Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 217 (2006) 589–611
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Space–time discontinuous Galerkin method for the compressible Navier–Stokes equations

C.M. Klaij [a,*], J.J.W. van der Vegt [a], H. van der Ven [b]

[a] University of Twente, Department of Applied Mathematics, P.O. Box 217, 7500 AE Enschede, The Netherlands
[b] National Aerospace Laboratory NLR, P.O. Box 90502, 1006 BM Amsterdam, The Netherlands

Received 22 July 2005; received in revised form 22 December 2005; accepted 9 January 2006
Available online 23 February 2006

Abstract

A space–time discontinuous Galerkin finite element method for the compressible Navier–Stokes equations is presented. We explain the space–time setting, derive the weak formulation and discuss our choices for the numerical fluxes. The resulting numerical method allows local grid adaptation as well as moving and deforming boundaries, which we illustrate by computing the flow around a 3D delta wing on an adapted mesh and by simulating the dynamic stall phenomenon of a 2D airfoil in rapid pitch-up maneuver.

PACS: 02.60.Cb; 02.70.Dh; 03.40.Gc

Keywords: Compressible Navier–Stokes equations; Discontinuous Galerkin finite element methods; Numerical fluxes; Arbitrary Lagrangian Eularian (ALE) formulation

1. Introduction

Many applications in fluid dynamics require the solution of the compressible Navier–Stokes equations on a domain with time dependent boundaries. Examples are aero-elastic problems such as helicopter rotors in forward flight, flaps and slats on wings and piston engines. The accurate solution of these problems frequently requires time-dependent moving and deforming meshes and it is non-trivial to maintain a conservative and accurate scheme on this type of meshes [16,21]. Finite volume ALE methods, for example, do not automatically satisfy the geometric conservation law on deformed and adapted meshes, which was proven to be essential [21] for the accuracy of the solution. These issues have been the main motivation in [26,29] to develop a space–time discontinuous Galerkin (DG) finite element method for inviscid compressible flows. This algorithm combines a fully conservative arbitrary Lagrangian Eulerian (ALE) approach to deal with deforming

* Corresponding author.
E-mail addresses: c.m.klaij@math.utwente.nl (C.M. Klaij), j.j.w.vandervegt@math.utwente.nl (J.J.W. van der Vegt), venvd@nlr.nl (H. van der Ven).

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.01.018

meshes with the well known benefits of the compact stencil of a DG method, such as optimal flexibility for local mesh refinement, adjustment of the polynomial order in each element (*hp*-adaptation) and excellent performance on parallel computers. The method has been demonstrated for a variety of aerodynamic applications, including rotorcraft [8,28] and deforming wings [30].

The space–time DG method discussed in [26,29] has been limited so far to inviscid compressible flows. In this article we aim at extending the space–time DG formulation to the compressible Navier–Stokes equations, which significantly extends its range of applications. The key feature of the space–time DG method discussed in this article is that no distinction is made between space and time variables and the discretization is directly performed in four dimensional space. This provides optimal flexibility to deal with time dependent boundaries and deforming elements and naturally results in a conservative discretization, even on deforming, locally refined meshes with hanging nodes. The space–time algorithm results in an implicit time-integration method which is unconditionally stable and preserves accuracy also on non-smooth meshes. A complete *hp*-error and stability analysis of the space–time DG discretization for the linear advection–diffusion equation is given in [24].

Discontinuous Galerkin methods have recently received significant attention and are applied to a wide range of hyperbolic and (incompletely) parabolic problems. For a survey, see [1,10–12,14]. An important step towards a DG discretization for the compressible Navier–Stokes equations was made by the pioneering work of Bassi and Rebay [3] and in a different formulation by Baumann and Oden [7]. These algorithms provide discretization techniques for the diffusion operator and extend the DG formulation for hyperbolic equations developed by Cockburn and Shu (see [14] for a detailed survey) to incompletely parabolic equations. Improvements to the original formulation [3], which showed a weak instability, have been provided in [5] and analyzed in [1,9]. Applications to the solution of the compressible Reynolds averaged Navier–Stokes equations are discussed in [2,4,15,17]. A slightly different approach to deal with the diffusion operator has been proposed by Cockburn and Shu [13] with the local discontinuous Galerkin method. Although the various DG formulations for the diffusion operator are quite different, there are no major differences in terms of accuracy, computational cost and complexity between the methods which proved to be consistent, adjoint consistent and of optimal order in the analysis given in [1]. In this article we follow the approach of Brezzi [9] for the diffusion operator and include this technique in the space–time discretization for compressible flows which we presented in [26,29] and to which we refer for details on the inviscid part of the algorithm.

The outline of this article is as follows. In Section 2, we first summarize the equations of gas dynamics. Next, in Section 3, we discuss the space–time discontinuous Galerkin discretization of the compressible Navier–Stokes equations. We start with the definition of the geometry of the space–time domain and discuss the necessary functional spaces and operators. This setting is used to define the weak formulation and a crucial part is the discussion of the space–time numerical fluxes. The proper definition of these fluxes allows the transformation of the space–time formulation into an arbitrary Lagrangian Eulerian formulation which combines well with upwind schemes based on approximate Riemann solvers. In Section 4, we derive the non-linear algebraic equations for the expansion coefficients of the solution in each element. In Section 5, we demonstrate the method with several test cases and concluding remarks are drawn in Section 6.

2. The compressible Navier–Stokes equations

The equations of motion considered in this article are the Navier–Stokes equations describing viscous compressible flows, which form a system of five coupled equations expressing conservation of mass, momentum and energy. Using the summation convention on repeated indices and the comma notation to denote partial differentiation the compressible Navier–Stokes equations can be written as

$$U_{,t} + F^e_k(U)_{,k} - F^v_k(U, \nabla U)_{,k} = 0 \tag{1}$$

with the vector of conservative variables $U \in \mathbb{R}^5$, the inviscid flux $F^e \in \mathbb{R}^{5\times 3}$ and the viscous flux $F^v \in \mathbb{R}^{5\times 3}$ given by

$$U = \begin{bmatrix} \rho \\ \rho u_j \\ \rho E \end{bmatrix}, \quad F_k^e = \begin{bmatrix} \rho u_k \\ \rho u_j u_k + p\delta_{jk} \\ u_k(\rho E + p) \end{bmatrix}, \quad F_k^v = \begin{bmatrix} 0 \\ \tau_{jk} \\ \tau_{kj} u_j - q_k \end{bmatrix}, \tag{2}$$

with $j, k = 1, 2, 3$. The conservative variables are the density ρ, the momentum density vector $\rho\vec{u}$ and the total energy density ρE, with $\vec{u}$ the velocity vector and E the total energy. The pressure is denoted by p and the symbol δ represents the Kronecker delta function. The total stress tensor τ is defined as

$$\tau_{jk} = \lambda u_{i,i}\delta_{jk} + \mu(u_{j,k} + u_{k,j})$$

with $i = 1, 2, 3$ and 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: $3\lambda + 2\mu = 0$. The heat flux vector $\vec{q}$ is defined as

$$q_k = -\kappa T_{,k}$$

with κ the thermal conductivity coefficient. For a calorically perfect gas, the pressure p and internal energy e are given by the following equations of state:

$$p = \rho R T, \quad e = c_v T,$$

where $R = c_p - c_v$ is the specific gas constant and c_p and c_v the specific heats at constant pressure and constant volume, respectively. Since the total energy is the sum of the internal and kinetic energy

$$E = e + \frac{1}{2}u_i u_i$$

the pressure and temperature can be expressed in terms of the conservative variables as

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

where $\gamma = c_p/c_v$ is the ratio of specific heats. We are mainly interested in the flow around aircraft and therefore use uniform flow as initial condition and far-field boundary condition

$$\rho = \rho_\infty, \quad \vec{u} = \vec{u}_\infty, \quad p = p_\infty.$$

At the solid surface we apply the isothermal no-slip boundary condition

$$\vec{u} = 0, \quad T = T_\infty.$$

We conclude this section by noticing that the viscous flux F^v is homogeneous with respect to the gradient of the conservative variables ∇U

$$F_{ik}^v(U, \nabla U) = A_{ikrs}(U)U_{r,s}$$

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

$$A_{ikrs}(U) := \frac{\partial F_{ik}^v(U, \nabla U)}{\partial(U_{r,s})}$$

with $i, r = 1, \ldots, 5$ and $k, s = 1, 2, 3$, see Appendix A. This property plays a crucial role in the derivation of the weak formulation of the compressible Navier–Stokes equations.

3. Space–time discontinuous Galerkin discretization

This section covers the space–time discontinuous Galerkin discretization of the compressible Navier–Stokes equations. We first define the geometry of the space–time domain, then the necessary functional spaces and operators and finish with the derivation of the primal formulation, while discussing our choices for the numerical fluxes.

3.1. Geometry of the space–time domain

The space–time discontinuous Galerkin finite element method does not distinguish between space and time variables; instead the equations are considered in an open domain $\mathscr{E} \subset \mathbb{R}^4$, where a point with position $\bar{x} = (x_1, x_2, x_3)$ at time $t = x_0$ has Cartesian coordinates $(x_0, \bar{x})$. At time t, the flow domain $\Omega(t)$ is defined as $\Omega(t) := \{\bar{x} \in \mathbb{R}^3 : (t, \bar{x}) \in \mathscr{E}\}$. Let t_0 and T be the initial and final time of the evolution of the flow domain, then the space–time domain boundary $\partial\mathscr{E}$ consists of the hypersurfaces $\Omega(t_0) := \{x \in \mathscr{E} : x_0 = t_0\}$, $\Omega(T) := \{x \in \mathscr{E} : x_0 = T\}$, and $\mathscr{Q} := \{x \in \partial\mathscr{E} : t_0 < x_0 < T\}$.

First, consider the partitioning of the time interval $[t_0, T]$ by an ordered series of time levels $t_0 < t_1 < \cdots < T$. The space–time domain $\mathscr{E}$ is divided into N_t space–time slabs $\mathscr{E}^n = \mathscr{E} \cap I_n$, with $I_n = (t_n, t_{n+1})$ the nth time interval. Each space–time slab $\mathscr{E}^n$ is bounded by $\Omega(t_n)$, $\Omega(t_{n+1})$ and $\mathscr{Q}^n = \partial\mathscr{E}^n / (\Omega(t_n) \cup \Omega(t_{n+1}))$.

Second, consider an approximation $\Omega_h(t_n)$ of $\Omega(t_n)$ and divide $\Omega_h(t_n)$ into N_n non-overlapping hexahedral spatial elements $K_j(t_n)$, where $\Omega_h(t) \to \Omega(t)$ as $h \to 0$, with h the radius of the smallest sphere completely containing each element $K_j(t_n)$. Similarly, $\Omega_h(t_{n+1})$ approximates $\Omega(t_{n+1})$ (see Fig. 1). Each element K^n is related to the master element $\widehat{K} = (-1, 1)^3$ through the mapping F_K^n

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

with x_i the spatial coordinates of the vertices of the hexahedron K^n and χ_i the usual tri-linear finite element shape functions for hexahedra (see Fig. 2). The space–time elements $\mathscr{K}_j^n$ of $\mathscr{E}^n$ are constructed by connecting $K_j(t_n)$ with $K_j(t_{n+1})$ using linear interpolation in time, which results in the mapping $G_{\mathscr{K}}$ from the master element $\widehat{\mathscr{K}} = (-1, 1)^4$ to the space–time element $\mathscr{K}^n$

$$G_{\mathscr{K}}^n : \widehat{\mathscr{K}} \to \mathscr{K}^n : \xi \mapsto (t, \bar{x}) = \left(\frac{1}{2}(t_{n+1} + t_n) + \frac{1}{2}(t_{n+1} - t_n)\xi_0, \frac{1}{2}(1 - \xi_0)F_K^n(\bar{\xi}) + \frac{1}{2}(1 + \xi_0)F_K^{n+1}(\bar{\xi})\right).$$

The tessellation $\mathscr{T}_h^n$ of the space–time slab $\mathscr{E}_h^n$ consists of all space–time elements $\mathscr{K}_j^n$, thus the tessellation $\mathscr{T}_h$ of the discrete flow domain $\mathscr{E}_h$ is simply $\mathscr{T}_h = \cup_{n=0}^{N_t-1} \mathscr{T}_h^n$.

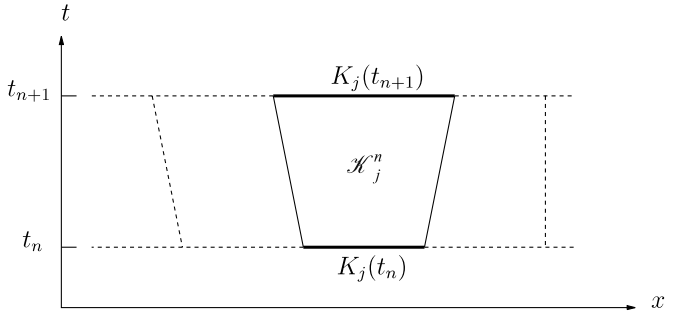

Fig. 1. The spatial element $K_j(t_n)$ moves and deforms into $K_j(t_{n+1})$. The space–time element $\mathscr{K}_j^n$ is constructed by linear interpolation in time.

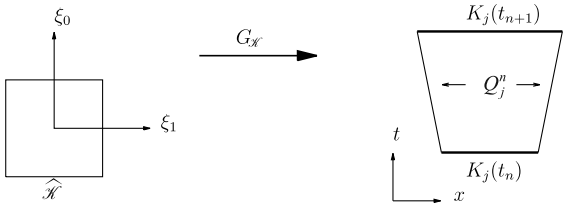

Fig. 2. Mapping $G_{\mathscr{K}}$ between the master element $\widehat{\mathscr{K}}$ and the space–time element $\mathscr{K}_j^n$ with boundaries $\partial\mathscr{K}_j^n = K_j(t_n^+) \cup K_j(t_{n+1}^-) \cup Q_j^n$.

Finally, consider the element boundary $\partial\mathscr{K}$ which is the union of open faces of $\mathscr{K}_j^n$ and consists of three parts: $K_j(t_n^+) = \lim_{\epsilon\downarrow 0} K_j(t_n+\epsilon)$, $K_j(t_{n+1}^-) = \lim_{\epsilon\downarrow 0} K_j(t_{n+1}-\epsilon)$ and $Q_j^n = \partial\mathscr{K}_j^n/(K_j(t_n^+)\cup K_j(t_{n+1}^-))$. The space–time normal vector at an element boundary point moving with velocity $\vec{v}$ is given by

$$n = \begin{cases} (1,0,0,0) & \text{at } K(t_{n+1}^-), \\ (-1,0,0,0) & \text{at } K(t_n^+), \\ (-v_k\bar{n}_k, \bar{n}) & \text{at } Q^n. \end{cases} \tag{3}$$

It is often convenient to consider the faces separately instead of the whole element boundary. Therefore, in addition to the previously defined faces $K_j(t_n^+)$ and $K_j(t_{n+1}^-)$, we also define interior and boundary faces as follows. A face $\mathscr{S}$ is an interior face if it is shared by two neighboring elements $\mathscr{K}_i^n$ and $\mathscr{K}_j^n$, such that $\mathscr{S} = Q_i^n \cap Q_j^n$, and a boundary face if $\mathscr{S} = \partial\mathscr{E}^n \cap Q_j^n$. The set of all interior faces in time slab I_n is denoted by $\mathscr{S}_I^n$, the set of all boundary faces by $\mathscr{S}_B^n$, and the total set of faces by $\mathscr{S}_{I,B}^n = \mathscr{S}_I^n \cup \mathscr{S}_B^n$.

3.2. Functional spaces and operators

Each element $\mathscr{K}$ of the tessellation $\mathscr{T}_h$ is an image of the master element $\widehat{\mathscr{K}}$: $\mathscr{K} = G_{\mathscr{K}}(\widehat{\mathscr{K}})$, where $\widehat{\mathscr{K}} = (-1,1)^4$ is the open unit hypercube in $\mathbb{R}^4$. The finite element space associated with the tessellation $\mathscr{T}_h$ is given by

$$W_h = \{W \in (L^2(\mathscr{E}_h))^5 : W|_{\mathscr{K}} \circ G_{\mathscr{K}} \in (P^k(\widehat{\mathscr{K}}))^5 \ \forall \mathscr{K} \in \mathscr{T}_h\},$$

where $L^2(\mathscr{E}_h)$ is the space of square integrable functions on $\mathscr{E}_h$ and $P^k(\widehat{\mathscr{K}})$ denotes the space of polynomials of degree at most k in element $\widehat{\mathscr{K}}$. We will also use the following space:

$$V_h = \{V \in (L^2(\mathscr{E}_h))^{5\times 3} : V|_{\mathscr{K}} \circ G_{\mathscr{K}} \in (P^k(\widehat{\mathscr{K}}))^{5\times 3} \ \forall \mathscr{K} \in \mathscr{T}_h\}.$$

Note that $\nabla_h W_h \subset V_h$, where the broken gradient ∇_h of W_h is defined as $(\nabla_h W_h)|_{\mathscr{K}} = \nabla(W_h|_{\mathscr{K}})$. This relation between the functional spaces is essential for the discretization.

The trace of a function $f \in W_h$ at the element boundary $\partial\mathscr{K}^{\mathrm{L}}$ is defined as

$$f^{\mathrm{L}} = \lim_{\epsilon\downarrow 0} f(x - \epsilon n^{\mathrm{L}}),$$

with n^{L} the unit outward space–time normal at $\partial\mathscr{K}^{\mathrm{L}}$. We will also use the notation $\bar{n}^{\mathrm{L}}$ when only the space components of the outward normal vector are considered. Because of the discontinuous function approximation, a function f in W_h and V_h can have a double-valued trace at the element boundaries $\partial\mathscr{K}$. The traces of the function f at an internal face $\mathscr{S} = \overline{\mathscr{K}}^{\mathrm{L}} \cap \overline{\mathscr{K}}^{\mathrm{R}}$ are denoted by f^{L} and f^{R}, respectively. The jump of f at an internal face $\mathscr{S} \in \mathscr{S}_I^n$ in the space direction k of a Cartesian coordinate system is defined as

$$[\![f]\!]_k = f^{\mathrm{L}}\bar{n}_k^{\mathrm{L}} + f^{\mathrm{R}}\bar{n}_k^{\mathrm{R}}. \tag{4}$$

Furthermore, we define the average of f at $\mathscr{S} \in \mathscr{S}_I^n$ as

$$\{\!\{f\}\!\} = \frac{1}{2}(f^{\mathrm{L}} + f^{\mathrm{R}}). \tag{5}$$

The jumps and averages are not needed at faces other than internal faces. Note that the jump operator satisfies the following product rule at $\mathscr{S} \in \mathscr{S}_I^n$ for $f \in V_h$ and $g \in W_h$

$$[\![g_i f_{ik}]\!]_k = \{\!\{g_i\}\!\}[\![f_{ik}]\!]_k + [\![g_i]\!]_k\{\!\{f_{ik}\}\!\}, \tag{6}$$

which can be verified by straightforward substitution of (4) and (5) into (6). We will also use the following relation for the element boundary integrals which occur in the weak formulation:

$$\sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\partial} g_i^{\mathrm{L}} f_{ik}^{\mathrm{L}} \bar{n}_k^{\mathrm{L}} \,\mathrm{d}\partial = \sum_{\mathscr{S}\in\mathscr{S}_I^n} \int_{\mathscr{S}} [\![g_i f_{ik}]\!]_k \,\mathrm{d}\mathscr{S} + \sum_{\mathscr{S}\in\mathscr{S}_B^n} \int_{\mathscr{S}} g_i^{\mathrm{L}} f_{ik}^{\mathrm{L}} \bar{n}_k^{\mathrm{L}} \,\mathrm{d}\mathscr{S}. \tag{7}$$

To verify this relation, note that in the sum over all element boundary integrals, the internal faces are counted twice. Therefore, when summing over the internal faces, the contributions from the left and the right must be counted, which is done by taking the jump.

3.3. Flux formulation in arbitrary Lagrangian Eulerian context

Now that the space–time context is well defined, we proceed by expressing the compressible Navier–Stokes equations in the domain $\mathscr{E} \subset \mathbb{R}^4$ as

$$\begin{cases} U_{i,0} + F^e_{ik,k} - (A_{ikrs}U_{r,s})_{,k} = 0 & \text{on } \mathscr{E}, \\ U = U_0 & \text{on } \Omega(t_0), \\ U = \mathscr{B}(U, U^{\mathrm{b}}) & \text{on } \mathscr{Q}, \end{cases}$$

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 derived from the initial condition described in Section 2. The boundary operator is denoted by $\mathscr{B} : \mathbb{R}^{5\times 5} \to \mathbb{R}^5$ and is a function of the internal data U and the boundary data U^{b} derived from the boundary conditions in Section 2. At the far-field boundary, suitable in- and out-flow conditions can be derived using local characteristics. The main idea is that characteristic variables of incoming characteristics are set equal to their free-stream values, while the other variables are extrapolated from within the flow domain, see for example [19]. At solid surfaces, the isothermal no-slip boundary condition is applied.

Following the framework described in [1], we write the compressible Navier–Stokes equations as a first-order system by introducing the auxiliary variable $\Theta(U)$

$$U_{i,0} + F^e_{ik,k} - \Theta_{ik,k} = 0, \tag{8a}$$

$$\Theta_{ik} - A_{ikrs}U_{r,s} = 0. \tag{8b}$$

The *flux formulation* of (8a) is obtained after multiplying by a test function $W \in W_h$, integrating by parts in space–time over an element $\mathscr{K} \in \mathscr{T}_h$ and summing over all elements of the tessellation

$$-\sum_{\mathscr{K}\in\mathscr{T}_h}\int_{\mathscr{K}} \left(W_{i,0}U_i + W_{i,k}(F^e_{ik} - \Theta_{ik})\right) \mathrm{d}\mathscr{K} + \sum_{\mathscr{K}\in\mathscr{T}_h}\int_{\partial\mathscr{K}} W^{\mathrm{L}}_i \left(\widehat{U}_i n^{\mathrm{L}}_0 + (\widehat{F}^e_{ik} - \widehat{\Theta}_{ik})\bar{n}^{\mathrm{L}}_k\right) \mathrm{d}(\partial\mathscr{K}) = 0, \tag{9}$$

where n^{L} is the outward normal vector at $\partial\mathscr{K}$. At the element boundaries, U can be double-valued due to the discontinuous function approximation in each element. Therefore, in order to uniquely define the element boundary integrals and provide a coupling between neighboring elements, we introduce numerical fluxes $\widehat{(\cdot)}$ which depend on both the left and right trace of U at the element boundary. The numerical fluxes will be defined later on.

The auxiliary variable Θ is only needed as an intermediate step in the derivation of the discretization and will be eliminated as we go from the flux formulation to the *primal formulation*, which is expressed solely in terms of the primary unknowns U.

But first we turn to the arbitrary Lagrangian Eulerian (ALE) context in order to accommodate moving and deforming meshes. The flux formulation in ALE context is obtained following the approach described in Van der Vegt and Van der Ven [26]. Using the definition (3) of the space–time normal vector, the boundary integral in (9) becomes

$$\begin{aligned} &\sum_{\mathscr{K}\in\mathscr{T}_h}\int_{\partial\mathscr{K}} W^{\mathrm{L}}_i \left(\widehat{U}_i n^{\mathrm{L}}_0 + (\widehat{F}^e_{ik} - \widehat{\Theta}_{ik})\bar{n}^{\mathrm{L}}_k\right) \mathrm{d}(\partial\mathscr{K}) \\ &\quad = \sum_{\mathscr{K}\in\mathscr{T}_h}\left(\int_{K(t^-_{n+1})} W^{\mathrm{L}}_i \widehat{U}_i \,\mathrm{d}K - \int_{K(t^+_n)} W^{\mathrm{L}}_i \widehat{U}_i \,\mathrm{d}K\right) + \sum_{\mathscr{K}\in\mathscr{T}_h}\int_{\mathscr{Q}} W^{\mathrm{L}}_i (\widehat{F}^e_{ik} - \widehat{U}_i v_k - \widehat{\Theta}_{ik})\bar{n}^{\mathrm{L}}_k \,\mathrm{d}\mathscr{Q}. \end{aligned}$$

The numerical flux $\widehat{U}$ at the faces $K(t^+_n)$ and $K(t^-_{n+1})$ is defined as an upwind flux to ensure causality in time

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

With this numerical flux, the flux formulation in each space–time slab only depends on the previous space–time slab, therefore the summation over the space–time slabs can be dropped and the ALE flux formulation of (8a) becomes

$$- \sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{K}} \left(W_{i,0}U_i + W_{i,k}\left(F_{ik}^e - \Theta_{ik}\right)\right) \mathrm{d}\mathscr{K} + \sum_{K\in\mathscr{T}_h^n} \left(\int_{K(t_{n+1}^-)} W_i^{\mathrm{L}} U_i^{\mathrm{L}} \,\mathrm{d}K - \int_{K(t_n^+)} W_i^{\mathrm{L}} U_i^{\mathrm{R}} \,\mathrm{d}K \right)$$
$$+ \sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{Q}} W_i^{\mathrm{L}} \left(\widehat{F}_{ik}^e - v_k \widehat{U}_i - \widehat{\Theta}_{ik}\right) \bar{n}_k^{\mathrm{L}} \,\mathrm{d}\mathscr{Q} = 0. \tag{10}$$

3.4. The auxiliary variable

The mixed formulation (8) has the disadvantage that both U and Θ have to be stored and solved during a computation. Fortunately, it is possible to eliminate the auxiliary variable using a weak expression for Θ in terms of the primary unknowns U, so only U has to be stored. To derive this expression, we multiply (8b) by a test function $V \in V_h$, integrate by parts in space (twice) over an element $\mathscr{K} \in \mathscr{T}_h$ and sum over all elements of the tessellation

$$\sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{K}} V_{ik}\Theta_{ik} \,\mathrm{d}\mathscr{K} = \sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{K}} V_{ik}A_{ikrs}U_{r,s} \,\mathrm{d}\mathscr{K} + \sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{Q}} V_{ik}^{\mathrm{L}} A_{ikrs}^{\mathrm{L}} \left(\widehat{U}_r - U_r^{\mathrm{L}}\right) \bar{n}_s^{\mathrm{L}} \,\mathrm{d}\mathscr{Q}, \tag{11}$$

where we introduced the numerical flux $\widehat{U}$ after the first integration by parts. In this case, the numerical flux does not have a time contribution because we only integrated in space. Instead of using integrals over the element boundary $\mathscr{Q}$, it is more convenient to use integrals over the element faces $\mathscr{S}$. We therefore apply relation (7) to the element boundary integral of Eq. (11):

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

Now that we explicitly distinguish between internal and boundary faces, we can follow the approach by Bassi and Rebay [3–5] and define the numerical flux as

$$\widehat{U} = \begin{cases} \{\!\{U\}\!\} & \text{at } \mathscr{S}_I^n, \\ U^{\mathrm{b}} & \text{at } \mathscr{S}_B^n. \end{cases}$$

With this choice for the numerical flux at the internal faces and using relation (6) we obtain: $\left[\!\left[V_{ik}A_{ikrs}(\widehat{U}_r - U_r) \right]\!\right]_s = -\{\!\{V_{ik}A_{ikrs}\}\!\}\left[\!\left[U_r\right]\!\right]_s$, which leads to the following expression for the auxiliary variable:

$$\sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{K}} V_{ik}\Theta_{ik} \,\mathrm{d}\mathscr{K} = \sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{K}} V_{ik}A_{ikrs}U_{r,s} \,\mathrm{d}\mathscr{K} - \sum_{\mathscr{S}\in\mathscr{S}_I^n} \int_{\mathscr{S}} \{\!\{V_{ik}A_{ikrs}\}\!\}\left[\!\left[U_r\right]\!\right]_s \mathrm{d}\mathscr{S}$$
$$- \sum_{\mathscr{S}\in\mathscr{S}_B^n} \int_{\mathscr{S}} V_{ik}^{\mathrm{L}} A_{ikrs}^{\mathrm{L}} \left(U_r^{\mathrm{L}} - U_r^{\mathrm{b}}\right) \bar{n}_s^{\mathrm{L}} \,\mathrm{d}\mathscr{S}.$$

In order to obtain an explicit expression for the auxiliary variable, we need to define a *global lifting operator*. The global lifting operator $\mathscr{R} \in \mathbb{R}^{5\times 3}$ is defined in a weak sense as: *Find an* $\mathscr{R} \in V_h$, *such that for all* $V \in V_h$

$$\sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{K}} V_{ik}\mathscr{R}_{ik} \,\mathrm{d}\mathscr{K} = \sum_{\mathscr{S}\in\mathscr{S}_I^n} \int_{\mathscr{S}} \{\!\{V_{ik}A_{ikrs}\}\!\}\left[\!\left[U_r\right]\!\right]_s \mathrm{d}\mathscr{S} + \sum_{\mathscr{S}\in\mathscr{S}_B^n} \int_{\mathscr{S}} V_{ik}^{\mathrm{L}} A_{ikrs}^{\mathrm{L}} \left(U_r^{\mathrm{L}} - U_r^{\mathrm{b}}\right) \bar{n}_s^{\mathrm{L}} \,\mathrm{d}\mathscr{S}. \tag{12}$$

More details on the lifting operator are given in Section 4. According to this definition, the face integrals in the expression for Θ can now be written as element integrals, leading to the weak expression of the auxiliary variable:

$$\sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{K}} V_{ik}\Theta_{ik} \,\mathrm{d}\mathscr{K} = \sum_{\mathscr{K}\in\mathscr{T}_h^n} \int_{\mathscr{K}} V_{ik}\left(A_{ikrs}U_{r,s} - \mathscr{R}_{ik}\right) \mathrm{d}\mathscr{K} \quad \forall V \in V_h. \tag{13}$$

In other words, $\Theta_{ik} = A_{ikrs}U_{r,s} - \mathscr{R}_{ik}$ almost everywhere in $\mathscr{E}_h^n$. The lifting operator $\mathscr{R}$ effectively penalizes the jumps at the faces. For smooth solutions $\mathscr{R} = 0$.

3.5. Primal formulation

The primal formulation can be obtained using the expression (13) for the auxiliary variable Θ. Since $\nabla_h W_h \subset V_h$, the special case $V_{ik} = W_{i,k}$ can be considered in (13), and the auxiliary variable Θ can be replaced in the element integral of (10)

$$\sum_{\mathscr{K}\in\mathscr{T}_h^n}\int_{\mathscr{K}} W_{i,k}\Theta_{ik}\,\mathrm{d}\mathscr{K} = \sum_{\mathscr{K}\in\mathscr{T}_h^n}\int_{\mathscr{K}} W_{i,k}(A_{ikrs}U_{r,s} - \mathscr{R}_{ik})\,\mathrm{d}\mathscr{K}.$$

Now, only the numerical fluxes $\widehat{F}^e$ and $\widehat{\Theta}$ remain to be chosen. We therefore consider the element boundary integrals of (10) and use relation (7) to get the element face integrals

$$\sum_{\mathscr{K}\in\mathscr{T}_h^n}\int_{\mathscr{Q}} W_i^{\mathrm{L}}(\widehat{F}^e_{ik} - v_k\widehat{U}_i - \widehat{\Theta}_{ik})\bar{n}_k^{\mathrm{L}}\,\mathrm{d}\mathscr{Q} = \sum_{\mathscr{S}\in\mathscr{S}_I^n}\int_{\mathscr{S}} [\![W_i(\widehat{F}^e_{ik} - v_k\widehat{U}_i - \widehat{\Theta}_{ik})]\!]_k\,\mathrm{d}\mathscr{S} + \sum_{\mathscr{S}\in\mathscr{S}_B^n}\int_{\mathscr{S}} W_i^{\mathrm{L}}(\widehat{F}^e_{ik} - v_k\widehat{U}_i - \widehat{\Theta}_{ik})\bar{n}_k^{\mathrm{L}}\,\mathrm{d}\mathscr{S}.$$

The inviscid numerical flux $\widehat{F}^e$ is based on the HLLC approximate Riemann solver [6,25,26], because of its computational efficiency, accuracy and straightforward implementation. The HLLC flux is consistent and conservative and is obtained by interpreting the discontinuity between U^{L} and U^{R} at a face $\mathscr{S}$ as a local Riemann problem, which is solved approximately while taking into account the grid velocity $\vec{v}$. Following [26] and using the fact that $n^{\mathrm{R}} = -n^{\mathrm{L}}$, we have

$$[\![W_i(\widehat{F}^e_{ik} - v_k\widehat{U}_i)]\!]_k = \left(W_i^{\mathrm{L}} - W_i^{\mathrm{R}}\right)H_i$$

with $H = H(U^{\mathrm{L}}, U^{\mathrm{R}}, v, \bar{n}^{\mathrm{L}})$ the HLLC flux. At a face moving with velocity v, the HLLC flux is given by

$$H_i = \frac{1}{2}\left(\left(F^e_{ik}\right)^{\mathrm{L}}\bar{n}_k^{\mathrm{L}} - \left(F^e_{ik}\right)^{\mathrm{R}}\bar{n}_k^{\mathrm{R}}\right) + \frac{1}{2}\left((|S^M - v| - |S^{\mathrm{L}} - v|)U^{\mathrm{L}}_{i*} - (v - |S^{\mathrm{L}} - v|)U_i^{\mathrm{L}}\right) + \frac{1}{2}\left((|S^{\mathrm{R}} - v| - |S^M - v|)U^{\mathrm{R}}_{i*} - (v + |S^{\mathrm{R}} - v|)U_i^{\mathrm{R}}\right),$$

with $(F^e)^{\mathrm{L,R}} = F^e(U^{\mathrm{L,R}})$. The intermediate states U_*^{L} and U_*^{R} are given by

$$U_*^{\mathrm{L,R}} = \frac{S^{\mathrm{L,R}} - q^{\mathrm{L,R}}}{S^{\mathrm{L,R}} - S^M}U^{\mathrm{L,R}} + \frac{1}{S^{\mathrm{L,R}} - S^M}\begin{bmatrix} 0 \\ (p_* - p^{\mathrm{L,R}})\bar{n}_k^{\mathrm{L}} \\ p_*S^M - p^{\mathrm{L,R}}q^{\mathrm{L,R}} \end{bmatrix}$$

with $q = \bar{n}_k^{\mathrm{L}}u_k$ the normal velocity and p_* the intermediate pressure

$$p_* = \rho^{\mathrm{L}}(S^{\mathrm{L}} - q^{\mathrm{L}})(S^M - q^{\mathrm{L}}) + p^{\mathrm{L}} = \rho^{\mathrm{R}}(S^{\mathrm{R}} - q^{\mathrm{R}})(S^M - q^{\mathrm{R}}) + p^{\mathrm{R}}.$$

The middle wave speed is defined as

$$S^M = \frac{\rho^{\mathrm{R}}q^{\mathrm{R}}(S^{\mathrm{R}} - q^{\mathrm{R}}) - p^{\mathrm{R}} - \rho^{\mathrm{L}}q^{\mathrm{L}}(S^{\mathrm{L}} - q^{\mathrm{L}}) + p^{\mathrm{L}}}{\rho^{\mathrm{R}}(S^{\mathrm{R}} - q^{\mathrm{R}}) - \rho^{\mathrm{L}}(S^{\mathrm{L}} - q^{\mathrm{L}})}$$

and the left and right wave speeds as

$$S^{\mathrm{L}} = \min\{q^{\mathrm{L}} - a^{\mathrm{L}}, q^{\mathrm{R}} - a^{\mathrm{R}}\}, \quad S^{\mathrm{R}} = \max\{q^{\mathrm{L}} + a^{\mathrm{L}}, q^{\mathrm{R}} + a^{\mathrm{R}}\}$$

with $a = \sqrt{\gamma p/\rho}$ the speed of sound. At the boundary faces, we use $H^{\mathrm{b}} = H(U^{\mathrm{L}}, U^{\mathrm{b}}, v, \bar{n}^{\mathrm{L}})$.

The numerical flux $\widehat{\Theta}$ is defined following Brezzi [9] as a central flux $\widehat{\Theta} = \{\!\{\Theta\}\!\}$, using the weak expression (13) for the auxiliary variable. This is a suitable choice as viscosity does not have a preferred direction. The numerical flux can thus be written as

$$\widehat{\Theta}_{ik}(U^{\mathrm{L}}, U^{\mathrm{R}}) = \begin{cases} \{\!\{A_{ikrs}U_{r,s} - \eta\mathscr{R}_{ik}^{\mathscr{S}}\}\!\} & \text{for } \mathscr{S}\in\mathscr{S}_I^n, \\ A^{\mathrm{b}}_{ikrs}U^{\mathrm{b}}_{r,s} - \eta\mathscr{R}_{ik}^{\mathscr{S}} & \text{for } \mathscr{S}\in\mathscr{S}_B^n, \end{cases}$$

where η is a stabilization constant and $A^{\mathrm{b}} = A(U^b)$ and $U^{\mathrm{b}}_{r,s}$ denotes the derivatives of U at the boundary. The *local lifting operator* $\mathscr{R}^{\mathscr{S}}$ is an approximation of the global lifting operator $\mathscr{R}$ and is preferable because it reduces the width of the stencil to the minimum, see [9]. The local lifting operator $\mathscr{R}^{\mathscr{S}} \in \mathbb{R}^{5\times 3}$ is defined as follows: *Find an* $\mathscr{R}^{\mathscr{S}} \in V_h$, *such that for all* $V \in V_h$

$$\sum_{\mathscr{K}\in\mathscr{T}_h^n}\int_{\mathscr{K}} V_{ik}\mathscr{R}^{\mathscr{S}}_{ik}\,\mathrm{d}\mathscr{K} = \begin{cases} \int_{\mathscr{S}} \{\!\{V_{ik}A_{ikrs}\}\!\}[\![U_r]\!]_s\,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S}\in\mathscr{S}_I,\\ \int_{\mathscr{S}} V^{\mathrm{L}}_{ik}A^{\mathrm{L}}_{ikrs}\big(U^{\mathrm{L}}_r - U^{\mathrm{b}}_r\big)\bar{n}_s\,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S}\in\mathscr{S}_B. \end{cases} \tag{14}$$

With these numerical fluxes the space–time weak formulation of the compressible Navier–Stokes equations in terms of the primary unknown U can be written as follows: *find a* $U \in W_h$, *such that for all* $W \in W_h$

$$\begin{aligned} &-\sum_{\mathscr{K}\in\mathscr{T}_h^n}\int_{\mathscr{K}}\big(W_{i,0}U_i + W_{i,k}\big(F^e_{ik} - A_{ikrs}U_{r,s} + \mathscr{R}_{ik}\big)\big)\,\mathrm{d}\mathscr{K} + \sum_{K\in\mathscr{T}_h^n}\left(\int_{K(t^-_{n+1})} W^{\mathrm{L}}_i U^{\mathrm{L}}_i\,\mathrm{d}K - \int_{K(t^+_n)} W^{\mathrm{L}}_i U^{\mathrm{R}}_i\,\mathrm{d}K\right)\\ &+\sum_{\mathscr{S}\in\mathscr{S}_I^n}\int_{\mathscr{S}}\big(W^{\mathrm{L}}_i - W^{\mathrm{R}}_i\big)H_i\,\mathrm{d}\mathscr{S} + \sum_{\mathscr{S}\in\mathscr{S}_B^n}\int_{\mathscr{S}} W^{\mathrm{L}}_i H^{\mathrm{b}}_i\,\mathrm{d}\mathscr{S} - \sum_{\mathscr{S}\in\mathscr{S}_I^n}\int_{\mathscr{S}}[\![W_i]\!]_k\{\!\{A_{ikrs}U_{r,s} - \eta\mathscr{R}^{\mathscr{S}}_{ik}\}\!\}\,\mathrm{d}\mathscr{S}\\ &-\sum_{\mathscr{S}\in\mathscr{S}_B^n}\int_{\mathscr{S}} W^{\mathrm{L}}_i\Big(A^{\mathrm{b}}_{ikrs}U^{\mathrm{b}}_{r,s} - \eta\mathscr{R}^{\mathscr{S}}_{ik}\Big)\bar{n}^{\mathrm{L}}_k\,\mathrm{d}\mathscr{S} = 0, \end{aligned} \tag{15}$$

where we used the relation $[\![W_i\widehat{\Theta}_{ik}]\!]_k = [\![W_i]\!]_k\widehat{\Theta}_{ik}$, which follows from the viscous numerical flux being conservative: $\widehat{\Theta}(U^{\mathrm{L}},U^{\mathrm{R}}) = \widehat{\Theta}(U^{\mathrm{R}},U^{\mathrm{L}})$.

Discontinuous Galerkin methods are known to suffer from numerical oscillations around shocks and sharp gradients. This problem can be overcome using a slope limiter (see for example [12]), but we prefer the artificial dissipation proposed in [26] as it allows convergence to steady-state up to machine precision. We refer to [26] for a detailed description of the artificial dissipation operator.

4. Algebraic system

In this section, the space–time discretization of the compressible Navier–Stokes equations is completed by defining the basis functions, computing the local lifting operator and constructing the system of algebraic equations.

4.1. Basis functions

We use polynomials of degree k to represent the trial function U and the test function W in each element $\mathscr{K} \in \mathscr{T}_h^n$

$$\begin{aligned} U(t,\bar{x})|_{\mathscr{K}} &= \widehat{U}_m\psi_m(t,\bar{x}),\\ W(t,\bar{x})|_{\mathscr{K}} &= \widehat{W}_l\psi_l(t,\bar{x}), \end{aligned}$$

with $\widehat{(\cdot)}$ the expansion coefficients and 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 [26]. The basis functions ψ are given by

$$\psi_m = \begin{cases} 1 & m = 0,\\ \phi_m(t,\bar{x}) - \frac{1}{|K_j(t^-_{n+1})|}\int_{K_j(t^-_{n+1})}\phi_m(t,\bar{x})\,\mathrm{d}K, & m = 1,\ldots,4, \end{cases}$$

where the functions ϕ in an element $\mathscr{K}$ are related to the basis functions $\hat{\phi}$ on the master element $\widehat{\mathscr{K}}$ through the mapping G

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

where ξ are the local coordinates in the master element $\widehat{\mathscr{K}}$ defined in Section 3.

4.2. Lifting operators

The global and local lifting operators contained in the primal formulation (15) must be computed first in order to obtain the system of algebraic equations for the expansion coefficients $\widehat{U}$ of the trial function U. The volume integral containing the global lifting operator can simply be replaced by face integrals using its definition (12)

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

These face integrals can be directly computed by replacing the test and trial functions by their polynomial expansions. The local lifting operator, however, cannot be computed directly. Like the test and trial functions, it is represented by a linear polynomial

$$\mathscr{R}^{\mathscr{S}}(t,\bar{x})|_{\mathscr{K}} = \widehat{R}_j\psi_j(t,\bar{x})$$

and a small linear system must be solved for the expansion coefficients $\widehat{R}_j$. The linear system follows from the definition of the local lifting operator (14). By this definition, the local lifting operator is only non-zero on the two elements $\mathscr{K}^{\mathrm{L}}$ and $\mathscr{K}^{\mathrm{R}}$ connected to the face $\mathscr{S}\in\mathscr{S}_I^n$, hence

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

Since V is an arbitrary test function, this is equivalent with the two following equations:

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

where the superscript L, R refers to the traces from either the left or right element. Replacing $\mathscr{R}^{\mathscr{S}}$ by its polynomial approximation leads to two systems of linear equations for the expansion coefficients $\widehat{R}_{ikj}$ of $\mathscr{R}_{ik}^{\mathscr{S}}$ on $\mathscr{S}\in\mathscr{S}_I$

$$\widehat{R}_{ikj}^{\mathrm{L,R}}\int_{\mathscr{K}^{\mathrm{L,R}}}\psi_l\psi_j\,\mathrm{d}\mathscr{K} = \frac{1}{2}\int_{\mathscr{S}}\psi_l^{\mathrm{L,R}}A_{ikrs}^{\mathrm{L,R}}[\![U_r]\!]_s\,\mathrm{d}\mathscr{S}.$$

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

$$\widehat{R}_{ikj}^{\mathrm{L,R}} = \frac{1}{2}(M^{-1})_{jl}^{\mathrm{L,R}}\int_{\mathscr{S}}\psi_l^{\mathrm{L,R}}A_{ikrs}^{\mathrm{L,R}}[\![U_r]\!]_s\,\mathrm{d}\mathscr{S}. \tag{16}$$

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

$$\widehat{R}_{ikj}^{\mathrm{L}} = (M^{-1})_{jl}^{\mathrm{L}}\int_{\mathscr{S}}\psi_l^{\mathrm{L}}A_{ikrs}^{\mathrm{L}}\big(U_r^{\mathrm{L}}-U_r^{\mathrm{b}}\big)\bar{n}_s^{\mathrm{L}}\,\mathrm{d}\mathscr{S}. \tag{17}$$

Note that the mass matrices M only have to be inverted once per element in each space–time slab, after which the local lifting operator can be computed as a small matrix–vector multiplication.

4.3. Equations for the expansion coefficients of the flow field

The system of algebraic equations for the expansion coefficients $\widehat{U}$ of the trial function U is obtained by replacing U and the test function W in (15) by their polynomial expansions. We distinguish between the inviscid and viscous part

$$\mathscr{L}^e(\widehat{U}^n,\widehat{U}^{n-1}) + \mathscr{L}^v(\widehat{U}^n) = 0.$$

The term $\mathscr{L}^e$ corresponds to the inviscid part of the residuals and is defined as

$$\mathscr{L}_{il}^e = -\sum_{\mathscr{K}\in\mathscr{T}_h^n}(A_{il}+B_{il}) + \sum_{K\in\mathscr{T}_h^n}C_{il} + \sum_{\mathscr{S}\in\mathscr{S}_{I,B}^n}E_{il}$$

with $i = 1,\ldots,5$ the equation number, $l = 0,\ldots,4$, the index of the expansion coefficients and the terms A, B, C and E defined as

$$A_{il} = \int_{\mathscr{K}} \psi_{l,0} U_i \,\mathrm{d}\mathscr{K}, \tag{18}$$

$$B_{il} = \int_{\mathscr{K}} \psi_{l,k} F^e_{ik} \,\mathrm{d}\mathscr{K}, \tag{19}$$

$$C_{il} = \int_{K(t^-_{n+1})} \psi^{\mathrm{L}}_l U^{\mathrm{L}}_i \,\mathrm{d}K - \int_{K(t^+_n)} \psi^{\mathrm{L}}_l U^{\mathrm{R}}_i \,\mathrm{d}K, \tag{20}$$

$$E_{il} = \begin{cases} \int_{\mathscr{S}} (\psi^{\mathrm{L}}_l - \psi^{\mathrm{R}}_l) H_i \,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S} \in \mathscr{S}_I, \\ \int_{\mathscr{S}} \psi^{\mathrm{L}}_l H^{\mathrm{b}}_i \,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S} \in \mathscr{S}_B, \end{cases} \tag{21}$$

with $F^e_{ik} = F^e_{ik}(U)$ the Euler flux and $H_i = H_i(U^{\mathrm{L}}, U^{\mathrm{R}}, v, \bar{n}^{\mathrm{L}})$ the HLLC flux. The term $\mathscr{L}^v$ corresponds to the viscous part of the residual and is defined as

$$\mathscr{L}^v_{il} = \sum_{\mathscr{K}\in\mathscr{T}^n_h} D_{il} + \sum_{\mathscr{S}\in\mathscr{S}^n_{I,B}} (-F_{il} - G_{il} + H_{il})$$

with

$$D_{il} = \int_{\mathscr{K}} \psi_{l,k} A_{ikrs} U_{r,s} \,\mathrm{d}\mathscr{K}, \tag{22}$$

$$F_{il} = \begin{cases} \int_{\mathscr{S}} \{\!\{\psi_{l,k} A_{ikrs}\}\!\} [\![U_r]\!]_s \,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S} \in \mathscr{S}_I, \\ \int_{\mathscr{S}} \psi^{\mathrm{L}}_{l,k} A^{\mathrm{L}}_{ikrs} (U^{\mathrm{L}}_r - U^{\mathrm{b}}_r) \bar{n}^{\mathrm{L}}_s \,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S} \in \mathscr{S}_B, \end{cases} \tag{23}$$

$$G_{il} = \begin{cases} \int_{\mathscr{S}} [\![\psi_l]\!]_k \{\!\{A_{ikrs} U_{r,s}\}\!\} \,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S} \in \mathscr{S}_I, \\ \int_{\mathscr{S}} \psi^{\mathrm{L}}_l \left(A^{\mathrm{b}}_{ikrs} U^{\mathrm{b}}_{r,s}\right) \bar{n}^{\mathrm{L}}_k \,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S} \in \mathscr{S}_B, \end{cases} \tag{24}$$

$$H_{il} = \begin{cases} \eta \int_{\mathscr{S}} [\![\psi_l]\!]_k \{\!\{\mathscr{R}^{\mathscr{S}}_{ik}\}\!\} \,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S} \in \mathscr{S}_I, \\ \eta \int_{\mathscr{S}} \psi^{\mathrm{L}}_l \mathscr{R}^{\mathscr{S}}_{ik} \bar{n}^{\mathrm{L}}_k \,\mathrm{d}\mathscr{S} & \text{for } \mathscr{S} \in \mathscr{S}_B, \end{cases} \tag{25}$$

with $\mathscr{R}^{\mathscr{S}}_{ik} = \mathscr{R}^{\mathscr{S}}_{ik}(U)$ the local lifting operator and $A_{ikrs} = A_{ikrs}(U)$ the homogeneity tensor.

Thus, the space–time discontinuous Galerkin discretization of the compressible Navier–Stokes equations results in a system of coupled non-linear equations for the expansion coefficients, which is solved by adding a pseudo-time derivative

$$\frac{\partial \widehat{U}^n}{\partial \tau} = -\frac{1}{\Delta t}\left(\mathscr{L}^e(\widehat{U}^n, \widehat{U}^{n-1}) + \mathscr{L}^v(\widehat{U}^n)\right)$$

and integrating to steady-state in pseudo-time. Different pseudo-time stepping methods suitable for this purpose are presented in [18], where we analyze their stability in pseudo-time and compare their efficiency. Computing the viscous part $\mathscr{L}^v$ of the residual takes roughly twice the CPU time needed for the inviscid part $\mathscr{L}^e$,

Table 1
Relative computational effort

	Equation	Notation	CPU time (%)
Time flux	(18)	A_{il}	1.7
	(20)	C_{il}	1.8
Euler flux	(19)	B_{il}	4
	(21)	E_{il}	25
Viscous flux	(22)	D_{il}	10
	(23)	F_{il}	10
	(24)	G_{il}	7
	(25)	H_{il}	40.5

see Table 1. The stabilization term (25) is by far the most expensive as it requires the expansion coefficients of the local lifting operator, which must be computed first using (16) and (17).

5. Numerical results

The space–time discontinuous Galerkin method for the compressible Navier–Stokes equations is implemented in the NLR program HEXADAP and, in this section, numerical results are presented. We consider a model problem of two-dimensional laminar dynamic stall and the three-dimensional vortex flow around a delta wing.

5.1. Laminar dynamic stall of NACA0012 airfoil

We consider the laminar flow over a NACA0012 airfoil in rapid pitch-up maneuver, comparable to the situation described in [22,31]. The flow is characterized by a complex interaction of an unsteady leading-edge vortex, shear layer vortices and trailing edge vortex, resulting in the detachment of the leading edge vortex: the dynamic stall phenomenon. The complexity of the unsteady flow and the significant grid movement make this a challenging test case for the space–time discretization of the Navier–Stokes equations, where the deforming elements in the neighborhood of the moving airfoil are accommodated with the deformation algorithm proposed in [26].

In this case, the far-field Reynolds number is $Re_\infty = 10^4$ and the Mach number $M_\infty = 0.2$, based on a non-dimensionalization with the reference length c of the airfoil, the free-stream speed of sound a_∞, density ρ_∞ and temperature T_∞. The pitch axis is situated at 25% from the leading edge and the airfoil rotates in such a way that the angle of attack α evolves as follows:

$$\alpha(t) = a + bt - a\exp(-ct). \tag{26}$$

The coefficients are $a = -1.2455604$, $b = 2.2918312$, $c = 1.84$ and the time t ranges from 0 to 25. With these coefficients, the movement of the airfoil is the same as the movement of the NACA0015 used in [31]. At time $t = 0$, both $\alpha = 0$ and $\mathrm{d}\alpha/\mathrm{d}t = 0$ and, after a short transition, the movement becomes mainly linear. The basis functions in the discretization are linear and the stabilization constant is $\eta = 5$.

Remark 1. Although, in this paper, we limit ourselves to linear basis functions, the space–time method allows higher-order basis functions to be used both in space and time. We refer to [24] for the detailed analysis of high-order space–time approximations.

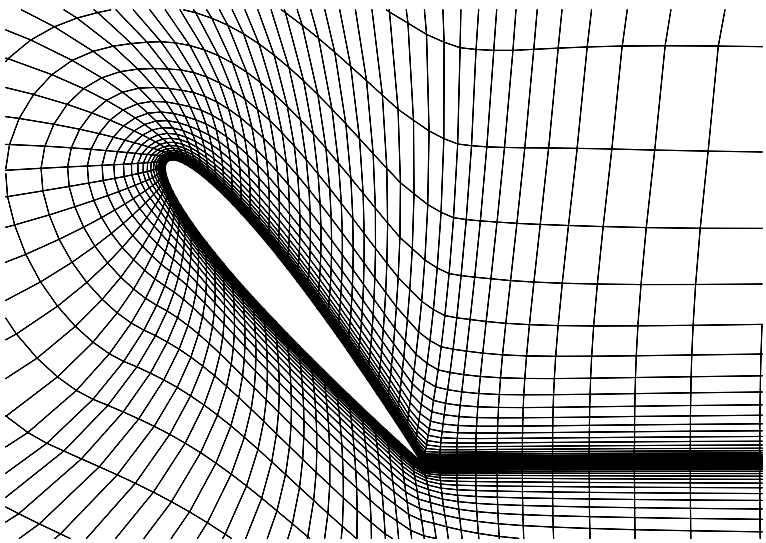

Fig. 3. Overview of the mesh deformation in the dynamic stall case ($\alpha = 50°$).

Since the flow is still laminar, the boundary layer thickness is estimated as $b \approx 5/\sqrt{Re}$ and the computational mesh should be fine enough to accurately represent this layer. In this case, $b \approx 0.05$ and we use a C-type grid with 112×38 elements which results in 14 elements in this boundary layer. The dimensionless physical time step is $\Delta t = 0.005$, based on the non-dimensionalization with the chord length L and the far-field speed of sound a_∞. Each physical time step requires about 20 pseudo-time iterations to solve the algebraic system. At each step the mesh moves and deforms according to the motion of the airfoil prescribed by (26), see Figs. 3–5 for the details of the mesh at a 50° angle of attack. At this point, the mesh lines are no longer perpendicular to the airfoil geometry and are sharply bend near the trailing edge. Yet, even on this mesh of reduced quality, the space–time discontinuous Galerkin method still performs well as can be seen in Figs. 6–8 which show the streamlines at angles of attack $\alpha = 30°$, 40° and 50°, respectively. The sudden drop

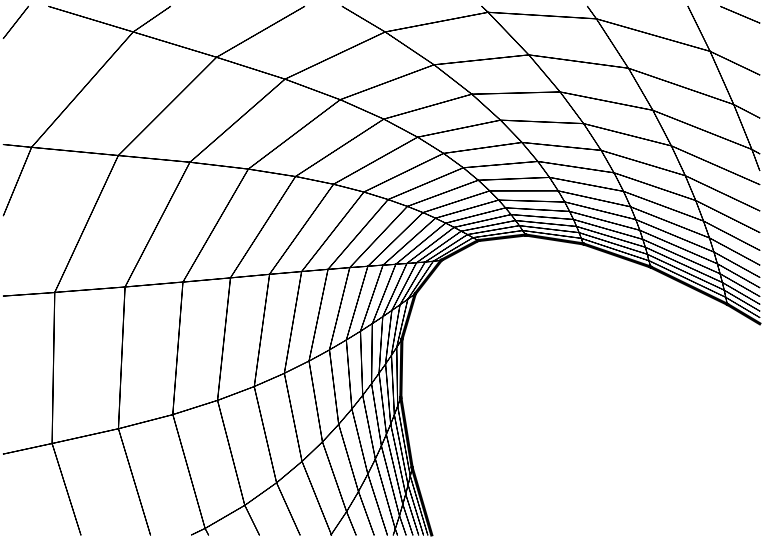

Fig. 4. Detail of the mesh deformation near the leading edge in the dynamic stall case ($\alpha = 50°$).

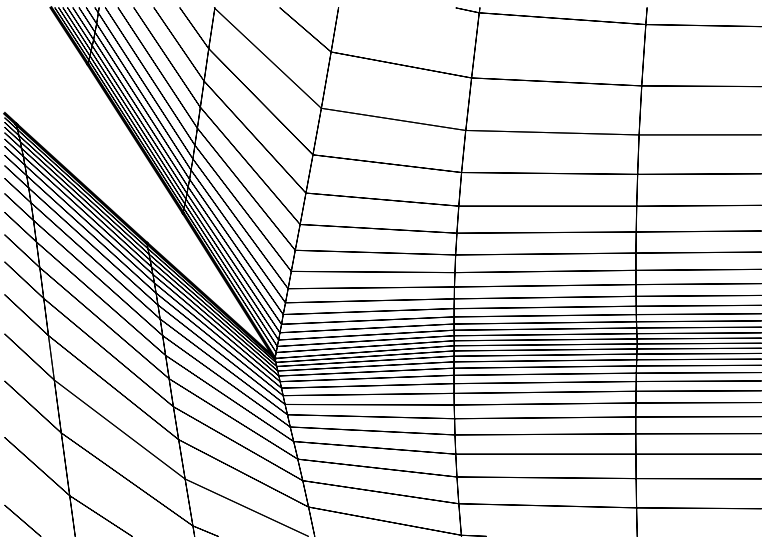

Fig. 5. Detail of the mesh deformation near the trailing edge in the dynamic stall case ($\alpha = 50°$).

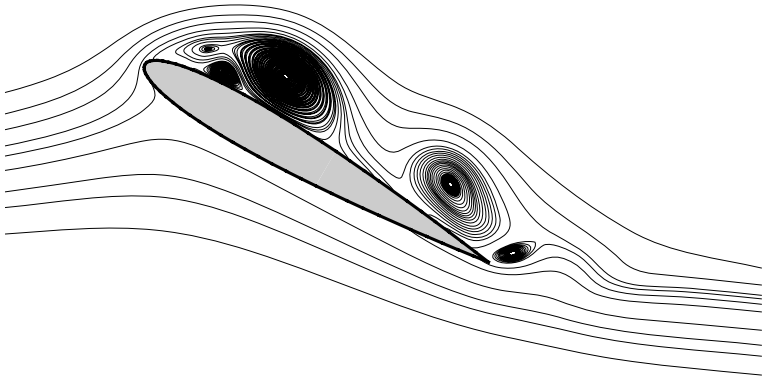

Fig. 6. Streamlines in the dynamic stall case for $\alpha = 30°$.

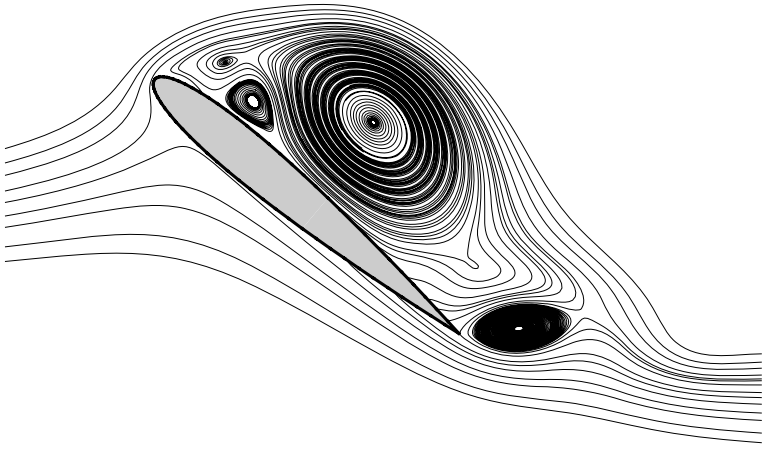

Fig. 7. Streamlines in the dynamic stall case when $\alpha = 40°$.

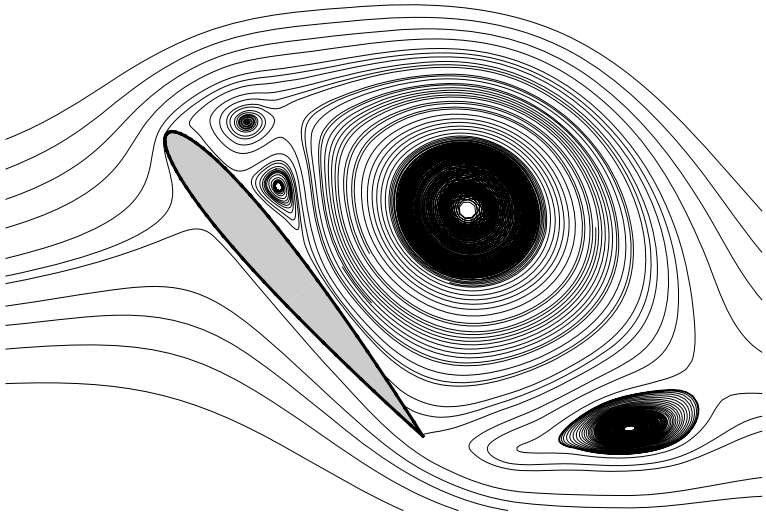

Fig. 8. Streamlines in the dynamic stall case when $\alpha = 50°$.

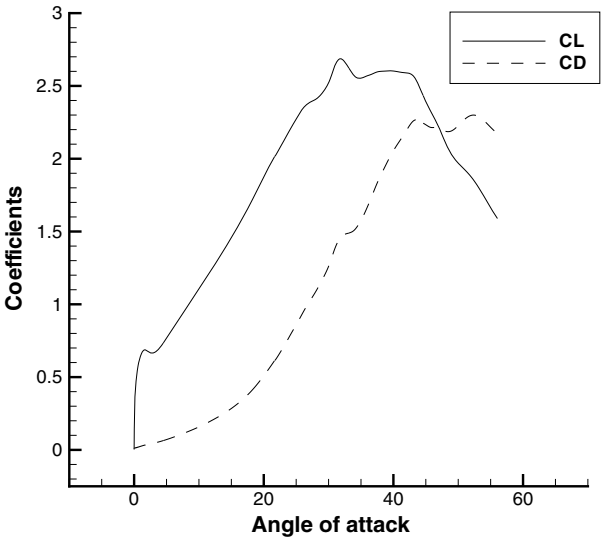

Fig. 9. The lift and drag coefficients.

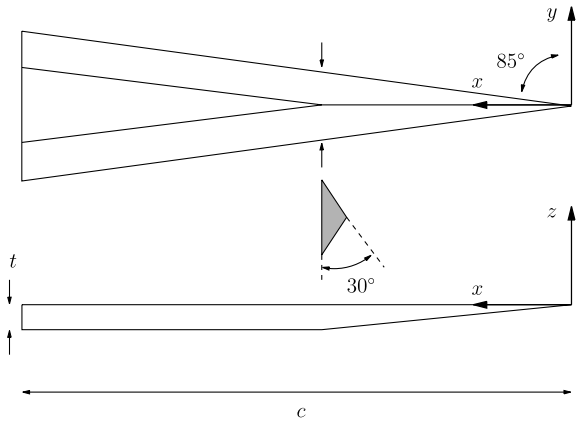

Fig. 10. The geometry of the delta wing ($t/c = 0.024$).

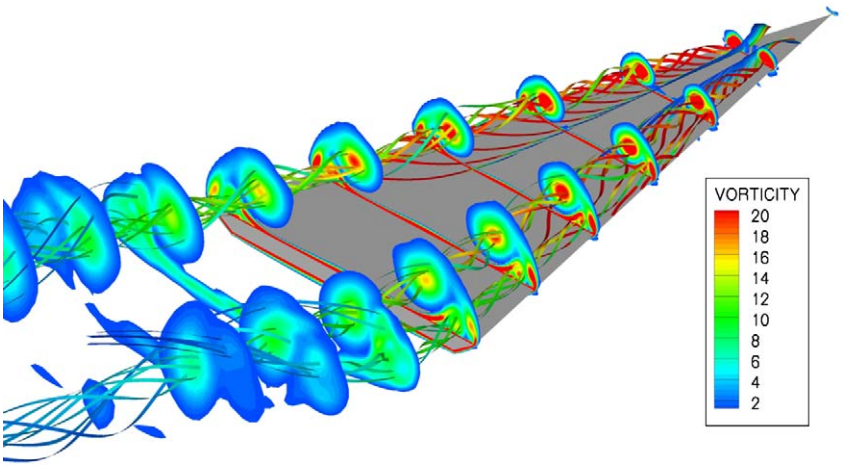

Fig. 11. Streamlines and vorticity in several cross-sections of the delta wing.

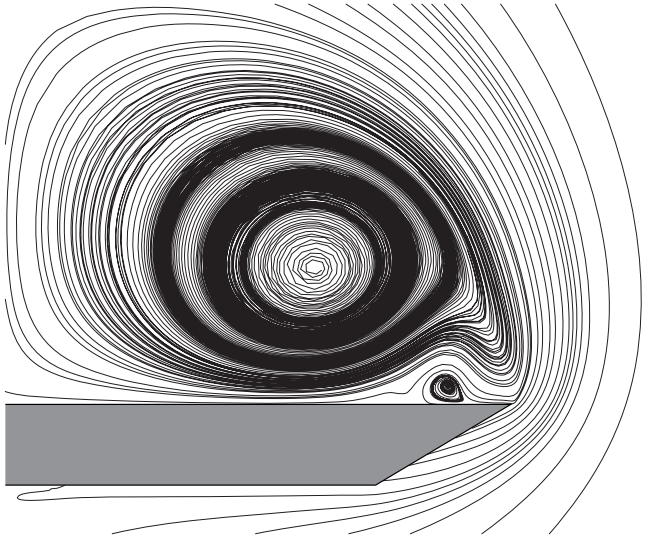

Fig. 12. Streamlines around the delta wing (cross-section $x/c = 0.6$).

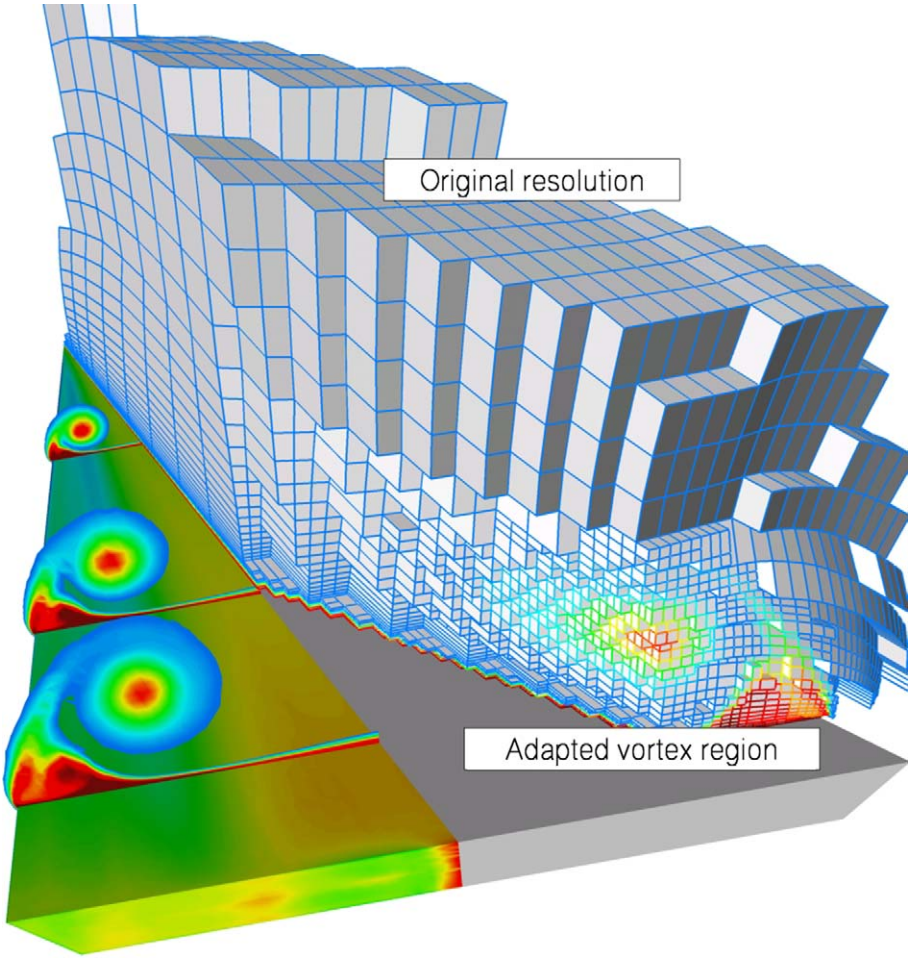

Fig. 13. Impression of the vorticity based mesh adaptation.

in lift and increase in drag associated with the detachment of the leading edge vortex (between $\alpha = 40°$ and 50°) can clearly be seen in Fig. 9, where we show the lift and drag coefficients as a function of the angle of attack.

We conclude therefore that the space–time discontinuous Galerkin method combined with grid movement and deformation has significant potential to simulate the complex flow phenomenon which occur in dynamic stall situations.

5.2. 3D Delta wing with mesh adaptation

To test the performance of the space–time method with local mesh adaptation in a 3D situation, we consider the steady state flow around the 85° delta wing used in the experiments by Riley and Lowson [23], see

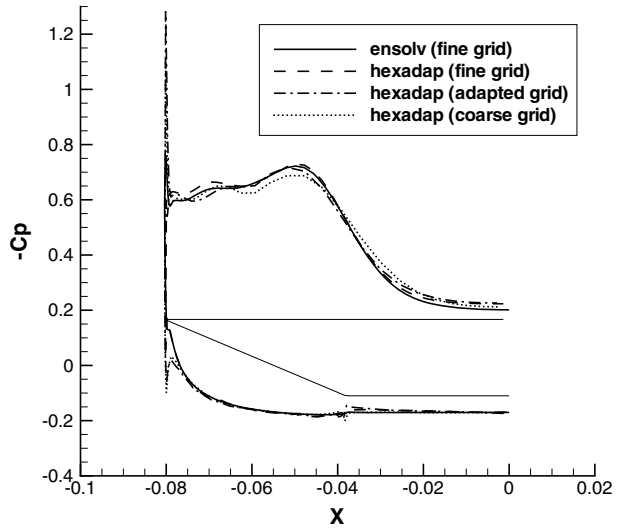

Fig. 14. Pressure coefficient C_p at cross-section $x/c = 0.3$ of the delta wing.

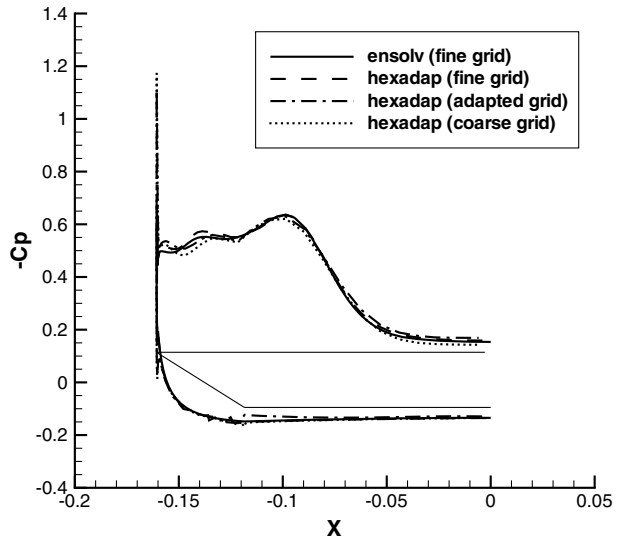

Fig. 15. Pressure coefficient C_p at cross-section $x/c = 0.6$ of the delta wing.

Fig. 10 for details on the geometry. The flow is symmetric with, along both sides of the wing, a large steady vortex and two secondary vortices, see for example the similar situation in [23].

We consider the case with far-field Reynolds number $Re_\infty = 4 \times 10^4$, Mach number $M_\infty = 0.3$ and angle of attack $\alpha = 12.5°$. The non-dimensionalization is similar to the dynamic stall case. We compute the solution on a coarse mesh with 208,896 elements and on a fine mesh with 1,671,168 elements. The basis functions are linear and the stabilization constant is $\eta = 7$. Since this is a steady-state case, we take one huge physical time step $\Delta t = 10^{21}$ (non-dimensionalization with the chord length L and farfield speed of sound a_∞) and solve the algebraic system in about 1000 pseudo-time iterations on the coarse mesh and 5000 on the fine mesh. Fig. 11 shows

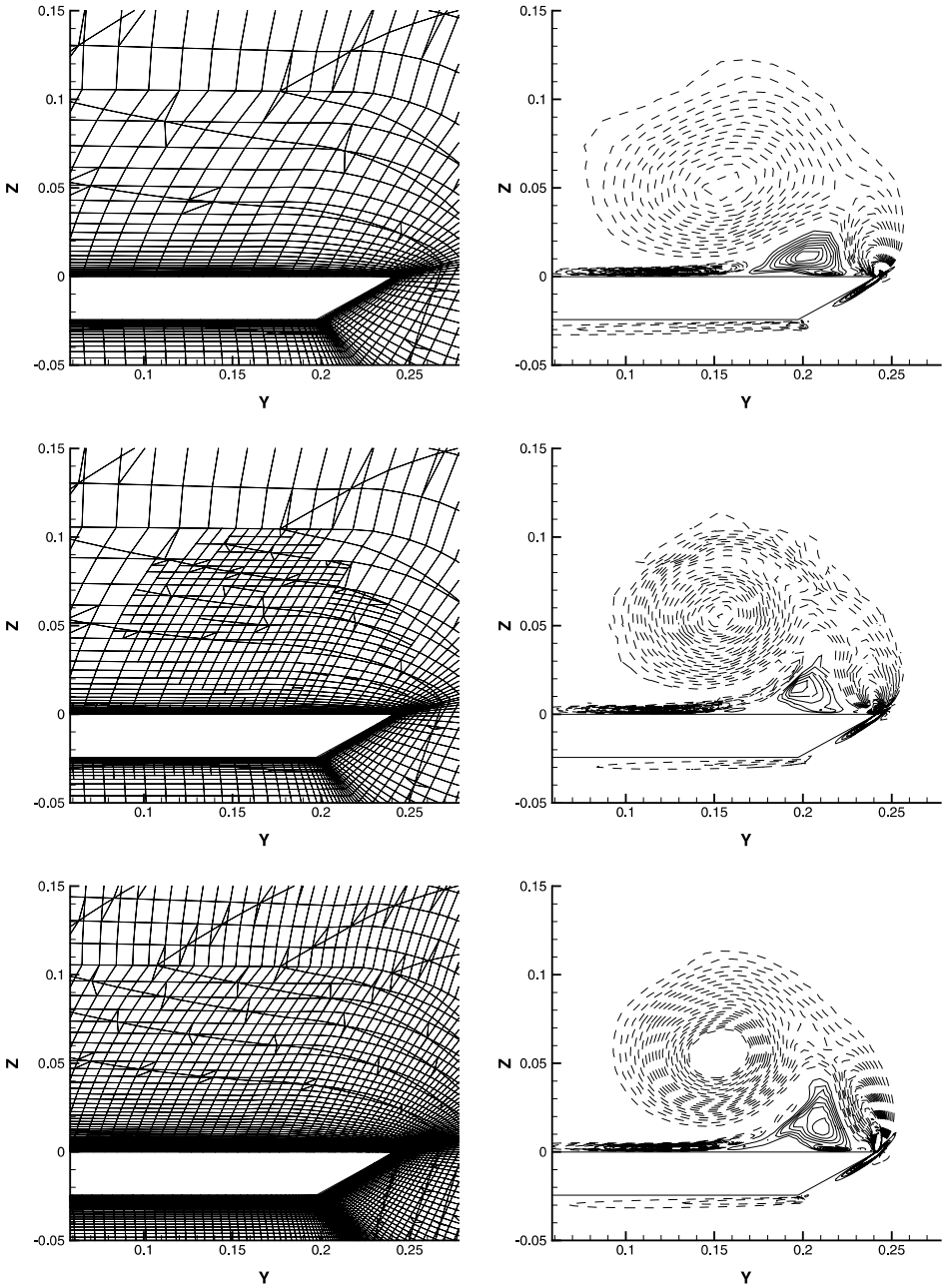

Fig. 16. Grid and helicity isolines at cross-section $x/c = 0.9$ for the coarse, adapted and fine mesh. The helicity ranges from -5 to 2 with step size 0.2, the negative part being represented with solid lines, the positive part with dashed lines.

the streamlines and the vorticity in several cross-sections of the flow field computed on the fine mesh. The two main vortices are clearly visible as well as the secondary vortices near the edges of the wing, see also Fig. 12 for the streamlines in cross-sections $x/c = 0.6$.

In the local mesh adaptation procedure, we start with the solution on the coarse mesh, then refine the mesh in the regions with the highest vorticity, thereby increasing the number of elements by 10%. Then we compute the solution on the adapted mesh and repeat the same procedure until the mesh has been adapted three times. The final adapted mesh has 286,416 elements, see Fig. 13 for an impression of the 3D adaptation. Note that the refinement mainly takes place in the stream-wise direction.

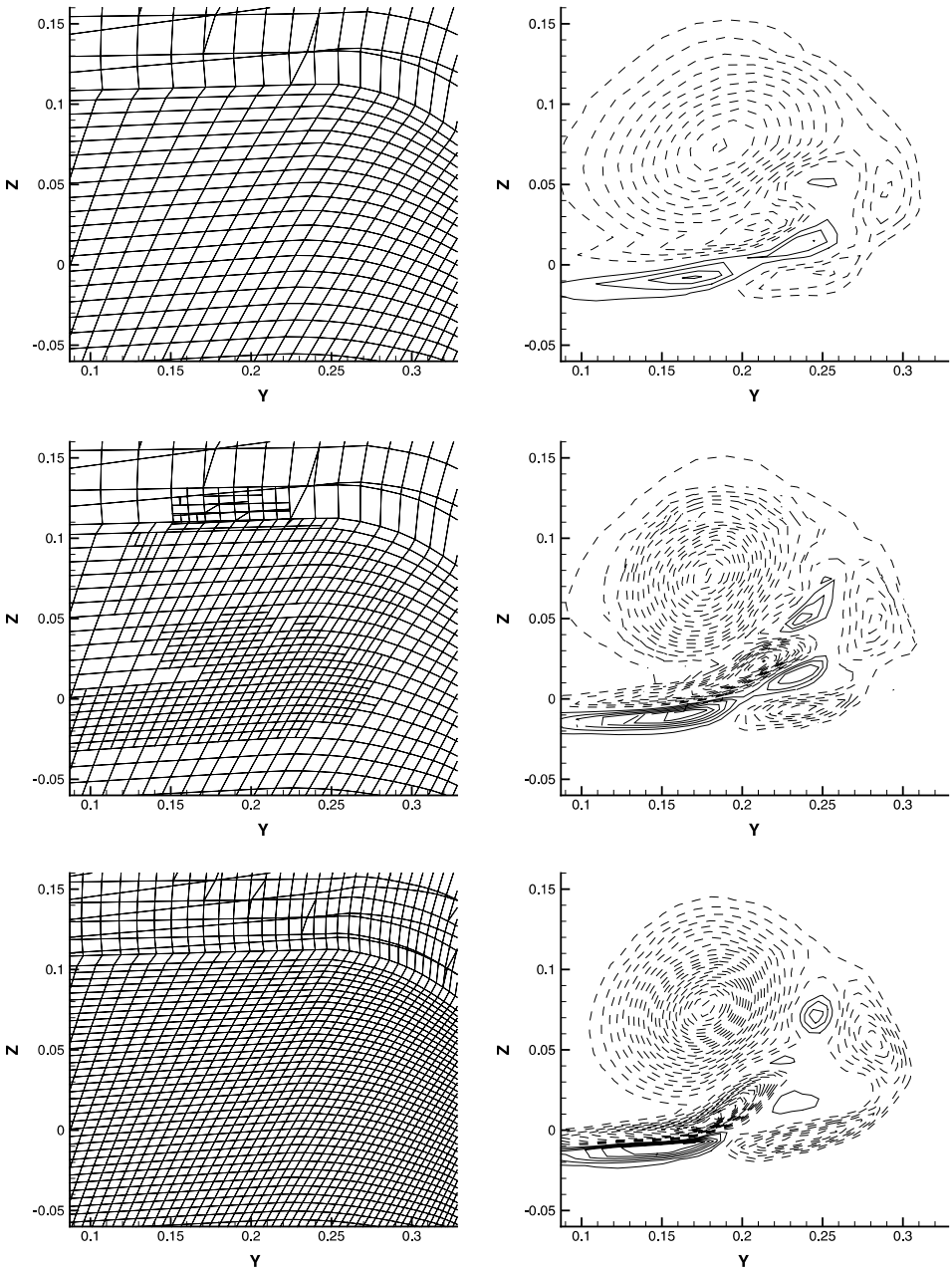

Fig. 17. Grid and helicity isolines at cross-section $x/c = 1.1$ for the coarse, adapted and fine mesh. The helicity ranges from -5 to 2 with step size 0.2, the negative part being represented with solid lines, the positive part with dashed lines.

The effect of vorticity driven mesh adaptation is shown by comparing the pressure coefficient and the helicity ($u \cdot \omega$ with ω the vorticity) obtained on the coarse and adapted mesh with those on the fine mesh. Figs. 14 and 15 show the pressure coefficient C_p on the delta wing at cross-sections $x/c = 0.3$ and $x/c = 0.6$, respectively. In these figures, we also show the C_p computed with the NLR finite volume code ENSOLV [20] on the fine mesh and found some small differences. For instance, the suction peak with HEXADAP on the coarse and adapted mesh is higher than the one on the fine mesh and the one obtained with ENSOLV. Also, the sharp edge at the bottom of the wing induces a small oscillation in C_p with HEXADAP on the coarse and adapted mesh, while the fine mesh results of both HEXADAP and ENSOLV are smoother. We conclude that the pressure coefficient is not very sensitive to the mesh quality, even the coarse mesh gives reasonable results.

The helicity, however, is much more sensitive to the mesh quality as can be seen in Figs. 16 and 17 where we show the mesh and helicity contours in cross-sections $x/c = 0.9$ and $x/c = 1.1$, respectively. At $x/c = 0.9$, the results on the coarse mesh are rather poor, while the results on the adapted mesh are much closer to those on the fine mesh. Downstream of the delta wing ($x/c = 1.1$), the advantage of grid adaptation is even clearer: on the coarse grid the details in the helicity are almost lost, while on the adapted grid the helicity still strongly resembles the one on fine mesh. Since the adapted mesh has five times less elements than the fine grid, the computational cost is much lower.

This demonstrates that a solution adaptive space–time method can result in significant cost savings when applied to vortex dominated viscous flows.

Remark 2. Cost saving is important as DG methods are known to be computationally expensive in comparison to finite volume methods. In [29], the computational complexity of the space–time DG method for the Euler equations is considered in detail and in [27] it was shown that the CPU time per degree of freedom is comparable to a Jameson finite volume solver. In Table 1, the relative cost of the viscous part is given and in [18] the efficiency of the pseudo-time stepping method is analyzed.

6. Discussion and conclusions

In this article, we presented a space–time discontinuous Galerkin method for the compressible Navier–Stokes equations aimed at the accurate solution of time dependent problems on moving and deforming grids. The method does not distinguish between space and time, thereby providing optimal flexibility to accommodate time-dependent boundaries and element deformation. We have discussed our choices for the space–time numerical fluxes and emphasized the treatment of the viscous part of the Navier–Stokes equations needed to maintain locality of the stencil and optimal order of accuracy.

The method was implemented in the NLR program HEXADAP, parallelized using OpenMP and typically runs at 6.4 Gflops/s on 8 processors of an SGI Altix supercomputer. This method accurately handles complex aerodynamic problems, which we demonstrated by computing the flow around a 3D delta wing and around a 2D NACA0012 airfoil in rapid pitch-up maneuver. We found that the results for the 3D delta wing on a coarse adapted mesh are comparable with those on a (costly) fine mesh. The method remains accurate even in the case of significant mesh movement and deformation as required by the NACA0012 airfoil in rapid pitch-up maneuver.

Acknowledgments

This research has been conducted in the STW project TWI.5541, entitled *Advanced simulation techniques for vortex dominated flows in aerodynamics*. The financial support from STW and the National Aerospace Laboratory NLR is gratefully acknowledged.

Appendix A. The homogeneity tensor

The elements of the homogeneity tensor (A_{ikrs}) are calculated by applying the definition:

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

for $i, r = 1, \ldots, 5$ and $k, s = 1, \ldots, 3$ and by using the Stokes hypothesis $3\lambda + 2\mu = 0$ to eliminate λ. For clarity's sake, the elements are grouped in the following matrices:

$$\begin{aligned}
A11 &:= A|^{k=1}_{s=1}, \quad A12 := A|^{k=1}_{s=2}, \quad A13 := A|^{k=1}_{s=3}, \\
A21 &:= A|^{k=2}_{s=1}, \quad A22 := A|^{k=2}_{s=2}, \quad A23 := A|^{k=2}_{s=3}, \\
A31 &:= A|^{k=3}_{s=1}, \quad A32 := A|^{k=3}_{s=2}, \quad A33 := A|^{k=3}_{s=3},
\end{aligned}$$

which are given by

$$A11 = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -\frac{4}{3}\mu u_1 & \frac{4}{3}\mu & 0 & 0 & 0 \\ -\mu u_2 & 0 & \mu & 0 & 0 \\ -\mu u_3 & 0 & 0 & \mu & 0 \\ A_{5111} & \left(\frac{4}{3}\mu - \frac{\kappa}{c_v}\right)u_1 & \left(\mu - \frac{\kappa}{c_v}\right)u_2 & \left(\mu - \frac{\kappa}{c_v}\right)u_3 & \frac{\kappa}{c_v} \end{bmatrix}$$

with

$$A_{5111} = -\frac{1}{3}\mu u_1^2 - \mu \|\vec{u}\|^2 - \frac{\kappa}{c_v}(E - \|\vec{u}\|^2),$$

$$A12 = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3}\mu u_2 & 0 & -\frac{2}{3}\mu & 0 & 0 \\ -\mu u_1 & \mu & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{3}\mu u_1 u_2 & \mu u_2 & -\frac{2}{3}\mu u_1 & 0 & 0 \end{bmatrix},$$

$$A13 = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3}\mu u_3 & 0 & 0 & -\frac{2}{3}\mu & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\mu u_1 & \mu & 0 & 0 & 0 \\ -\frac{1}{3}\mu u_1 u_3 & \mu u_3 & 0 & -\frac{2}{3}\mu u_1 & 0 \end{bmatrix},$$

$$A21 = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -\mu u_2 & 0 & \mu & 0 & 0 \\ \frac{2}{3}\mu u_1 & -\frac{2}{3}\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{3}\mu u_1 u_2 & -\frac{2}{3}\mu u_2 & \mu u_1 & 0 & 0 \end{bmatrix},$$

$$A22 = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -\mu u_1 & \mu & 0 & 0 & 0 \\ -\frac{4}{3}\mu u_2 & 0 & \frac{4}{3}\mu & 0 & 0 \\ -\mu u_3 & 0 & 0 & \mu & 0 \\ A_{5212} & \left(\mu - \frac{\kappa}{c_v}\right)u_1 & \left(\frac{4}{3}\mu - \frac{\kappa}{c_v}\right)u_2 & \left(\mu - \frac{\kappa}{c_v}\right)u_3 & \frac{\kappa}{c_v} \end{bmatrix}$$

with

$$A_{5212} = -\frac{1}{3}u_2^2 - \mu \|\vec{u}\|^2 - \frac{\kappa}{c_v}(E - \|\vec{u}\|^2),$$

$$A23 = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3}\mu u_3 & 0 & 0 & -\frac{2}{3}\mu & 0 \\ -\mu u_2 & 0 & \mu & 0 & 0 \\ -\frac{1}{3}\mu u_2 u_3 & 0 & \mu u_3 & -\frac{2}{3}\mu u_2 & 0 \end{bmatrix},$$

$$A31 = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -\mu u_3 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3}\mu u_1 & -\frac{2}{3}\mu & 0 & 0 & 0 \\ -\frac{1}{3}\mu u_1 u_3 & -\frac{2}{3}\mu u_3 & 0 & \mu u_1 & 0 \end{bmatrix},$$

$$A32 = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\mu u_3 & 0 & 0 & \mu & 0 \\ \frac{2}{3}\mu u_2 & 0 & -\frac{2}{3}\mu & 0 & 0 \\ -\frac{1}{3}\mu u_2 u_3 & 0 & -\frac{2}{3}\mu u_3 & \mu u_2 & 0 \end{bmatrix},$$

$$A33 = \frac{1}{\rho}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -\mu u_1 & \mu & 0 & 0 & 0 \\ -\mu u_2 & 0 & \mu & 0 & 0 \\ -\frac{4}{3}\mu u_3 & 0 & 0 & \frac{4}{3}\mu & 0 \\ A_{5313} & \left(\mu - \frac{\kappa}{c_v}\right)u_1 & \left(\mu - \frac{\kappa}{c_v}\right)u_2 & \left(\frac{4}{3}\mu - \frac{\kappa}{c_v}\right)u_3 & \frac{\kappa}{c_v} \end{bmatrix}$$

with

$$A_{5313} = -\frac{1}{3}\mu u_3^2 - \mu\|\vec{u}\|^2 - \frac{\kappa}{c_v}(E - \|\vec{u}\|^2).$$

References

[1] D. Arnold, F. Brezzi, B. Cockburn, D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal. 39 (2002) 1749–1779.

[2] F. Bassi, A. Crivellini, S. Rebay, M. Savini, Discontinuous Galerkin solution of the Reynolds-averaged Navier–Stokes and k–ω turbulence model equations, Comput. Fluids 34 (4–5) (2004) 507–540.

[3] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, J. Comput. Phys. 131 (1997) 267–279.

[4] F. Bassi, S. Rebay, Numerical evaluation of two discontinuous Galerkin methods for the compressible Navier–Stokes equations, Int. J. Numer. Methods Fluids 40 (2002) 197–207.

[5] F. Bassi, S. Rebay, G. Mariotti, S. Pedinotti, M. Savini, A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flow, in: R. Decuypere, G. Dibelius (Eds.), Second European Conference on Turbomachinery, Fluid Dynamics and Thermodynamics, Technologisch Instituut, Antwerpen, 1997, pp. 99–108.

[6] P. Batten, N. Clarke, C. Lambert, D. Causon, On the choice of wavespeeds for the HLLC Riemann solver, SIAM J. Sci. Comput. 18 (6) (1997) 1553–1570.

[7] C.E. Baumann, J.T. Oden, A discontinuous *hp* finite element method for the Euler and Navier–Stokes equations, in: Tenth International Conference on Finite Elements in Fluids (Tucson, AZ, 1998), Int. J. Numer. Methods Fluids 31 (1) (1999) 79–95.

[8] O.J. Boelens, H. van der Ven, B. Oskam, A.A. Hassan, The boundary conforming discontinuous Galerkin finite element approach for rotorcraft simulations, J. Aircraft 39 (5) (2002) 776–785.

[9] F. Brezzi, G. Manzini, D. Marini, P. Pietra, A. Russo, Discontinuous Galerkin approximations for elliptic problems, Numer. Methods Part. Differen. Equat. 16 (4) (2000) 365–378.

[10] B. Cockburn, Discontinuous Galerkin methods for convection-dominated problems, in: T.J. Barth, H. Deconinck (Eds.), Lecture Notes in Computer Science and Engineering, vol. 9, Springer-Verlag, 1999.

[11] B. Cockburn, Discontinuous Galerkin methods, ZAMM Z. Angew. Math. Mech. 11 (2003) 731–754, 65–02.

[12] B. Cockburn, G.E. Karniadakis, C.-W. Shu, The development of discontinuous Galerkin methods, in: B. Cockburn, G.E. Karniadakis, C.-W. Shu (Eds.), Lecture Notes in Computer Science and Engineering, vol. 11, Springer-Verlag, 1999.

[13] B. Cockburn, C.-W. Shu, The local discontinuous Galerkin method for time-dependent method for convection–diffusion systems, SIAM J. Numer. Anal. 35 (1998) 2240–2463.
[14] B. Cockburn, C.-W. Shu, Runge–Kutta discontinuous Galerkin methods for convection-dominated problems, J. Sci. Comput. 16 (3) (2001) 173–261.
[15] V. Dolejší, On the discontinuous Galerkin method for the numerical solution of the Navier–Stokes equations, Int. J. Numer. Methods Fluids 45 (2004) 1083–1106.
[16] C. Farhat, P. Geuzaine, C. Grandmont, The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids, J. Comput. Phys. 174 (2) (2001) 669–694.
[17] R. Hartmann, P. Houston, Adaptive discontinuous Galerkin finite element methods with interior penalty for the compressible Navier–Stokes equations, in: M. Feistauer, V. Dolejší, P. Knobloch, K. Najzar (Eds.), Numerical Mathematics and Advanced Applications, ENUMATH 2003, Springer, 2004, pp. 410–419.
[18] C.M. Klaij, J.J.W. van der Vegt, H. van der Ven, Pseudo-time stepping methods for space–time discontinuous Galerkin discretizations of the compressible Navier–Stokes equations, J. Comput. Phys., submitted for publication, See also Technical Memorandum 1782 <http://www.math.utwente.nl/publications>, 2005.
[19] J. Kok, An Industrially applicable solver for compressible, turbulent flows, Ph.D. thesis, Delft University of Technology, Delft, The Netherlands, 1998.
[20] J.C. Kok, S.P. Spekreijse, Efficient and accurate implementation of the k–ω turbulence model in the NLR multi-block Navier–Stokes system, NLR TP-2000-144, in: Presented at ECCOMAS 2000, Barcelona, Spain, 11–14 September, 2000.
[21] M. Lesoinne, C. Farhat, Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations, Comput. Methods. Appl. Mech. Eng. 134 (1996) 71–90.
[22] G.A. Osswald, K.N. Ghia, U. Ghia, Simulation of dynamic stall phenomenon using unsteady Navier–Stokes equations, Comput. Phys. Commun. 65 (1991) 209–218.
[23] A.J. Riley, M.V. Lowson, Development of a three dimensional free shear layer, J. Fluid. Mech. 369 (1998) 49–89.
[24] J.J. Sudirham, J.J.W. van der Vegt, R.M.J. van Damme, Space–time discontinuous Galerkin method for advection–diffusion problems on time-dependent domains, Appl. Numer. Mathematics, in press.
[25] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics. A Practical Introduction, Springer-Verlag, 1997.
[26] J.J.W. van der Vegt, H. van der Ven, Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows. I. General formulation, J. Comput. Phys 182 (2002) 546–585.
[27] J.J.W. van der Vegt, H. van der Ven, Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows, in: 33rd Computational Fluid Dynamics Course – Novel Methods for Solving Convection Dominated Systems, VKI Lectures Series Monographs: Computational Fluid Dynamics, vol. 1, 2003.
[28] H. van der Ven, O.J. Boelens, A framework for aeroelastic simulations of trimmed rotor systems in forward flight, in: Proceedings of the 30th European Rotorcraft Forum, Marseille, France, September 14–16, 2004.
[29] H. van der Ven, 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. Methods Appl. Mech. Engrg. 191 (2002) 4747–4780.
[30] H. van der Ven, J.J.W. van der Vegt, E.G. Bouwman, Space–time discontinuous Galerkin finite element method for inviscid gas dynamicsComputational Fluid and Solid Mechanics 2003 (MIT Boston), vol. 1, Elsevier Science, Oxford, UK, 2003, pp. 1181–1184.
[31] M.R. Visbal, J.S. Shang, Investigation of the flow structure around a rapidly pitching airfoil, AIAA J. 27 (8) (1989) 1044–1051.