3}$ is defined as follows

  Find an $\mathcal{R}^{\mathcal{S}} \in V_h$, such that for all $V \in V_h$

$$\sum_{\mathcal{K}\in\mathcal{T}^n_h}\int_{\mathcal{K}} V_{ik}\mathcal{R}^{\mathcal{S}}_{ik}\,d\mathcal{K} = \begin{cases} \displaystyle\int_{\mathcal{S}} \{\!\{V_{ik}A_{ikrs}\}\!\}[\![U_r]\!]_s\,d\mathcal{S} & \text{for internal faces,} \\ \displaystyle\int_{\mathcal{S}} V^L_{ik}A^L_{ikrs}(U^L_r - U^b_r)\bar{n}_s\,d\mathcal{S} & \text{for external faces.} \end{cases}$$

Space-Time Formulation for Compressible Navier-Stokes Equations

Find a $U \in W_h$, such that for all $W \in W_h$

$$
\begin{aligned}
&- \sum_{\mathcal{K} \in \mathcal{T}_h^n} \int_{\mathcal{K}} \Big(\frac{\partial W_i}{\partial x_0} U_i + \frac{\partial W_i}{\partial x_k} \big(F_{ik}^e - A_{ikrs} \frac{\partial U_r}{\partial x_s} + \mathcal{R}_{ik} \big) \Big) \, d\mathcal{K} \\
&+ \sum_{K \in \mathcal{T}_h^n} \Big(\int_{K(t_{n+1}^-)} W_i U_i^L \, dK - \int_{K(t_n^+)} W_i U_i^R \, dK \Big) \\
&+ \sum_{\mathcal{S} \in \mathcal{S}_{IB}^n} \int_{\mathcal{S}} (W_i^L - W_i^R) H_i(U^L, U^R, v, \bar{n}^L) \, d\mathcal{S} \\
&- \sum_{\mathcal{S} \in \mathcal{S}_I^n} \int_{\mathcal{S}} [\![W_i]\!]_k \{\!\{ A_{ikrs} \frac{\partial U_r}{\partial x_s} - \eta \mathcal{R}_{ik}^{\mathcal{S}} \}\!\} \, d\mathcal{S} \\
&- \sum_{\mathcal{S} \in \mathcal{S}_B^n} \int_{\mathcal{S}} W_i^L \big(A_{ikrs}^b \frac{\partial U_r^b}{\partial x_s} - \eta \mathcal{R}_{ik}^{\mathcal{S}} \big) \bar{n}_k^L \, d\mathcal{S} = 0,
\end{aligned}
$$

Basis Functions

- The basis functions are polynomials of degree k to represent the trial function U and the test function W in each element $\mathcal{K} \in \mathcal{T}_h^n$:

$$U_i(t, \bar{x})|_{\mathcal{K}} = \hat{U}_{im}\psi_m(t, \bar{x}),$$
$$W_l(t, \bar{x})|_{\mathcal{K}} = \psi_l(t, \bar{x}).$$

  with ψ the basis functions.

- The basis functions are defined such that the test and trial functions are split into an element mean at time t_{n+1} and a fluctuating part.

- This construction facilitates the definition of the artificial dissipation operator and of the multigrid convergence acceleration method.

- The basis functions ψ are given by

$$\begin{aligned} \psi_m &= 1, && m = 0, \\ &= \phi_m(t, \bar{x}) - \frac{1}{|K_j(t_{n+1}^-)|} \int_{K_j(t_{n+1}^-)} \phi_m(t, \bar{x})\, dK, && m = 1, \ldots, N, \end{aligned}$$

  where the basis functions ϕ are given by

$$\phi_m = \hat{\phi}_m \circ G_{\mathcal{K}}^{-1} \quad \text{with} \quad \hat{\phi}_m(\xi) \in P^k(\hat{\mathcal{K}}),$$

  with ξ the local coordinates in the master element $\hat{\mathcal{K}}$.

Lifting operators

- The DG coefficients of global and local lifting operators need to be expressed in terms of the DG coefficients of the primal variable U.
- Recall the expression for the lifting operator

$$\sum_{\mathcal{K}\in\mathcal{T}_h^n}\int_{\mathcal{K}} W_{i,k}\mathcal{R}_{ik}\,d\mathcal{K} = \sum_{\mathcal{S}\in\mathcal{S}_I^n}\int_{\mathcal{S}} \{\!\{W_{i,k}A_{ikrs}\}\!\}[\![U_r]\!]_s\,d\mathcal{S} + \sum_{\mathcal{S}\in\mathcal{S}_B^n}\int_{\mathcal{S}} W_{i,k}^L A_{ikrs}^L (U_r^L - U_r^b)\bar{n}_s^L\,d\mathcal{S}.$$

- The face integrals can be directly computed by replacing the test and trial functions by their polynomial expansions.

Lifting operators

- The local lifting are similarly expressed as

$$\mathcal{R}^{\mathcal{S}}(t,\bar{x})|_{\mathcal{K}} = \hat{R}_j \psi_j(t,\bar{x}).$$

  and a small linear system must be solved for the expansion coefficients $\hat{R}_j$.

- The local lifting operator is only non-zero on the two elements $\mathcal{K}^L$ and $\mathcal{K}^R$ connected to the face $\mathcal{S} \in \mathcal{S}_I^n$, hence

$$\int_{\mathcal{K}^R} V_{ik}\mathcal{R}_{ik}^{\mathcal{S}}\, d\mathcal{K} + \int_{\mathcal{K}^L} V_{ik}\mathcal{R}_{ik}^{\mathcal{S}}\, d\mathcal{K} = \int_{\mathcal{S}} \{\!\{ V_{ik}A_{ikrs} \}\!\} [\![U_r]\!]_s \, d\mathcal{S}.$$

- This is equivalent with the two following equations:

$$\int_{\mathcal{K}^{L,R}} V_{ik}\mathcal{R}_{ik}^{\mathcal{S}}\, d\mathcal{K} = \tfrac{1}{2}\int_{\mathcal{S}} V_{ik}^{L,R} A_{ikrs}^{L,R} [\![U_r]\!]_s \, d\mathcal{S},$$

  where the superscript L, R refers to the traces from either the left or right element.

Lifting operators

- Replacing $\mathcal{R}^{\mathcal{S}}$ by its polynomial approximation leads to two systems of linear equations for the expansion coefficients $\hat{R}_{ikj}$ of $\mathcal{R}^{\mathcal{S}}_{ik}$ on $\mathcal{S} \in \mathcal{S}_I$:

$$\hat{R}^{L,R}_{ikj} \int_{\mathcal{K}^{L,R}} \psi_l \psi_j \, d\mathcal{K} = \frac{1}{2} \int_{\mathcal{S}} \psi^{L,R}_l A^{L,R}_{ikrs} [\![U_r]\!]_s \, d\mathcal{S}.$$

- The element mass matrices on the l.h.s. are denoted by $M^{L,R}_{lj}$ and can easily be inverted leading to following expression for the expansion coefficients of the local lifting operator on $\mathcal{S} \in \mathcal{S}_I$:

$$\hat{R}^{L,R}_{ikj} = \frac{1}{2} (M^{-1})^{L,R}_{jl} \int_{\mathcal{S}} \psi^{L,R}_l A^{L,R}_{ikrs} [\![U_r]\!]_s \, d\mathcal{S}.$$

- Similarly, the expression for the expansion coefficients of the local lifting operator for the faces $\mathcal{S} \in \mathcal{S}_B$ is:

$$\hat{R}^{L}_{ikj} = (M^{-1})^{L}_{jl} \int_{\mathcal{S}} \psi^{L}_l A^{L}_{ikrs} (U^{L}_r - U^{b}_r) \bar{n}^{L}_s \, d\mathcal{S}.$$

- The expressions for the local lifting operator can now be introduced into the DG formulation, resulting in the primal formulation without auxiliary variables.

Ensuring Monotonicity of Second and Higher Order DG Discretizations

Second and higher-order DG discretizations do not preserve the monotonicity of the solution.

- Using a slope limiter in DG discretizations results in an inconsistent discretization with limit cycle behavior, which hampers implicit time integration methods and prevents convergence to steady state.
- The use of a slope limiter also seriously degrades the accuracy of a DG discretization.
- As an alternative the KKT-Limiting procedure could be used (vdVegt, Xia, Xu, SISC 2019), but this has not been done yet in the space-time context.

Stabilization Operator

- Stabilization operator for flow discontinuities 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 $A, B \in R^{n \times m}$.

- A stabilization operator results in a numerical scheme that can converge to steady state and has improved accuracy, but requires additional research to ensure monotonicity.

Stabilization Operator

- The effectiveness of the stabilization operator $\mathcal{D}$ strongly depends on the artificial viscosity matrix $\mathfrak{D} \in \mathbb{R}^{4\times 4}$.

- The definition of the artificial viscosity matrix is more straightforward if the stabilization operator acts independently in all computational coordinate directions.

- This is achieved by introducing the artificial viscosity matrix $\tilde{\mathfrak{D}} \in \mathbb{R}^{4\times 4}$ in computational space using the relation

$$\mathfrak{D}(U_h|_{\mathcal{K}^n}, U_h^*|_{\mathcal{K}^n}) = R^T\, \tilde{\mathfrak{D}}(U_h|_{\mathcal{K}^n}, U_h^*|_{\mathcal{K}^n})\, R,$$

  where the matrix $R \in \mathbb{R}^{4\times 4}$ is defined as

$$R = 2\, H^{-1}\, \mathrm{grad}\, G_K.$$

- The matrix $H \in \mathbb{R}^{4\times 4}$ is used introduced to ensure that both $\mathfrak{D}$ and $\tilde{\mathfrak{D}}$ have the same mesh dependence, and is defined as

$$H = \mathrm{diag}\,(h_0, h_1, h_2, h_3).$$

  with $h_i \in \mathbb{R}^+$ the leading terms in the expansion of the mapping G_K in the reference element coordinates $\xi_i (1 \leq i \leq 4)$.

Artificial Viscosity Model

- The integrals in the stabilization operator $\mathcal{D}_{nm}$ can now be further evaluated, resulting in:

$$\begin{aligned}\mathcal{D}_{nm}(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}) &= \int_{\mathcal{K}_j^n} \frac{\partial \psi_n}{\partial x_k} R_{pk} \tilde{\mathfrak{D}}_{pq}(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}) R_{ql} \frac{\partial \psi_m}{\partial x_l} d\mathcal{K} \\ &= 4 \int_{\hat{\mathcal{K}}} (H^{-1})_{pn} \tilde{\mathfrak{D}}_{pq}(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}) (H^{-1})_{qm} |J_{G_K}| d\hat{\mathcal{K}}, \\ &= \frac{4|\mathcal{K}_j^n|}{h_n^2} \delta_{nm} \tilde{\mathfrak{D}}_{nn}(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n})\end{aligned}$$

(no summation on n).

Here we used the relations: $(\mathrm{grad}\, G_K)_{ij} = \partial x_j / \partial \xi_i$ and $\partial \psi_n / \partial \xi_p = \delta_{np}$ and made the assumption that $\tilde{\mathfrak{D}}$ is constant in each element.

Artificial Viscosity Model

- Artificial viscosity model proposed and analyzed by Jaffre, Johnson and Szepessy, *Math. Models and Meth. in Appl. Sci*, **158**, pp. 81-116 (1998), is used.

- In this model both the jumps at the element faces and the element residual are used to define the artificial viscosity

$$\tilde{\mathfrak{D}}_{qq}(U_h|_{\mathcal{K}^n}, U_h^*|_{\mathcal{K}^n}) = \max\big(C_2 h_{\mathcal{K}}^{2-\beta} R_q(U_h|_{\mathcal{K}^n}, U_h^*|_{\mathcal{K}^n})\,,\; C_1 h_{\mathcal{K}}^{\frac{3}{2}}\big), \qquad q = 1, 2, 3,$$

$$= 0, \qquad \text{otherwise},$$

with

$$R(U_h|_{\mathcal{K}^n}, U_h^*|_{\mathcal{K}^n}) = \Big|\sum_{k=0}^{3} \frac{\partial \mathcal{F}(U_h)}{\partial U_{h,i}} \frac{\partial U_{h,i}(G_K(0))}{\partial x_k}\Big| + \frac{C_0}{h_{\mathcal{K}}} \big|U_h^+(x_{(7)}) - U_h^-(x_{(7)})\big| +$$

$$\sum_{m=1}^{6} \frac{1}{h_{\mathcal{K}}} \big|\bar{n}_{\mathcal{K}}^T \mathcal{F}(U_h^+(x_{(m)})) - \bar{n}_{\mathcal{K}}^T \mathcal{F}(U_h^-(x_{(m)}))\big|.$$

and $C_0 = 1.2$, $C_1 = 0.1$, $C_2 = 1.0$ and $\beta = 0.1$.

Pseudo-Time Integration

- The system of non-linear algebraic equations for the expansion coefficients in the DG discretization can be written as

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

  with $\mathcal{L} = \mathcal{L}^e + \mathcal{L}^v$.

- The system is solved by adding a pseudo-time derivative

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

  and integrating to steady state in pseudo-time.

- Alternatively, also a Newton method can be used.

Pseudo-Time Integration

- The algebraic equations are solved by combining different Runge-Kutta methods in the pseudo-time integration method.

- Explicit 5 stage Runge-Kutta method (EXI-method) with the correction proposed by Melson to enhance the stability of the pseudo-time integration.

  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 at stage s are denoted by α_s and defined as: $\alpha_1 = 0.0791451$, $\alpha_2 = 0.163551$, $\alpha_3 = 0.283663$, $\alpha_4 = 0.5$ and $\alpha_5 = 1.0$.

- The coefficients are optimized to ensure rapid convergence to steady state.

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

Pseudo-Time Integration

- Explicit 4 stage Runge-Kutta method (EXV), optimized 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 $\hat{U} = \hat{V}^4$.

- The Runge-Kutta coefficients at stage s are defined as: $\alpha_1 = 0.0178571$, $\alpha_2 = 0.0568106$, $\alpha_3 = 0.174513$ and $\alpha_4 = 1$.

- With these coefficients, the stability domain of the Runge-Kutta method is very different from the one associated with the classic 4 stage Runge-Kutta method for inviscid flows.

- Based on the local flow conditions, either EXI or EXV is used locally.

Pseudo-Time Integration

- Implicit-explicit (IMEX) version of the EXI method.
- The residual $\mathcal{L}$ consist of two parts: $\mathcal{L} = \mathcal{L}^e + \mathcal{L}^v$, where $\mathcal{L}^e$ stems from the inviscid part of the compressible Navier-Stokes equations and $\mathcal{L}^v$ from the viscous part.
- The implicit-explicit method can be derived by introducing a Newton matrix $\mathcal{D}$, which approximates the Jacobian of the viscous part of the residual:

$$\mathcal{D}\hat{V}^s \cong \mathcal{L}^v.$$

- Here, the approximation consists of freezing the (non-linear) homogeneity tensor A at the previous Runge-Kutta stage $s-1$.
- This approximation is relatively inexpensive compared with the Jacobian of the inviscid flux, which would be required by a Newton solver, since A is readily available in the discretization.

Pseudo-Time Integration

- Implicit-explicit (IMEX) Runge-Kutta method
  1. Initialize $\hat{V}^0 = \hat{U}$.
  2. For all stages $s = 1$ to 5 compute $\hat{V}^s$ by solving:
  $$\big(I + \alpha_s \lambda (I + \mathcal{D})\big)\hat{V}^s = \hat{V}^0 + \alpha_s \lambda \big((I + \mathcal{D})\hat{V}^{s-1} - \mathcal{L}(\hat{V}^{s-1}; \hat{U}^{n-1})\big).$$
  3. Return $\hat{U} = \hat{V}^5$.
- The coefficients α_s are the same as in EXI.
- The linear system is solved using the sparse iterative GMRES solver with Jacobi preconditioning.

Stability Pseudo-Time Integration

- Consider the scalar advection-diffusion equation

$$u_t + a\,u_x = d\,u_{xx}, \quad (x,t) \in \mathbb{R} \times \mathbb{R}^+,$$

  with $a > 0$ the advection constant and $d > 0$ the diffusion constant.

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

- The space-time discontinuous Galerkin discretization using linear basis functions is

$$\mathcal{L}(\hat{u}^n; \hat{u}^{n-1}) \equiv (\mathcal{L}^a + \mathcal{L}^d)\hat{u}^n + \mathcal{L}^t \hat{u}^{n-1} = 0,$$

Stability Pseudo-Time Integration

- The (block tridiagonal) inviscid part of the stencil depends on the Courant number:

$$\sigma = \frac{a\Delta t}{\Delta x},$$

and is given by:

$$\mathcal{L}^a = \left[\begin{array}{ccc|ccc|ccc} -\sigma & -\sigma & \sigma & 1+\sigma & \sigma & -\sigma & 0 & 0 & 0 \\ \sigma & \sigma & -\sigma & -\sigma & \frac{1}{3}+\sigma & \sigma & 0 & 0 & 0 \\ \sigma & \sigma & -\frac{4}{3}\sigma & -2-\sigma & -\sigma & 2+\frac{4}{3}\sigma & 0 & 0 & 0 \end{array}\right].$$

- The right block is zero because the advective numerical flux is upwind ($a > 0$).

Stability Pseudo-Time Integration

- The (block tridiagonal) viscous part of the stencil depends on the Von Neumann number:

$$\delta = \frac{d\Delta t}{(\Delta x)^2},$$

as well as on the stabilization constant η and is given by:

$$\mathcal{L}^d = \delta \left[\begin{array}{ccc|ccc|ccc} -2\eta & 1-2\eta & 2\eta & 4\eta & 0 & -4\eta & -2\eta & -1+2\eta & 2\eta \\ -1+2\eta & -2+2\eta & 1-2\eta & 0 & 4\eta & 0 & 1-2\eta & -2+2\eta & -1+2\eta \\ 2\eta & -1+2\eta & -\frac{13}{6}\eta & -4\eta & 0 & \frac{13}{3}\eta & 2\eta & 1-2\eta & -\frac{13}{6}\eta \end{array}\right].$$

- The (block diagonal) part of the stencil related to the previous space-time slab is given by:

$$\mathcal{L}^t = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -\frac{1}{3} & 0 \\ 2 & 0 & 0 \end{bmatrix}.$$

- Note that the matrix has a periodic block Toeplitz structure with 3×3 blocks, written symbolically as:

$$\mathcal{L} = \left[L \,\middle|\, D \,\middle|\, U\right].$$

Stability Pseudo-Time Integration

- The stability in pseudo-time is only affected by the transients, hence we only consider the homogeneous part of the linear system

$$\frac{\partial \hat{u}^n}{\partial \tau} = -\frac{1}{\Delta t}(\mathcal{L}^a + \mathcal{L}^d)\hat{u}^n.$$

- The vector of expansion coefficients in element j is assumed to be a Fourier mode:

$$\hat{u}_j^n = \hat{u}^F \exp(\imath\theta j)$$

with $\hat{u}^F$ the amplitude of the mode, $\imath = \sqrt{-1}$ and $\theta \in (-\pi, \pi]$.

- The Fourier transform of the discrete system becomes:

$$\mathbf{FT}(\mathcal{L})(\theta) = L\exp(-\imath\theta) + D + U\exp(\imath\theta),$$

with L the block-lower, D the block-diagonal and U the block-upper part of the matrix.

Stability Pseudo-Time Integration

- Notice that the matrix $\mathbf{FT}(\mathcal{L})(\theta)$ is non-singular and can be diagonalized as QMQ^{-1}.

  Q are the right eigenvectors and M a diagonal matrix with the (complex) eigenvalues $\mu_i(\theta)$ with $i = 1, 2, 3$.

- After introducing $w = Q^{-1}\hat{u}^n$ we obtain the scalar test model:

$$\frac{\partial w_i}{\partial \tau} = -\frac{\mu_i(\theta)}{\Delta t} w_i, \quad \text{for } i = 1, 2, 3, \text{ no summation on } i$$

- The Runge-Kutta stages w^s in the EXI method are computed as

$$(1 + \alpha_s \lambda) w^s = w^0 + \alpha_s \lambda (1 - \mu)\, w^{s-1},$$

  with $\lambda = \Delta\tau / \Delta t$.

- The Runge-Kutta stages w^s in the EXV method are computed as

$$w^s = w^0 - \alpha_s \lambda \mu\, w^{s-1}.$$

Stability Pseudo-Time Integration

- The relation between two consecutive pseudo-time steps can be derived written in generic form as:
$$w^n = G(-\lambda\mu)w^{n-1},$$
with G the algorithm dependent amplification factor.

- Starting with an initial condition w^{init}, we obtain after n steps:
$$w^n = G(-\lambda\mu)^n w^{\text{init}}.$$

- A sufficient condition for stability is that the values $-\lambda\mu_i(\theta)$ for $i = 1, 2, 3$ and $\theta \in (-\pi, \pi]$ all lie inside the stability domain S given by:
$$S = \{z \in \mathbb{C} \,:\, |G(z)| \leq 1\}.$$

Stability Pseudo-Time Integration

- Remember that the discretization of the advection-diffusion equation only depends on the Courant number, the Von Neumann number and the constant η.
- For given values of these numbers, the factor λ of the Runge-Kutta algorithm should be chosen such that $-\lambda\mu_i(\theta)$ lies inside the stability domain S.
- Once a suitable λ is found, it is convenient to express the stability in terms of the *pseudo-time* Courant and Von Neumann numbers: $\sigma_{\Delta\tau} = \lambda\sigma$ and $\delta_{\Delta\tau} = \lambda\delta$.
- For stability, the pseudo-time step $\Delta\tau$ must satisfy the Courant-Friedrichs-Levy (CFL) condition and the Von Neumann condition:

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

- We distinguish between flow regimes by introducing the *cell* Reynolds number, defined as:

$$\mathrm{Re}_{\Delta x} \equiv \frac{a\Delta x}{d}.$$

Stability Pseudo-Time Integration

- The stabilization parameter η has a significant effect on the stability of the pseudo-time integration:

  As η increases, the pseudo-time Von Neumann number decreases proportionally.

  η should be taken as small as allowed in the discontinuous Galerkin discretization, in general equal to the number of faces of an element.

- The Melson correction is applied to the EXI scheme to ensure stability for values of λ around one, which is the case for the time-dependent inviscid flow regime.

- For all other flow regimes, λ is small and the Melson correction vanishes.

Stability Pseudo-Time Integration

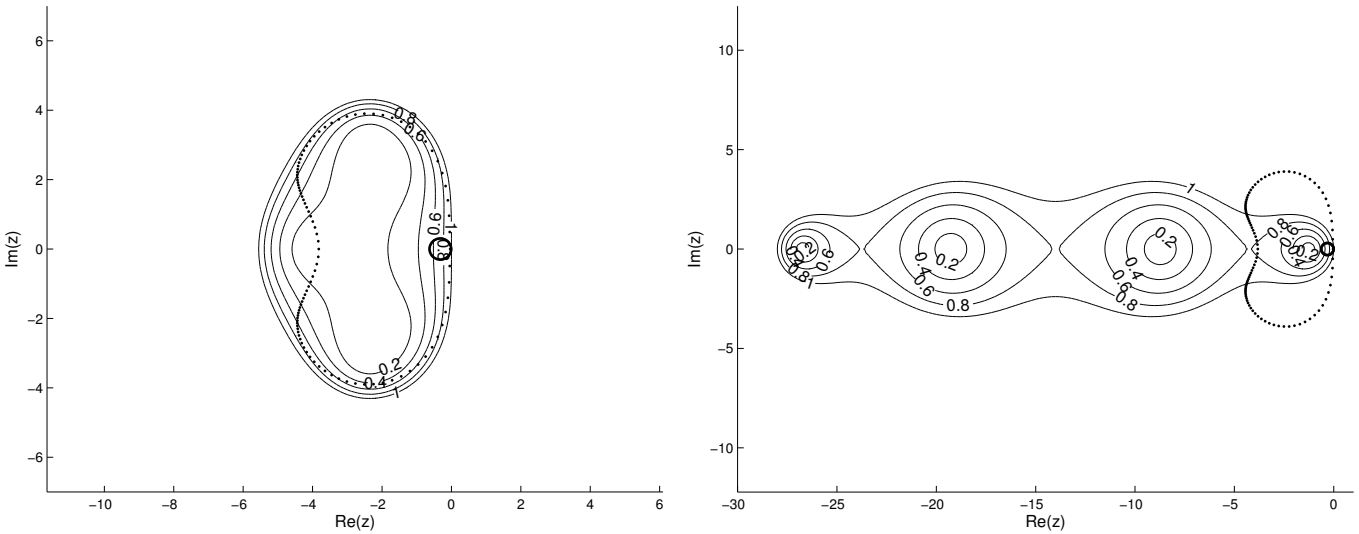


Figure: Stability domain S and values $-\lambda\mu_i$ (dots) for EXI (left) and EXV (right) in steady-state inviscid flow regime, $\lambda = 1.8 \cdot 10^{-2}$. Pseudo-time CFL number is 1.8. For this constraint only EXI is stable.

Stability Pseudo-Time Integration

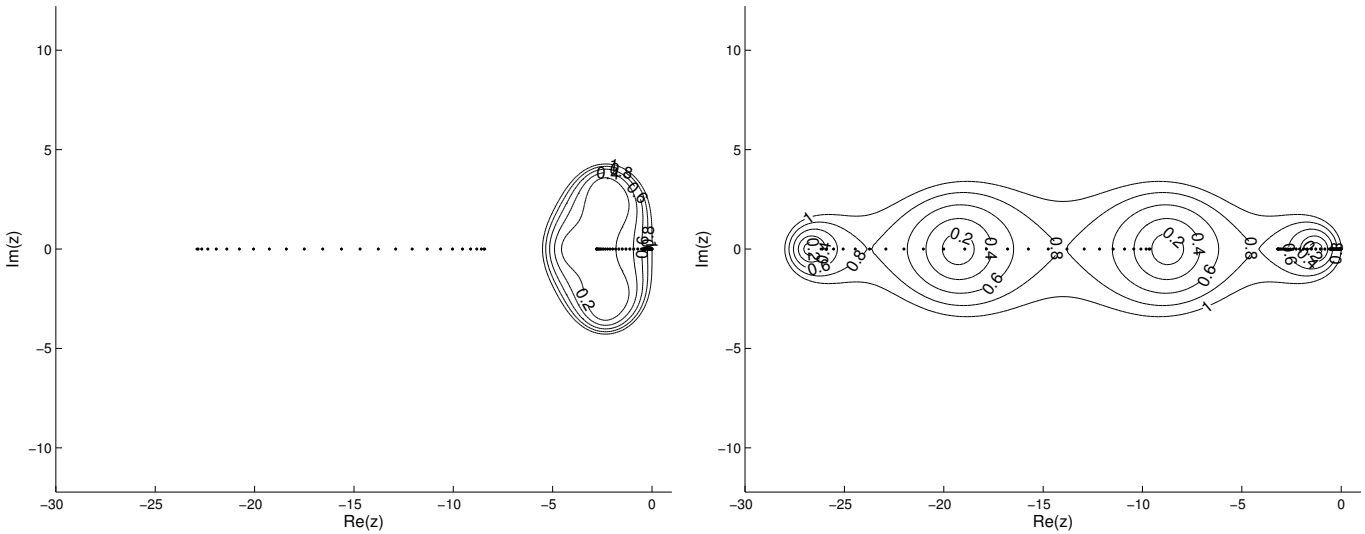


Figure: Stability domain S and values $-\lambda\mu_i$ (dots) for EXI (left) and EXV (right) in steady-state viscous flow regime, $\lambda = 8 \cdot 10^{-5}$. Pseudo-time Von Neumann number is 0.8. For this constraint only EXV method is stable.

Stability Pseudo-Time Integration

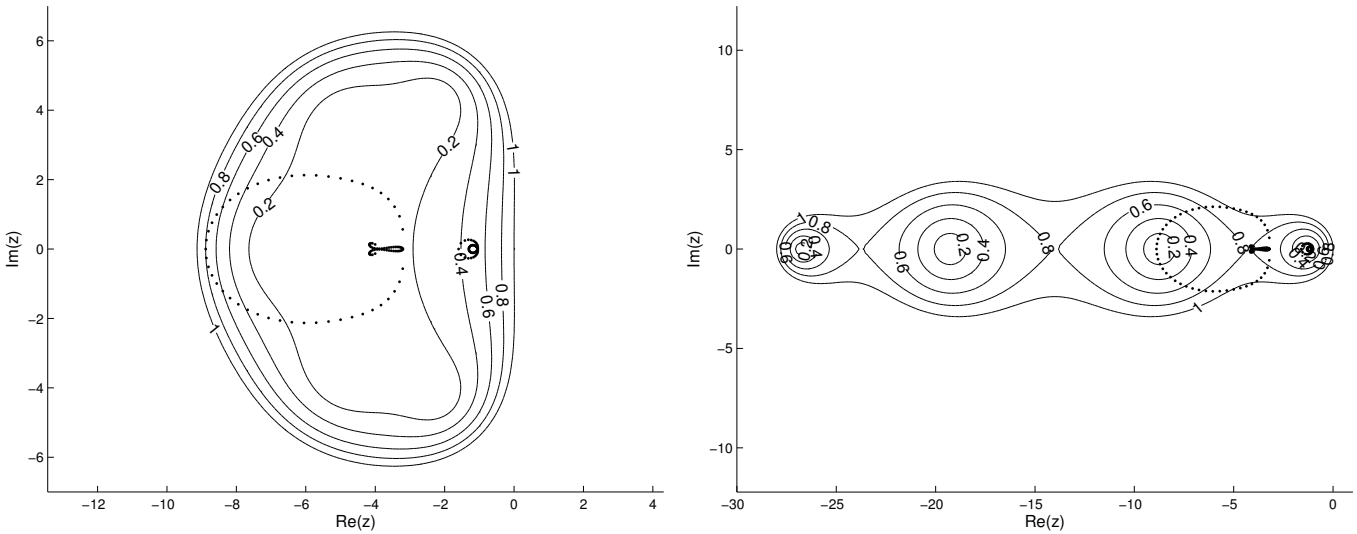


Figure: Stability domain S and values $-\lambda\mu_i$ (dots) for EXI (left) and EXV (right) in time-dependent inviscid flow regime, $\lambda = 1.6$. Pseudo-time CFL number is 1.6. For this constraint only EXI is stable.

Stability Pseudo-Time Integration

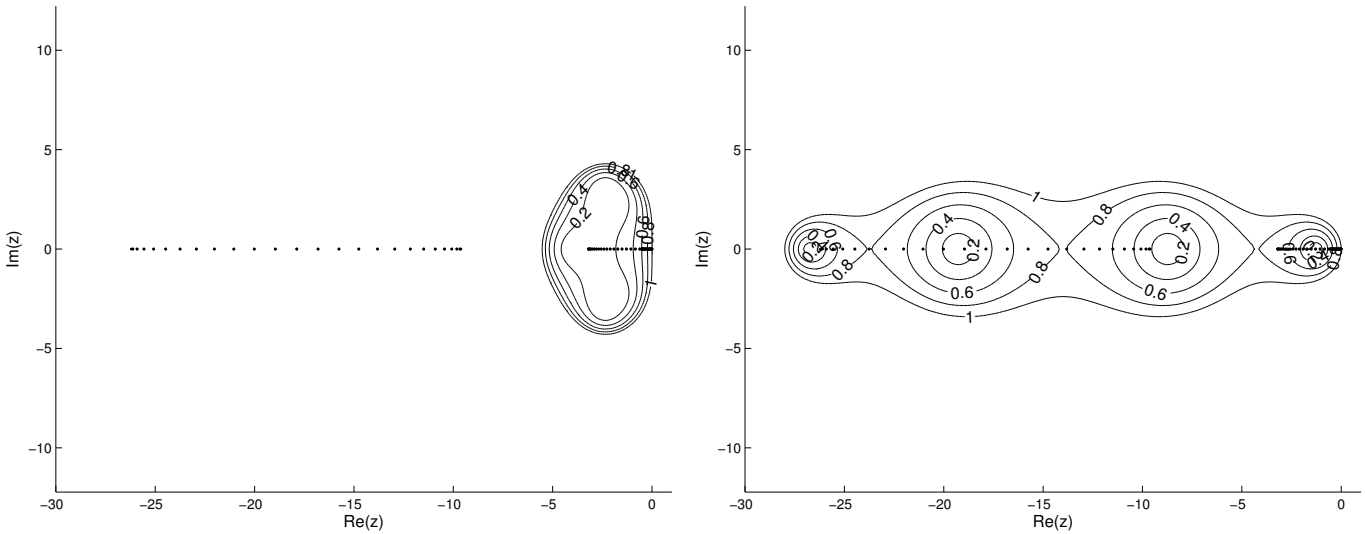


Figure: Stability domain S and values $-\lambda\mu_i$ (dots) for EXI (left) and EXV (right) in time-dependent viscous flow regime, $\lambda = 8 \cdot 10^{-3}$. Pseudo-time Von Neumann number is 0.8. For this constraint only EXV is stable.

Stability Pseudo-Time Integration

- The IMEX method solves the inviscid part of the equations with the EXI method and treats the viscous part implicitly.
- The main idea is that the stability should now only depend on the inviscid part, so only the CFL condition has to be satisfied.
- The matrices $\mathcal{L}^a$ and $\mathcal{L}^d$ in IMEX do not commute, which makes it impossible to obtain a scalar model problem through diagonalization.
- For the IMEX method the Runge-Kutta stages $\hat{v}^s$ are computed by solving the sparse linear system:

$$\left(I + \alpha_s \lambda (I + \mathcal{L}^d)\right) \hat{v}^s = \hat{v}^0 + \alpha_s \lambda (I - \mathcal{L}^a) \hat{v}^{s-1}.$$

Stability Pseudo-Time Integration

- The starting point of our analysis is the fact that $\mathcal{L}^d$ is a Hermitian matrix.

  Therefore $\mathcal{L}^d = QMQ^T$, where Q is a unitary matrix and M the diagonal matrix with the eigenvalues μ_i of $\mathcal{L}^d$.

- The eigenvalues of $\mathcal{L}^d$ are real and positive, and can be computed as the eigenvalues $\mu_i(\theta)$ with $i = 1, 2, 3$ and $\theta \in (-\pi, \pi]$ of the corresponding Fourier transform

$$\mathbf{FT}(\mathcal{L}^d)(\theta) = L^d \exp(-\imath\theta) + D^d + U^d \exp(\imath\theta).$$

  For a unitary matrix $Q^{-1} = Q^T$ and the l.h.s. can be written as:

$$\begin{aligned} I + \alpha_s\lambda(I + \mathcal{L}^d) &= Q\big(I + \alpha_s\lambda(I + M)\big)Q^T \\ &= QM_sQ^T, \end{aligned}$$

  with M_s the diagonal matrix with values $1 + \alpha_s\lambda(1 + \mu_i)$. Using the decomposition QM_sQ^T gives:

$$\begin{aligned} M_s w^s &= w^0 + \alpha_s\lambda Q^T(I - \mathcal{L}^a)Qw^{s-1}, \\ &= w^0 + \alpha_s\lambda P_a w^{s-1}, \end{aligned}$$

  with $w^s = Q^T\hat{v}^s$ and $P_a = Q^T(I - \mathcal{L}^a)Q$.

Stability Pseudo-Time Integration

- The relation between two consecutive pseudo-time steps is: $w^n = Gw^{n-1}$ with the amplification matrix G defined as:

$$G = M_5^{-1}(I + \alpha_5 \lambda P_a M_4^{-1}(I + \alpha_4 \lambda P_a \cdots M_1^{-1}(I + \alpha_1 \lambda P_a))).$$

- If $\|G\| \leq 1$, then $\|G^n\| \leq 1$ and the method is stable.

- Our stability analysis aims at a direct estimation of this norm, therefore we consider the following upper bound:

$$\begin{aligned}\|G\| \leq & \|M_5^{-1}\|(1 + \alpha_5 \lambda \|P_a\| \|M_4^{-1}\|(1 + \alpha_4 \lambda \|P_a\| \cdots \\ & \|M_1^{-1}\|(1 + \alpha_1 \lambda \|P_a\|))).\end{aligned}$$

- The matrices M_s^{-1} are equal to:

$$M_s^{-1} = \mathrm{diag}\Big(\frac{1}{1 + \alpha_s \lambda(1 + \mu_1)}, \cdots, \frac{1}{1 + \alpha_s \lambda(1 + \mu_n)}\Big),$$

with μ_i the eigenvalues of $\mathcal{L}^d$.

Stability Pseudo-Time Integration

- The Euclidian norm of M_s^{-1} can be estimated as:

$$\|M_s^{-1}\| = \max_{i\in\{1,\cdots,n\}} \frac{1}{1+\alpha_s\lambda(1+\mu_i)} < \frac{1}{1+\alpha_s\lambda},$$

  since $\mu_i, \alpha_s, \lambda > 0$.

- Using the estimate for M_s^{-1}, the upper bound for the Euclidian norm of G is then provided by the following estimate:

$$\|G\| \leq \frac{1}{1+\alpha_5\lambda}(1+\alpha_5\lambda\|P_a\|\frac{1}{1+\alpha_4\lambda}(1+\alpha_4\lambda\|P_a\|\ldots$$
$$\frac{1}{1+\alpha_1\lambda}(1+\alpha_1\lambda\|P_a\|))).$$

- The r.h.s. of this equation is called the stability function, denoted by $f(\lambda, \|P_a\|)$.

Stability Pseudo-Time Integration

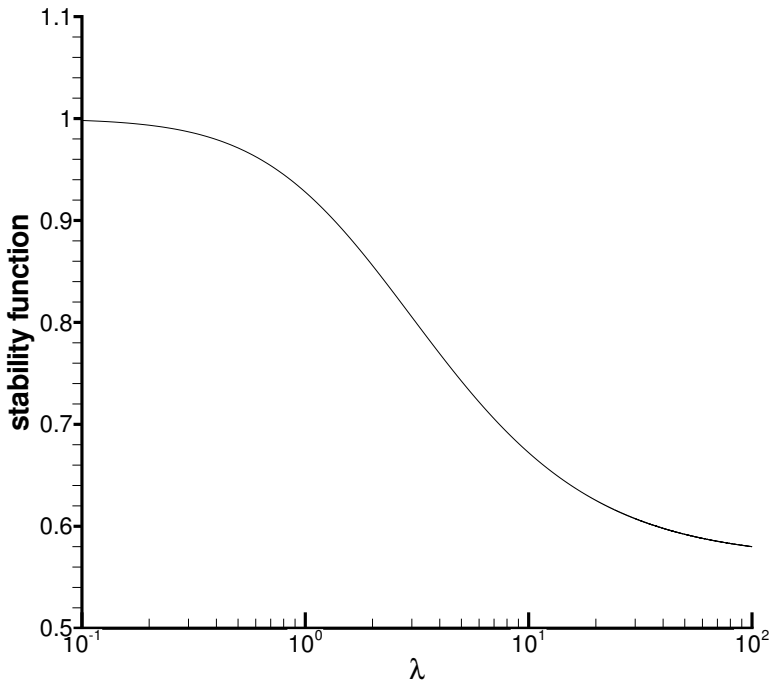


Figure: The stability function f for $\|P_a\| = 1$.

Stability Pseudo-Time Integration

- If $\|P_a\| < 1$ we find ourselves below the curve $f(\lambda, \|P_a\|)$.
- Therefore: $\|P_a\| \leq 1 \Rightarrow f(\lambda, \|P_a\|) \leq 1 \Rightarrow \|G\| \leq 1$ meaning $\|P_a\| \leq 1$ is a sufficient condition for stability of the implicit-explicit method.
- Since the matrix P_a is defined as $P_a = Q^T(I - \mathcal{L}^a)Q$, with Q a unitary matrix (hence $\|Q\| = 1$), this implies that the stability of the IMEX method is only determined by the following condition:

$$\|I - \mathcal{L}^a\| \leq 1.$$

- Since $\mathcal{L}^a$ only depends on the Courant number, this condition implies that the IMEX method stability only depends on the CFL condition.

Pseudo-Time Integration

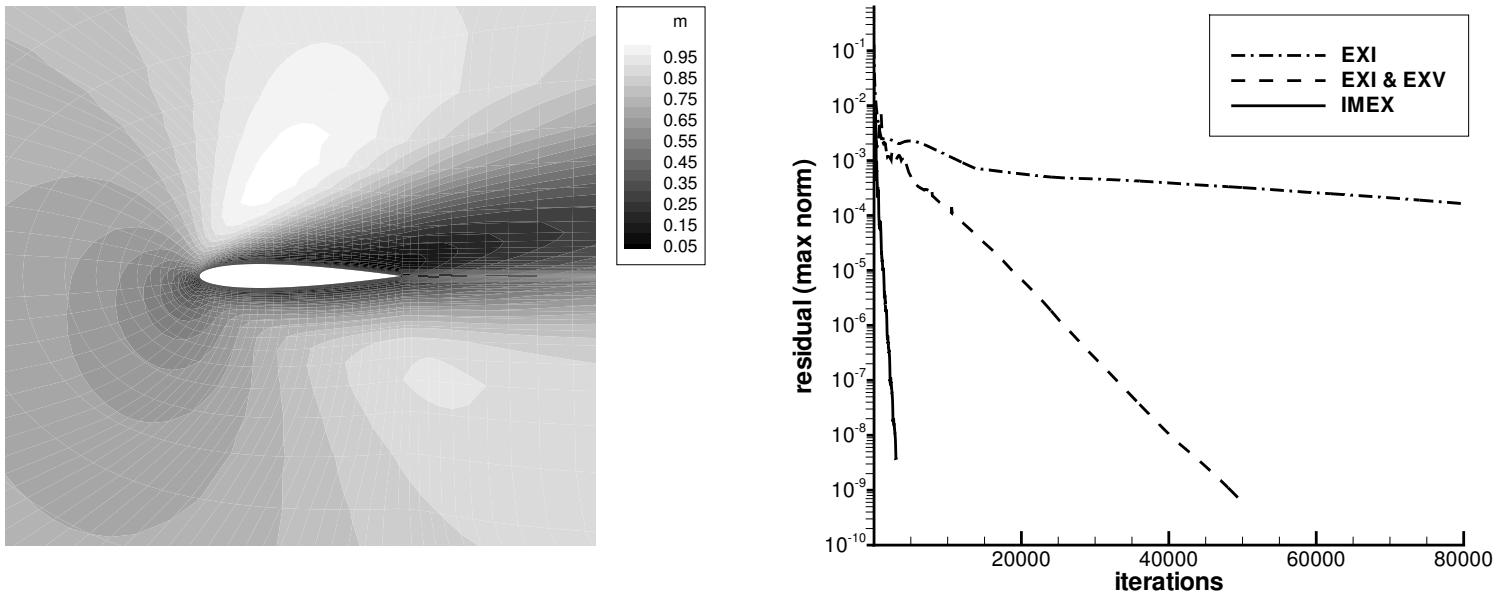


Figure: Local Mach numbers for A1 test case ($\mathrm{M}_\infty = 0.8$, $\mathrm{Re}_\infty = 73$, $\alpha = 12°$) on a 112×38 grid and convergence to steady-state for the different pseudo-time stepping methods.

Pseudo-Time Integration

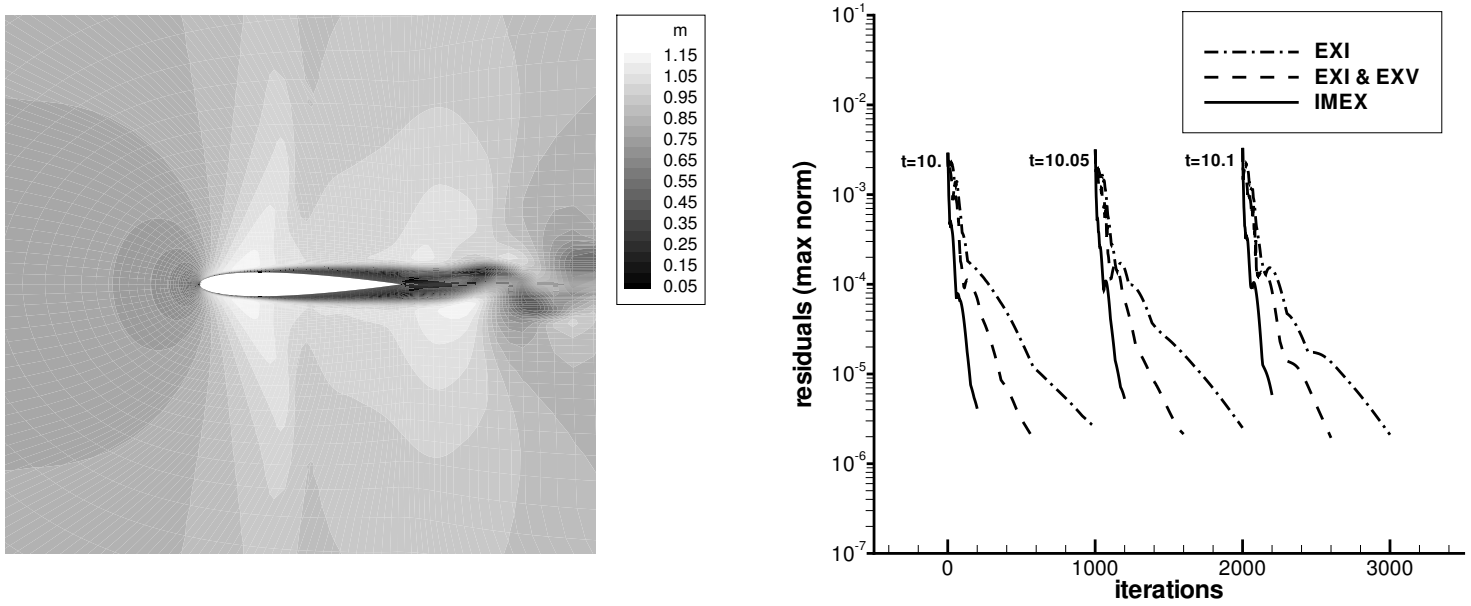


Figure: Local Mach numbers for A7 test case ($\mathrm{M}_\infty = 0.85$, $\mathrm{Re}_\infty = 10^4$, $\alpha = 0°$) on a 224×76 grid at $t = 10$ and convergence in pseudo-time for three physical time steps with the different pseudo-time stepping methods.

Flux Quadrature

- In the discontinuous Galerkin discretization the integration of the (approximate) Riemann solver flux is by far the computationally most expensive part.
- The standard Gauss (product) quadrature rules require too many flux evaluations to be practical. In particular for discretizations in three-dimensions or in space-time (4D).
- For linear basis functions an alternative is provided by the Taylor quadrature rule which also uses the gradient information available in a DG discretization.

Taylor Flux Quadrature Rule

Main Features:

- In order to improve the computational efficiency the flux function in the integrand is replaced with a second or higher order Taylor series expansion evaluated at the face or element center.
- This approximation improves the computational efficiency significantly:
  - since only one flux evaluation per element face is necessary, instead of 2^d or 3^d ($d = \dim(\Omega)$) for a product Gauss quadrature rule;
  - the locality of the required flow data improves the cache and vector performance.
- The approximate flux integration does not result in a loss of accuracy in the DG discretization when a sufficient number of terms in the Taylor series expansion are used.

Taylor Quadrature Rule for Space-Time DG

- The integrals of the flux tensor $\mathcal{F}_{ik}(U)$ over a space-time face $\mathcal{S}_m \subset \partial\mathcal{K}$ can be approximated as

$$\int_{\mathcal{S}_m} \phi_m \mathcal{F}_{ik}(U) n_k dx \cong \mathcal{F}_{ik}(U(\bar{\xi}_m)) \int_{\hat{\mathcal{S}}} \xi_m d\hat{\mathcal{S}}_k^m$$

$$+ \sum_{l \in I(\mathcal{S}_m)} \frac{\partial \mathcal{F}_{ik}}{\partial U^j}(U(\bar{\xi}_m)) \frac{\partial U^j}{\partial \xi_l}(\bar{\xi}_m) \int_{\hat{\mathcal{S}}} \xi_l \xi_m d\hat{\mathcal{S}}_k^m$$

$$+ \text{ higher-order terms,}$$

with $\bar{\xi}_m$ the computational face center of face $\mathcal{S}_m$, defined by $\xi_{m,i} = \pm\delta_{im}$.

Benefits of Taylor Quadrature Rule

The flow derivatives necessary for the Taylor approximation can be easily computed:

- In computational coordinates the solution vector U_h in cell $\mathcal{K}$, restricted to the face $\mathcal{S}_{m_1}$, can be written as

$$U_{|\mathcal{S}_{m_1}} = \bar{U}(\bar{\xi}_{m_1}) + \xi_{m_2}\hat{U}_{m_2}(\mathcal{K}) + \xi_{m_3}\hat{U}_{m_3}(\mathcal{K}) + \xi_4\hat{U}_4(\mathcal{K})$$

$$+ \text{ higher-order terms,}$$

  hence, the flow derivatives $D^\alpha U_{|\mathcal{S}_{m_1}}$ can be computed directly using the series representation of U.

- The integrals $\int_{\mathcal{S}_m} \hat{\phi}(\xi)\xi_{i_1}^{k_1}\xi_{i_2}^{k_2}\cdots\xi_{i_d}^{k_d}\, d\hat{\mathcal{S}}_k^m$ can be easily computed analytically.

- For multi-dimensional integrals the number of flux evaluations in the Taylor quadrature rule is nearly independent of the dimension d of the integration domain. A product Gauss quadrature rule would require 2^d or 3^d flux evaluations.

Taylor Quadrature Rule for Space-Time DG

Remarks

- The gradient contribution is necessary to obtain second order accuracy for the element face flux integrals.

- For the stability of a DG discretization with linear basis functions it is also essential to incorporate the gradient contributions in the approximation of the integral.

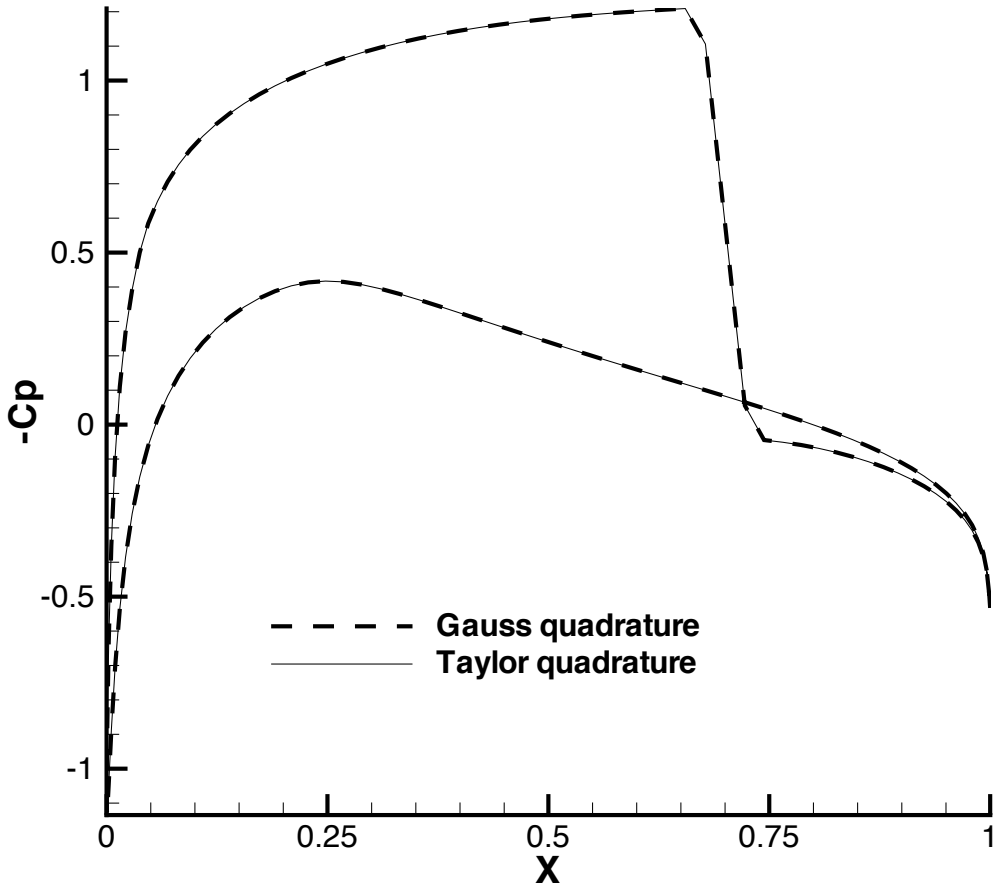
- For an upwind flux

  $$\hat{\mathcal{F}}(U_L, U_R) := \frac{1}{2}(\mathcal{F}(U_L) + \mathcal{F}(U_R))n - D(U_L, U_R),$$

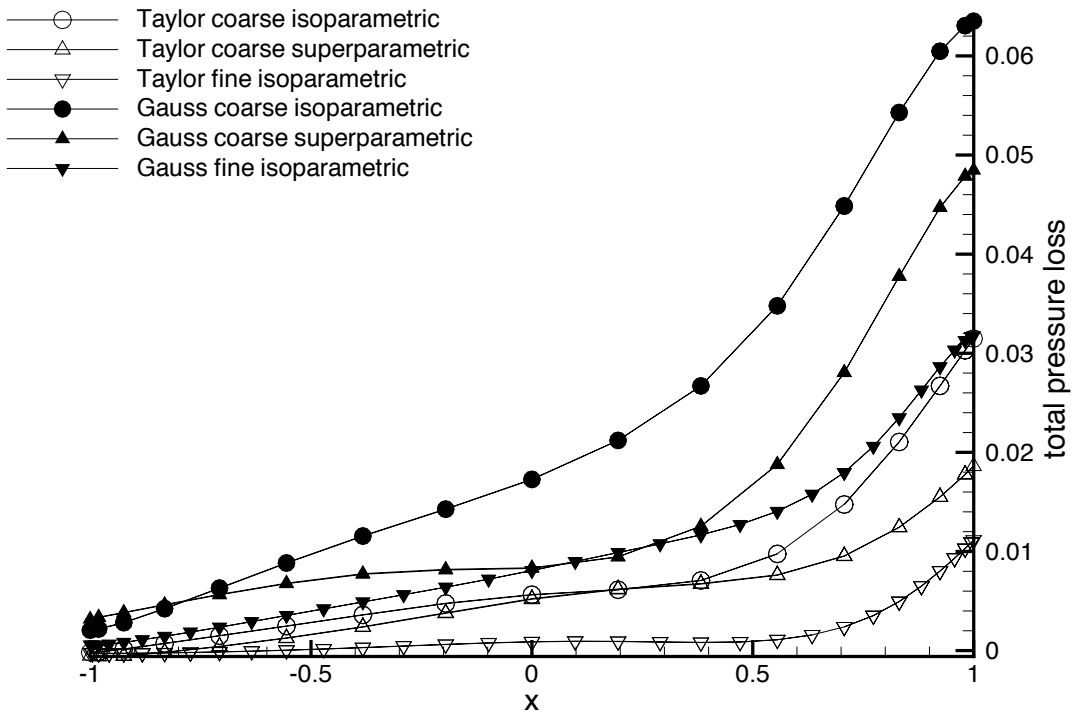
  it is essential to not just expand the central part of the flux, but also the dissipative part $D(U_L, U_R)$.

- It is important to incorporate the dissipative part of the upwind flux in the Taylor approximation which has been one of the reasons to apply the HLLC-flux, but a Lax-Friedrichs would also be a good alternative.

Comparison of Taylor and Gauss Quadrature Rules



Pressure distribution for transonic flow over a NACA0012 airfoil computed with Taylor and Gauss quadrature rules for the element and face fluxes ($M_\infty = 0.8$, $\alpha = 2^\circ$).



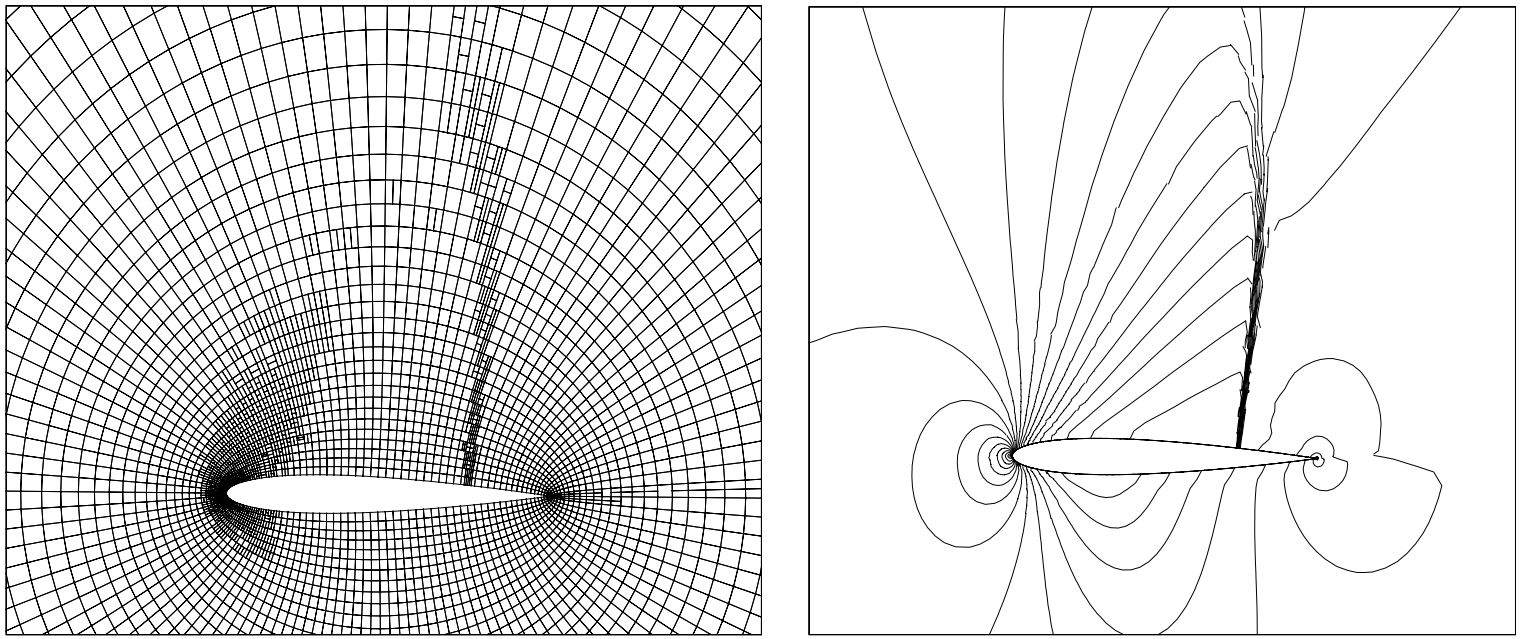
Total pressure loss at the wall for the flow around a circular cylinder ($M_\infty = 0.38$) using Gauss and Taylor flux quadrature rules on coarse (32×48 elements) and fine mesh (64×96).

Oscillating NACA 0012 Airfoil in Transonic Inviscid Flow

Space-time adaptive simulations of oscillating NACA 0012 airfoil:

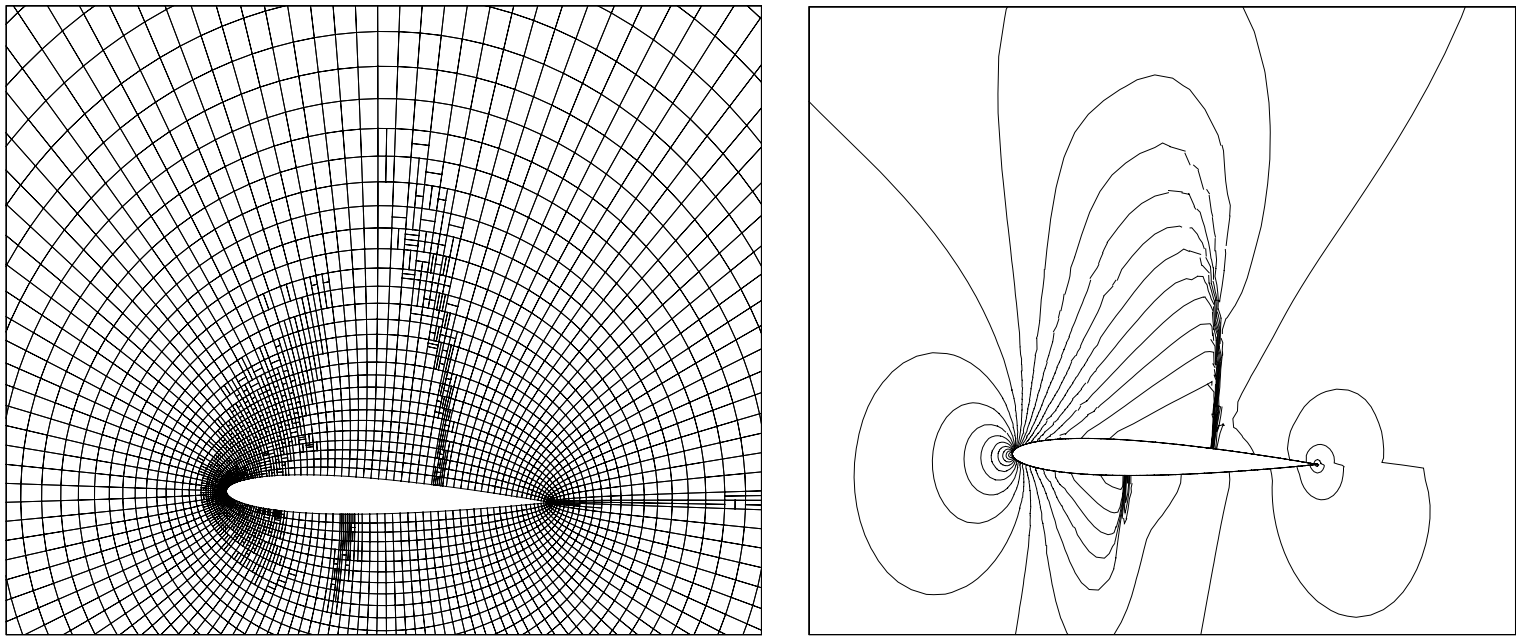
- Free stream Mach number $M_\infty = 0.8$.
- Oscillation period $T = 20$ (normalized with L/a_∞).
- Oscillating frequency $\omega = \pi/10$.
- Oscillation amplitude range between -0.5 degrees and 4.5 degrees.
- Fine mesh size 32,768 elements.
- Adapted mesh approximately 9400 elements during the simulation.
- Adaptation each time step.
- Inviscid flow.

Oscillating NACA 0012 Airfoil



Adapted mesh and density contours around oscillating NACA 0012 airfoil $\alpha = 0.5^\circ$ (pitching downward).

Oscillating NACA 0012 Airfoil

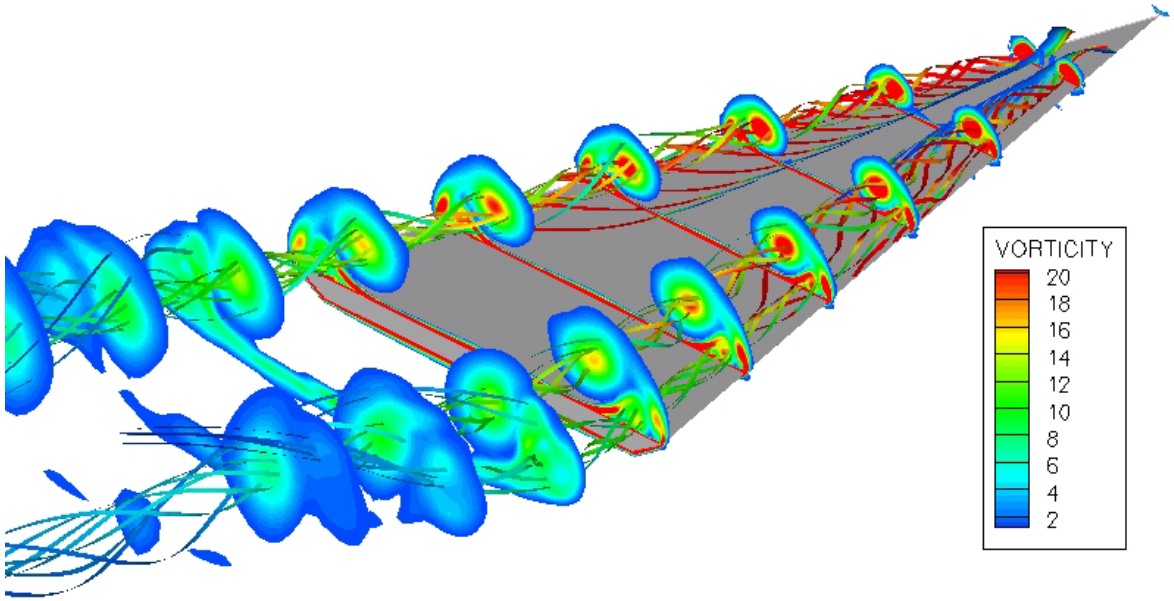


Adapted mesh and density contours around oscillating NACA 0012 airfoil $\alpha = 2.0°$ (pitching upward).

Delta Wing Simulations

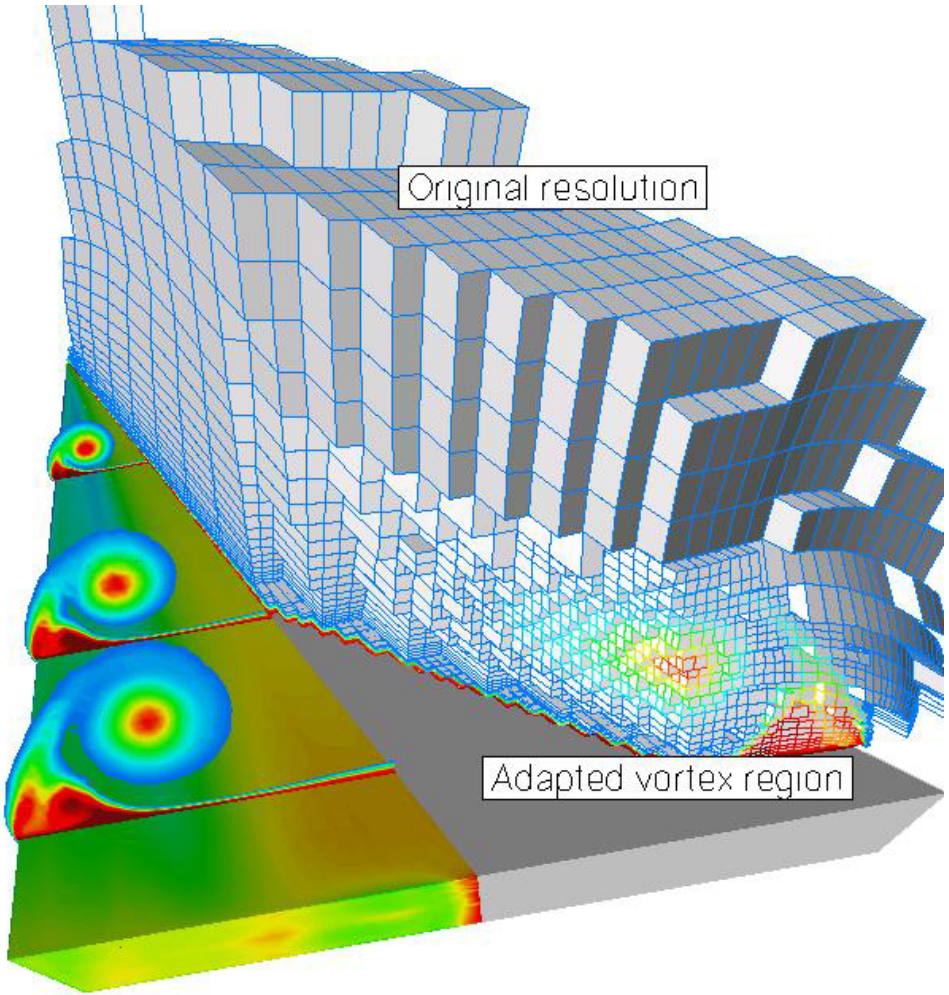
- Simulations of viscous flow about a delta wing with 85° sweep angle.
- Conditions
  - Mach number $M = 0.3$
  - Reynolds number $Re = 40.000$
  - Angle of attack $\alpha = 12.5°$.
  - Fine grid mesh 1.600.000 elements, 40.000.000 degrees of freedom
  - Adapted mesh, initial mesh 208.896 elements, after four adaptations 286.416 elements

Delta Wing Simulations



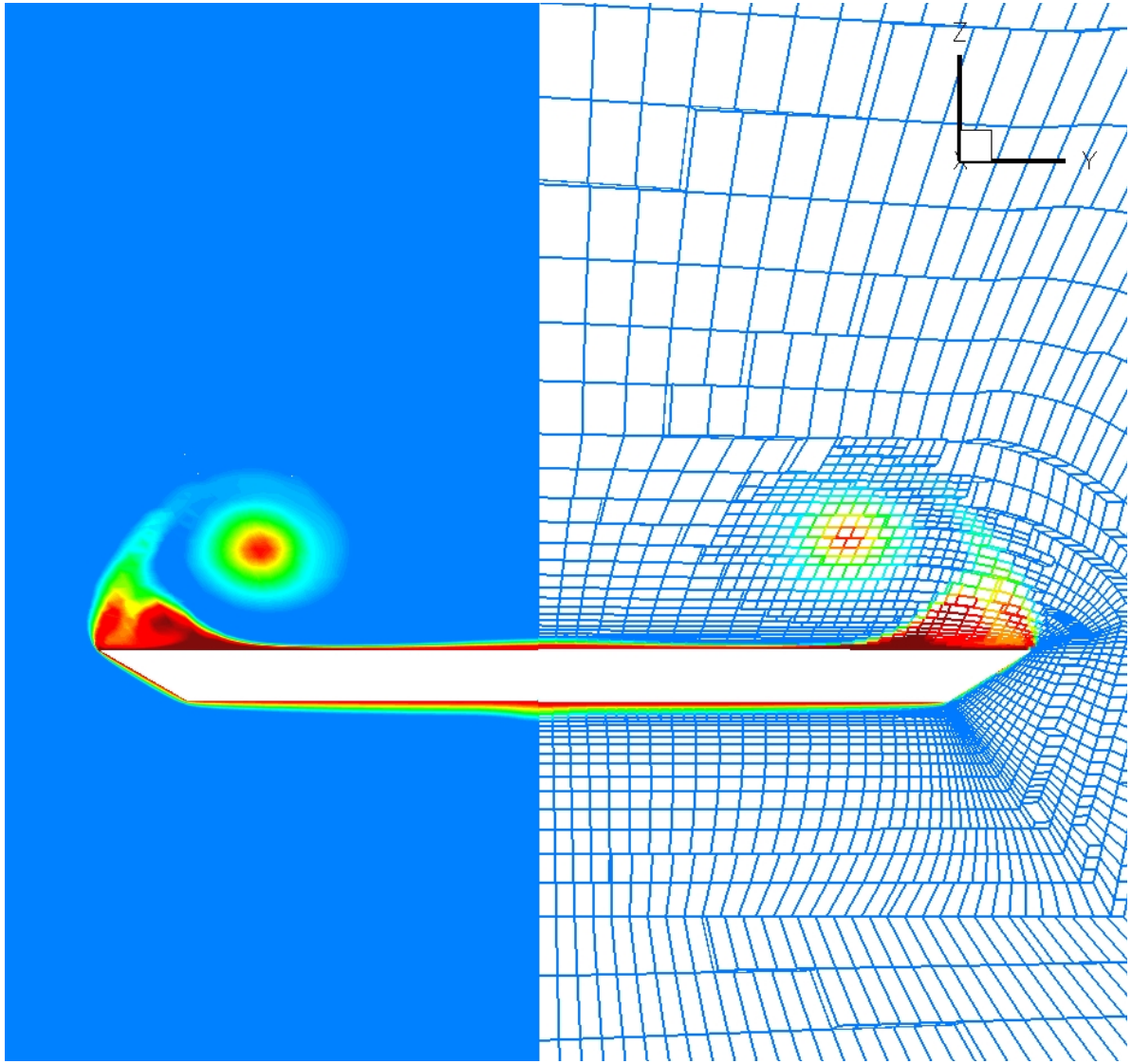
Streaklines and vorticity contours in various cross-sections

Delta Wing Simulations



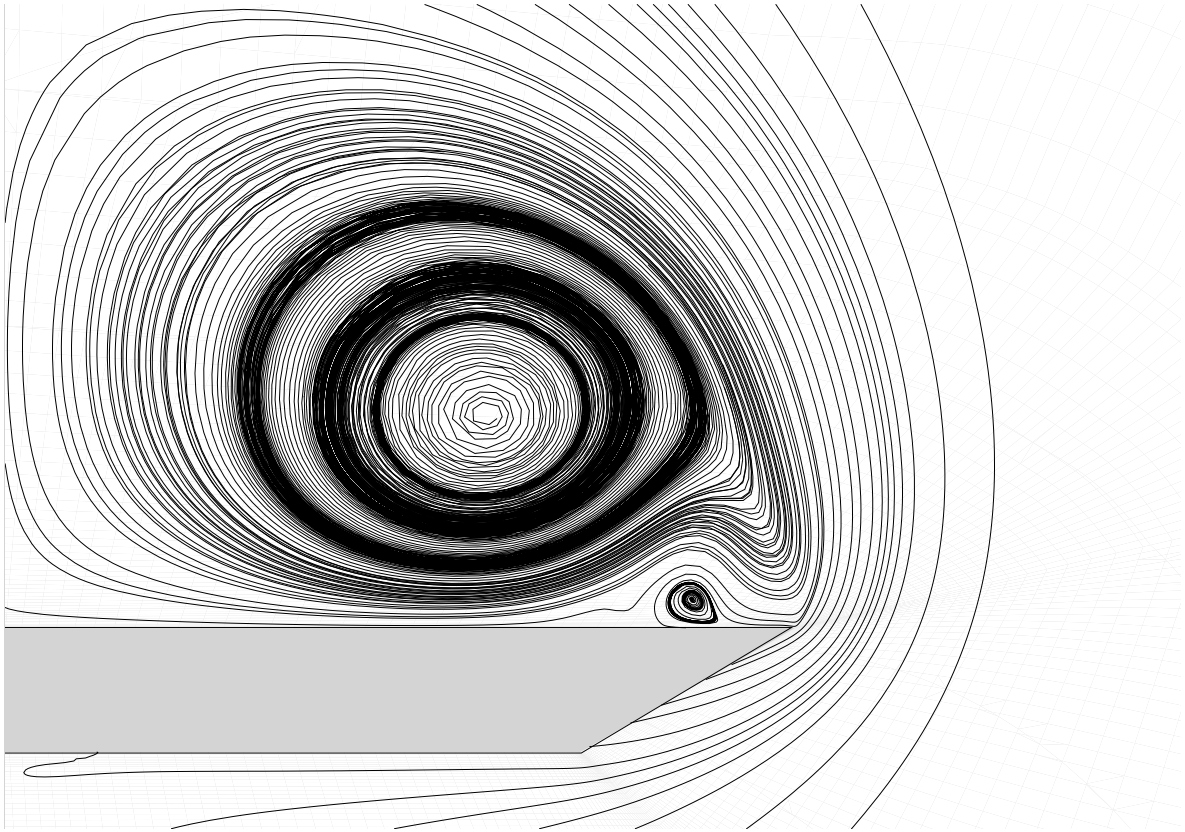
Pressure contours on delta wing surface and vorticity contours in cross-sections.
Left, unadapted mesh, (220.000 elements), right, adapted mesh (350.000 elements)

Delta Wing Simulations



Cross-section of delta wing simulation at x/c=0.8
(left, total pressure loss, right locally adapted mesh)

Delta Wing Simulations



Streamlines around the delta wing (cross section $x/c = 0.6$)

NACA0012 Airfoil in Laminar Dynamic Stall

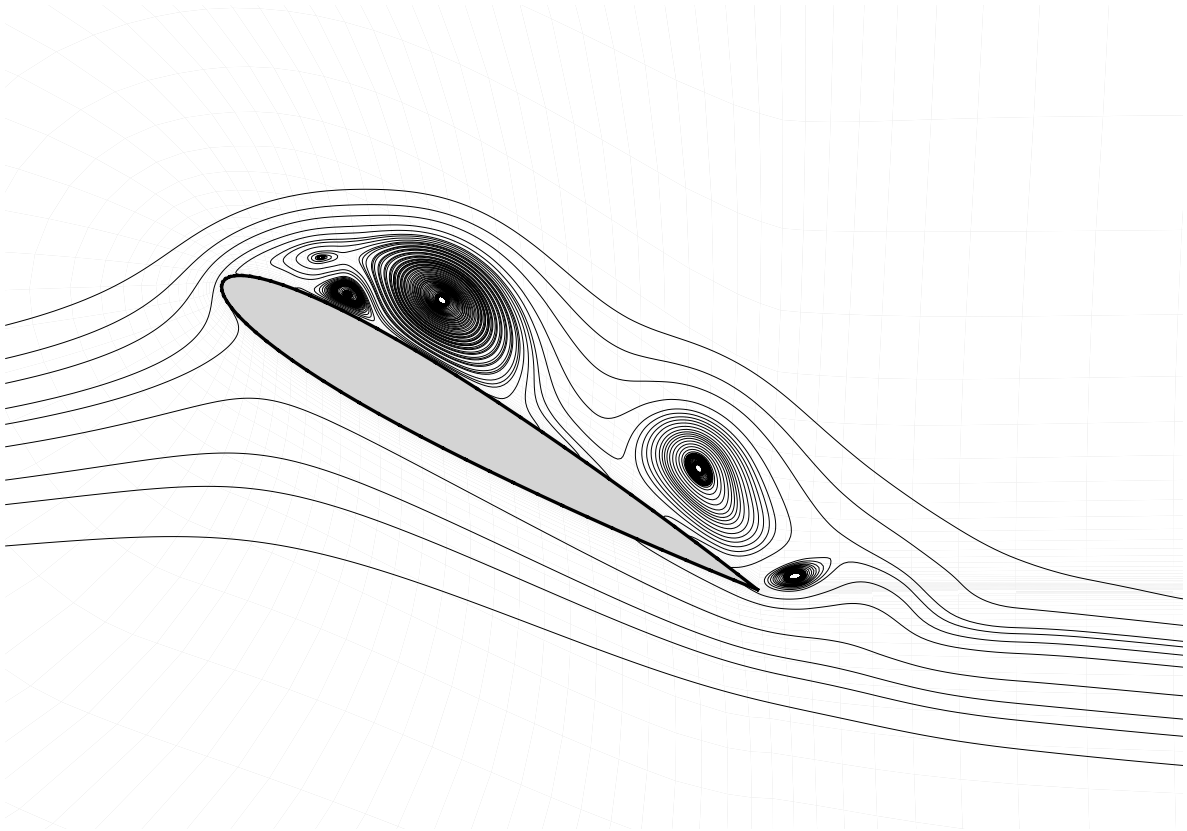
Conditions:

- 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:

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

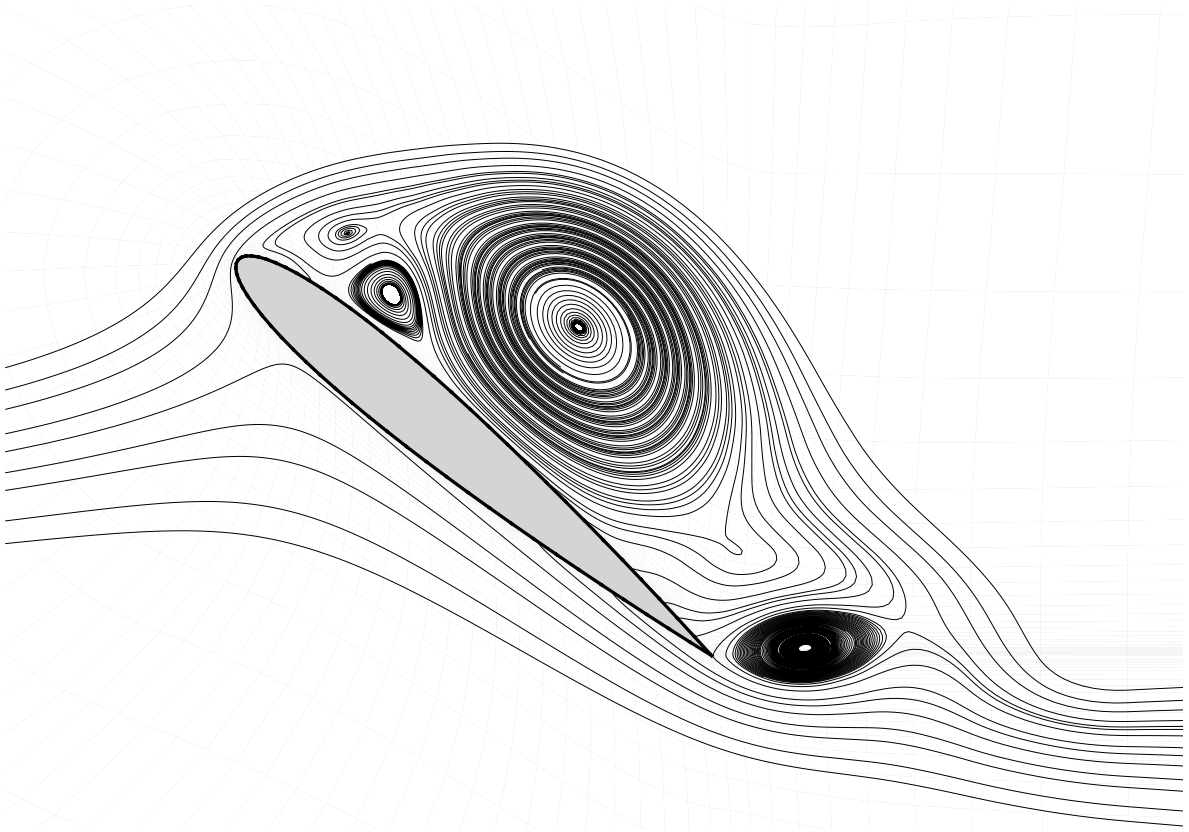
  with coefficients $a = -1.2455604$, $b = 2.2918312$, $c = 1.84$ and time $t \in [0, 25]$.
- Time step $\Delta t = 0.005$
- C-type mesh with 112×38 elements with 14 elements in the boundary layer

NACA0012 Airfoil in Laminar Dynamic Stall



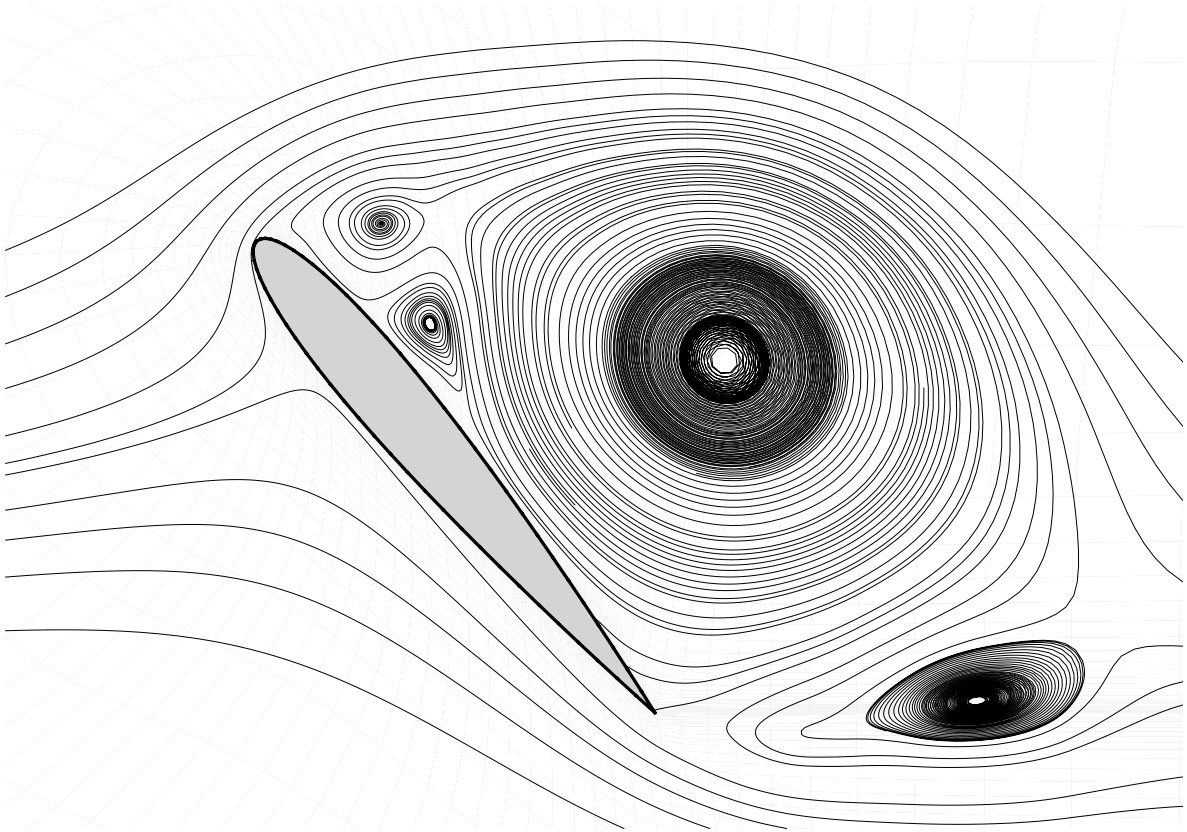
Streamlines around NACA 0012 airfoil in dynamic stall at $\alpha = 30°$.

NACA0012 Airfoil in Laminar Dynamic Stall



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

NACA0012 Airfoil in Laminar Dynamic Stall



Streamlines around NACA 0012 airfoil in dynamic stall at $\alpha = 50°$.

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

- The Taylor quadrature significantly improves the efficiency of the flux integrals without a reduction in accuracy.
- The use of a stabilization operator instead of a slope limiter makes it possible to converge to machine accuracy for steady state problems, whereas a limiter prevents convergence to steady state and reduces accuracy in a large part of the domain.

References

1. C.M. Klaij, J.J.W. van der Vegt and H. van der Ven, Space-time discontinuous Galerkin method for the compressible Navier-Stokes equations, *J. Comput. Phys.* **217**, pp. 589-611 (2006).

2. J.J.W. van der Vegt and H. van der Ven, Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows. Part I. General formulation., *J. Comput. Phys.* **182**, pp. 546-585 (2002).

3. H. van der Ven and J.J.W. van der Vegt, Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows. II. Efficient flux quadrature, *Comput. Meth. Appl. Mech. Engrg.* **191**, pp. 4747-4780 (2002).

References

4. J.J. Sudirham, J.J.W. van der Vegt and R.M.J. van Damme, Space-time discontinuous Galerkin method for advection-diffusion problems on time-dependent domains, *Applied Numerical Mathematics*, **56**, pp. 1491-1518 (2006).

5. J.J.W. van der Vegt and H. van der Ven, Discontinuous Galerkin finite element method with anisotropic local grid refinement for inviscid compressible flows, *J. Comput. Phys.* **141**, 46-77 (1998).