Journal of Computational Physics **182,** 546–585 (2002)
doi:10.1006/jcph.2002.7185

Space–Time Discontinuous Galerkin Finite Element Method with Dynamic Grid Motion for Inviscid Compressible Flows

I. General Formulation

J. J. W. van der Vegt*,[1] and H. van der Ven†

**University of Twente, Faculty of Mathematical Sciences, P.O. Box 217, 7500 AE, Enschede, The Netherlands; and †National Aerospace Laboratory NLR, P.O. Box 90502, 1006 BM Amsterdam, The Netherlands*
E-mail: j.j.w.vandervegt@math.utwente.nl and venvd@nlr.nl

Received January 1, 2002; revised June 27, 2002

A new space–time discontinuous Galerkin finite element method for the solution of the Euler equations of gas dynamics in time-dependent flow domains is presented. The discontinuous Galerkin discretization results in an efficient elementwise conservative upwind finite element method, which is particularly well suited for local mesh refinement. The upwind scheme uses a formulation of the HLLC flux applicable to moving meshes and several formulations for the stabilization operator to ensure that monotone solutions around discontinuities are investigated. The nonlinear equations of the space–time discretization are solved using a multigrid accelerated pseudo-time-integration technique with an optimized Runge–Kutta method. The linear stability of the pseudo-time-integration method is investigated for the linear advection equation. The numerical scheme is demonstrated with simulations of the flow field in a shock tube, a channel with a bump, and an oscillating NACA 0012 airfoil. These simulations show that using the data at the superconvergence points, the accuracy of the numerical discretization is $O(h^{5/2})$ in space for smooth subsonic flows, both on structured and on locally refined meshes, and that the space–time adaptation can significantly improve the accuracy and efficiency of the numerical method. © 2002 Elsevier Science (USA)

Key Words: discontinuous Galerkin finite element methods; local mesh refinement; dynamic grid motion; arbitrary Lagrangian Eulerian (ALE) technique; pseudo-time-integration methods; multigrid techniques; gas dynamics.

[1] Supported in part by a research grant from the Netherlands National Aerospace Laboratory.

0021-9991/02 $35.00

1. INTRODUCTION

This article discusses a new discontinuous Galerkin (DG) finite element method for the adaptive solution of the unsteady Euler equations of gas dynamics in three-dimensional time-dependent flow domains. The algorithm results in a second-order-accurate finite element discretization on deforming meshes and accuracy can be improved using local mesh refinement or h-type mesh adaptation. In the development of the numerical scheme the main objectives to be satisfied are the obtaining of a conservative discretization on deforming meshes, the accurate capturing of flow discontinuities using h-adaptation while maintaining accuracy on locally refined meshes, and the achieving of good computational efficiency on parallel computers. These requirements have been the main motivation for developing a discontinuous Galerkin finite element method. The main feature of discontinuous Galerkin methods is the use of basis functions which are discontinuous across element faces. This results in a finite element discretization with a very compact stencil, which can be combined well with h-adaptation. These properties are important for many problems and the main reason why discontinuous Galerkin methods currently are receiving significant attention.

Discontinuous Galerkin methods can be divided into two main classes, namely, discretizations with basis functions which are discontinuous either in space or in time. The first class of DG methods, in combination with a TVD Runge–Kutta time integration method, has been thoroughly investigated by Cockburn and co-workers. Detailed surveys can be found in [12, 13]. The second class of DG methods uses discontinuous basis functions in time and a streamline upwind Galerkin or Galerkin least-squares discretization in space. Both classes of discontinuous Galerkin methods are also extensively discussed in Barth [2] and Schwab [25].

The separation between space and time becomes cumbersome for time-dependent domain boundaries, which require the mesh to follow the boundary movement. We will therefore not separate space and time but consider the Euler equations directly in four dimensional space and use basis functions in the finite element discretization which are discontinuous across element faces, both in space and time. We refer to this technique as the space–time discontinuous Galerkin finite element method. The space–time DG method provides optimal efficiency for adapting and deforming the mesh while maintaining a conservative scheme which does not require interpolation of data after mesh refinement or deformation. The space–time DG method presented in this article is an extension of our research on a solution adaptive discontinuous Galerkin finite element method for steady three-dimensional inviscid compressible flows [31]. This article discusses the general formulation of the space–time DG method for the adaptive solution of the Euler equations in time-dependent flow domains. Important improvements in the computational efficiency are discussed in van der Ven and van der Vegt [34], where we present and analyze a new integration technique for the element face and volume integrals for discontinuous Galerkin discretizations. There we also demonstrate the maturity of the space–time DG discretization with three-dimensional aerodynamic applications, such as a deforming wing in transonic flow.

The combined use of space and time discontinuous basis functions in a discontinuous Galerkin method has been proposed by Jaffre *et al.* [19], who theoretically analyzed this technique for multidimensional scalar conservation laws on nondeforming meshes (see also Cockburn and Gremaud [8]). Until now, however, the use of space–time discontinuous basis functions in DG methods has not been fully explored for nonlinear hyperbolic systems of partial differential equations, such as the Euler equations of gas dynamics. An initial study was conducted by Lowrie *et al.* [22]. Their formulation results in a staggered space–time mesh, which is quite different from the DG discretization presented in this article, and does not easily extend to local mesh refinement, which is important for many applications.

In order to make the space–time DG method an accurate and efficient technique for the solution of the Euler equations of gas dynamics we had to deal with a number of issues. First, we will extensively discuss the weak formulation of the space–time discontinuous Galerkin finite element method using the arbitrary Lagrangian Eulerian (ALE) approach. This technique decouples the grid motion from the motion of the fluid particles and is widely used in fluid-structure interaction problems and ideally suited for deforming meshes. The discontinuous Galerkin discretization which we present automatically satisfies the geometric conservation law, which states that a uniform flow field should not be influenced by the grid motion, since the element face and volume integrals are calculated with sufficiently accurate quadrature rules. This problem was analyzed in detail by Lesoinne and Farhat [21] and is an essential condition for obtaining at least first-order accuracy in time, as was proven by Guillard and Farhat [17].

The space–time discontinuous Galerkin discretization results for each element in a coupled system of nonlinear equations. We will present and analyze a pseudo-time-integration method with multigrid acceleration which can efficiently solve these equations. In this technique the nonlinear equations of the DG discretization are augmented with a pseudo-time and marched to steady state in pseudo-time. The pseudo-time integration is significantly improved by optimizing the Runge–Kutta time integration method. The use of a multigrid technique for a DG discretization of hyperbolic partial differential equations is new and required a significant development effort. The proposed algorithm works well on locally refined meshes and maintains the local structure of a DG discretization, which allows a straightforward parallelization of the method.

Since the Euler equations of gas dynamics are hyperbolic and develop discontinuities in finite time it is important to ensure monotone solutions around discontinuities. In the TVD Runge–Kutta discontinuous Galerkin method this is accomplished by using a slope limiter (for a survey, see Cockburn [12]). In our earlier work we also used this limiter [31, 32], but the limiter in a DG method prevents convergence to steady state and also has a negative effect on the numerical accuracy. In this article we will discuss the use of a stabilization operator instead of a slope limiter to maintain monotone solutions. This technique significantly improved the accuracy and convergence to steady state of the pseudo-time integration.

The DG discretization combines well with local mesh refinement and this is considered one of its main benefits. The question of whether the DG discretization of the Euler equations maintains accuracy on nonsmooth meshes has, however, not been addressed until now. This has only been done for target functionals of the data, such as lift and drag on an airfoil (for a survey, see Giles and Süli [15]), but this does not provide a global error estimate for the solution. In this article we will investigate this issue with a number of mesh refinement studies on uniform and adapted meshes.

The outline of the article is as follows. After some preliminaries we discuss in Section 3 the definition of the space–time discontinuous Galerkin discretization. First, the ALE weak formulation is discussed, followed by a derivation of the nonlinear equations for the DG expansion coefficients. Next, a definition of the HLLC flux suitable for moving boundaries is given and the stabilization operator necessary to ensure monotone solutions around discontinuities is discussed. In Section 4 the multigrid accelerated pseudo-time-integration method for the solution of the nonlinear DG equations is presented and its stability is analyzed. The mesh adaptation is discussed in Section 5 and extensive results that demonstrate and validate the space–time DG method are presented in Section 6.

2. SPACE–TIME FORMULATION OF THE EULER EQUATIONS OF GAS DYNAMICS

We consider the Euler equations of gas dynamics in a time-dependent flow domain. Since the flow domain boundary is moving and deforming in time we do not make an explicit separation between the space and time variables and consider the Euler equations directly in $\mathbb{R}^4$. Let $\mathcal{E} \subset \mathbb{R}^4$ be an open domain. A point $x \in \mathbb{R}^4$ has coordinates $(x_1, \ldots, x_4)$, but we will also frequently use the notation $(\bar{x}, t) \in \mathbb{R}^4$, with $\bar{x} = (x_1, x_2, x_3) \in \mathbb{R}^3$ being the position vector at time t and $t = x_4$ representing time. The flow domain $\Omega(t)$ at time t is defined as $\Omega(t) := \{\bar{x} \in \mathbb{R}^3 \mid (\bar{x}, t) \in \mathcal{E}\}$, with t_0 and T the initial and final time of the evolution of the flow domain. The space–time domain boundary $\partial\mathcal{E}$ consists of the hypersurfaces $\Omega(t_0) := \{x \in \partial\mathcal{E} \mid x_4 = t_0\}$, $\Omega(T) := \{x \in \partial\mathcal{E} \mid x_4 = T\}$, and $\mathcal{Q} := \{x \in \partial\mathcal{E} \mid t_0 < x_4 < T\}$.

Let $\mathcal{F} : \mathbb{R}^5 \to \mathbb{R}^{5\times4}$ denote the flux tensor, which is defined as

$$\mathcal{F} = \begin{pmatrix} \rho u_1 & \rho u_2 & \rho u_3 & \rho \\ \rho u_1^2 + p & \rho u_1 u_2 & \rho u_1 u_3 & \rho u_1 \\ \rho u_1 u_2 & \rho u_2^2 + p & \rho u_2 u_3 & \rho u_2 \\ \rho u_1 u_3 & \rho u_2 u_3 & \rho u_3^2 + p & \rho u_3 \\ (\rho E + p)u_1 & (\rho E + p)u_2 & (\rho E + p)u_3 & \rho E \end{pmatrix},$$

with ρ, p, and E the density, pressure, and specific total energy, respectively, and u_i the velocity components in the Cartesian coordinate directions x_i, $i \in \{1, 2, 3\}$ of the velocity vector $u : \mathcal{E} \to \mathbb{R}^3$. Let the vector $U : \mathcal{E} \to \mathbb{R}^5$ denote the conservative flow variables with components:

$$U_i = \mathcal{F}_{i4}.$$

Then the Euler equations of gas dynamics are defined as

$$\operatorname{div} \mathcal{F}(U(x)) = 0, \quad x \in \mathcal{E}, \tag{1}$$

together with the initial and boundary conditions:

$$\begin{aligned} U(x) &= U_0(x), & x &\in \Omega(t_0), \\ U(x) &= \mathcal{B}(U, U_w), & x &\in \mathcal{Q}. \end{aligned}$$

Here $U_0 : \Omega(t_0) \to \mathbb{R}^5$ denotes the initial flow field, $\mathcal{B} : \mathbb{R}^5 \times \mathbb{R}^5 \to \mathbb{R}^5$ the boundary operator, and $U_w : \mathcal{Q} \to \mathbb{R}^5$ the prescribed boundary flow field data. The divergence of a second-order tensor is defined as $\operatorname{div} \mathcal{F} = \frac{\partial \mathcal{F}_{ij}}{\partial x_j}$, and the summation index is used on repeated indices in this article. The Euler equations are completed with the equation of state for a caloric perfect gas: $p = (\gamma - 1)\rho(E - \frac{1}{2}u_i u_i)$, with γ the ratio of specific heats.

3. SPACE–TIME DISCONTINUOUS GALERKIN DISCRETIZATION OF THE EULER EQUATIONS

3.1. Geometry Definition of Space–Time Elements

Consider a partitioning $t_0 < t_1 < \cdots < T$ of the time interval (t_0, T) and define the time interval I_n as $I_n = (t_n, t_{n+1})$. The space–time domain $\mathcal{E} \subset \mathbb{R}^4$ is split into a finite number of space–time slabs: $\{x \in \mathcal{E} \mid x_4 \in I_n\}$. The evolution of the flow domain during the time interval I_n is represented by the mapping Φ_t^n, which is defined as

$$\Phi_t^n : \Omega(t_n) \to \Omega(t) : \bar{x} \mapsto \Phi_t^n(\bar{x}), \quad t \in I_n. \tag{2}$$

The mapping Φ_t^n is assumed to be sufficiently smooth, orientation preserving, and invertible in each time interval I_n but can be different in different time intervals. This makes it possible to generate a new grid when elements become too severely distorted during the dynamic mesh movement. At the time level t_n we use hexahedral elements K to define the tessellation $\bar{\mathcal{T}}_h^n$; i.e.,

$$\bar{\mathcal{T}}_h^n := \left\{ K_j^n \,\middle|\, \bigcup_{j=1}^{N_n} \bar{K}_j^n = \bar{\Omega}_h(t_n) \quad \text{and} \quad K_j^n \cap K_{j'}^n = \emptyset \quad \text{if } j \neq j', 1 \leq j, j' \leq N_n \right\},$$

such that $\Omega_h(t_n) \to \Omega(t_n)$ as $h \to 0$, with h the radius of the smallest sphere completely containing each element $K \in \bar{\mathcal{T}}_h^n$, and N_n the total number of hexahedra in $\Omega_h(t_n)$. Each element $K^n \in \bar{\mathcal{T}}_h^n$ is related to the master element $\hat{K} = (-1, 1)^3$ through the mapping F_K^n; i.e.,

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

with $x_i(K^n) \in \mathbb{R}^3$, $1 \leq i \leq 8$, the spatial coordinates of the vertices of the hexahedron K^n at time t_n and $\chi_i(\bar{\xi})$ the standard trilinear finite element shape functions for hexahedra, with $\bar{\xi} = (\xi_1, \xi_2, \xi_3) \in \hat{K}$. The elements K^{n+1} are now obtained by moving the vertices of each hexahedron $K^n \in \bar{\mathcal{T}}_h^n$ with the mapping Φ_t^n to their new position at time $t = t_{n+1}$, and we can define the mapping:

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

The space–time elements are obtained by connecting the elements in $\Omega(t_n)$ and $\Omega(t_{n+1})$ by linear interpolation in time. This results in the parameterization of the space–time elements $\mathcal{K}^n$,

$$\begin{aligned} G_K^n : \hat{\mathcal{K}} \to \mathcal{K}^n : \xi \mapsto (\bar{x}, t) = \bigg(&\frac{1}{2}(1 - \xi_4)F_K^n(\bar{\xi}) + \frac{1}{2}(1 + \xi_4)F_K^{n+1}(\bar{\xi}), \\ &\frac{1}{2}(t_n + t_{n+1}) + \frac{1}{2}(t_{n+1} - t_n)\xi_4 \bigg), \end{aligned} \tag{3}$$

with $\xi \in \hat{\mathcal{K}}$ the computational coordinates in the master element $\hat{\mathcal{K}}$, which is defined as: $\hat{\mathcal{K}} = (-1, 1)^4$. The space–time tessellation is now defined as

$$\mathcal{T}_h^n := \left\{\mathcal{K} = G_K^n(\hat{\mathcal{K}}) \,\middle|\, K \in \bar{\mathcal{T}}_h^n\right\}.$$

We will also frequently use the notation $K(t)$ for the element K at time t, which is defined as $K(t) = \{\bar{x} \in \mathbb{R}^3 | (\bar{x}, t) \in \mathcal{K}\}$. The space–time element $\mathcal{K}^n$ is bounded by the hypersurfaces $K(t_n^+) = \lim_{\epsilon\downarrow 0} K(t_n + \epsilon)$, $K(t_{n+1}^-) = \lim_{\epsilon\downarrow 0} K(t_{n+1} - \epsilon)$, and $\mathcal{Q}^n = \partial\mathcal{K}^n \backslash (K(t_n^+) \cup K(t_{n+1}^-))$. This notation is used to indicate that the mesh can change discontinuously at the time levels t_n and t_{n+1}.

The boundary faces of $\mathcal{K}^n$ can also be represented using the mapping (3). For $1 \leq m \leq 8$ define the eight faces $\mathcal{S}_m$ of the space–time element $\mathcal{K}$, with $\partial\mathcal{K} = \cup_{m=1}^8 \mathcal{S}_m$, by

$$\begin{aligned} \mathcal{S}_{2m-1} &= \{G_K(\xi) \,|\, \xi \in \hat{\mathcal{K}}, \xi_m = -1\}, \\ \mathcal{S}_{2m} &= \{G_K(\xi) \,|\, \xi \in \hat{\mathcal{K}}, \xi_m = 1\}, \quad 1 \leq m \leq 4. \end{aligned} \tag{4}$$

Note that $\mathcal{Q}^n = \cup_{m=1}^6 \mathcal{S}_m$, $K(t_n^+) = \mathcal{S}_7$, and $K(t_{n+1}^-) = \mathcal{S}_8$. The reader is referred to Fig. 1 for a two-dimensional illustration of the elements and mappings.

Remark. The tessellation $\mathcal{T}_h^n$ does not impose a limitation on the number of elements which can connect to a face of an element. This is important because during the simulations the computational mesh will be adapted by subdividing elements in space and/or time in regions where more mesh resolution is required.

Remark. Since we use a trilinear representation of the elements in space, this implies that we use a bilinear representation of the geometry at slip flow boundaries. In an interesting article, Bassi and Rebay [1] concluded that a higher order representation of a slip flow boundary is mandatory in order to avoid strong numerical boundary layers and to obtain convergence. Recently, van der Vegt and van der Ven [33] showed that under grid refinement the numerical boundary layer diminishes for hexahedral-type elements. Since local mesh refinement already is an integral part of our algorithm, we use this technique to remove the numerical boundary layer at slip flow boundaries and it is not necessary to use a higher-order-accurate boundary representation.

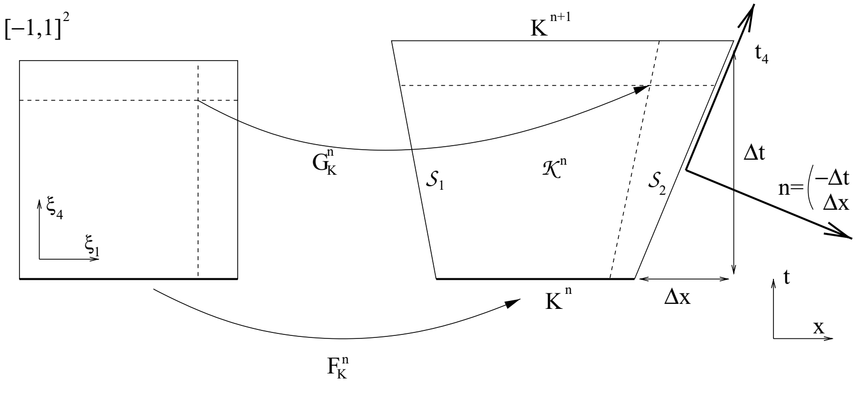


FIG. 1. Illustration of the geometry of two-dimensional space–time elements in both computational and physical space. Notations in the text.

3.2. Space–Time Discontinuous Galerkin Finite Element Approximation

3.2.1. *Weak Formulation of the Euler Equations*

In order to ensure that the different forms of the weak formulation of the Euler equations of gas dynamics, which are discussed in this section, are well defined we introduce the broken space $V(\mathcal{T}_h^n)$; i.e.,

$$\begin{aligned} V\left(\mathcal{T}_h^n\right) := \Big\{U : \mathcal{T}_h^n \to \mathbb{R}^5 \;\Big|\; & (\operatorname{grad} U^1)^T : \mathcal{F}(U^2)|_{\mathcal{K}_j^n} \in L^1\left(\mathcal{K}_j^n\right), \\ & ((\operatorname{grad} U^2)^T \cdot \mathfrak{D}(U^1)) : \operatorname{grad} U^1|_{\mathcal{K}_j^n} \in L^1\left(\mathcal{K}_j^n\right), \\ & \gamma^-(U^1) \cdot \left(n_{\mathcal{K}}^T \mathcal{F}(\gamma^-(U^2)) + n_{\mathcal{K}}^T \mathcal{F}(\gamma^+(U^3))\right) \in L^1\left(\partial\mathcal{K}_j^n\right), \\ & \forall (U^1, U^2, U^3) \in V\left(\mathcal{T}_h^n\right), \forall \mathcal{K}_j^n \in \mathcal{T}_h^n\Big\}, \end{aligned}$$

with L^1 the space of Lebesgue integrable functions, $\gamma^{\pm}(U) = \lim_{\epsilon\downarrow 0} U(x \pm \epsilon n_{\mathcal{K}})$ the traces of U at $\partial\mathcal{K}$, $n_{\mathcal{K}} \in R^4$ the unit outward normal vector at $\partial\mathcal{K}$, $\mathfrak{D} : \mathbb{R}^5 \to \mathbb{R}^{4\times 4}$ the artificial viscosity matrix, and superscript T the transposition of a vector. We will also frequently use the notation $U^{\pm}$ to denote $\gamma^{\pm}(U)$. The gradient operator grad : $\mathbb{R}^5 \to \mathbb{R}^{4\times 5}$ is defined as $(\operatorname{grad} U)_{ij} = \frac{\partial U_j}{\partial x_i}$ and the symbol : represents the dyadic product of two second-order tensors and is defined for $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{n\times m}$ as $\mathcal{A} : \mathcal{B} = \mathcal{A}_{ij}\mathcal{B}_{ij}$.

The discontinuous Galerkin finite element discretization is obtained by approximating the test and trial functions in each element $\mathcal{K} \in \mathcal{T}_h^n$ with polynomial expansions which are discontinuous across element faces, both in space and time. First, in the master element $\hat{\mathcal{K}}$ the basis functions $\hat{\phi}_m : \hat{\mathcal{K}} \to \mathbb{R}$ are defined which are linear in computational space:

$$\begin{aligned} \hat{\phi}_m(\xi) &= 1, \quad & m &= 0, \\ &= \xi_m, \quad & m &= 1, \ldots, 4. \end{aligned}$$

Next, the basis functions $\phi_m : \mathcal{K} \to \mathbb{R}$ are constructed through the parameterization G_K:

$$\phi_m = \hat{\phi}_m \circ G_K^{-1}, \quad m = 0, \ldots, 4.$$

We also introduce the basis functions $\psi_m : \mathcal{K} \to \mathbb{R}$, which are defined as

$$\begin{aligned} \psi_m(\bar{x}, t) &= 1, & m &= 0, \\ &= \phi_m(\bar{x}, t) - \frac{1}{|K_j(t_{n+1}^-)|} \int_{K_j(t_{n+1}^-)} \phi_m(\bar{x}, t_{n+1}^-)\, dK, & m &= 1, \ldots, 4, \end{aligned} \tag{5}$$

since this will result in a splitting of the test and trial functions into an element mean at time t_{n+1} and a fluctuating part. This property will be beneficial in the definition of the stabilization operator and the multigrid convergence acceleration, discussed in Sections 3.5 and 4.

The finite element space $V_h^1(\mathcal{T}_h^n)$ is now defined as

$$V_h^1\left(\mathcal{T}_h^n\right) := \{U_h \mid U_h|_{\mathcal{K}} \in (P^1(\mathcal{K}))^5\} \subset V\left(\mathcal{T}_h^n\right),$$

with the polynomial space $P^1(\mathcal{K}) = \operatorname{span}\{\psi_m, m = 0, \ldots, 4\}$. The trial functions $U_h : \mathcal{T}_h^n \to \mathbb{R}^5$ are defined in each element $\mathcal{K} \in \mathcal{T}_h^n$ as

$$U_h(\bar{x}, t)|_{\mathcal{K}} \equiv \mathcal{P}(U(\bar{x}, t)|_{\mathcal{K}}) = \sum_{m=0}^{4} \hat{U}_m(\mathcal{K})\psi_m(\bar{x}, t), \tag{6}$$

with $\mathcal{P} : \mathbb{R}^5 \to V_h^1(\mathcal{T}_h^n)$ the projection operator on the space V_h^1 and $\hat{U}_m \in \mathbb{R}^5$ the expansion coefficients. The test functions $W_h : \mathcal{T}_h^n \to \mathbb{R}^5$ are defined analogously, only with $\hat{U}_m$ replaced by $\hat{W}_m$. The weak formulation for the Euler equations of an inviscid compressible gas can now be formulated as follows.

Find a $U_h \in V_h^1(\mathcal{T}_h^n)$ that for all $W_h \in V_h^1(\mathcal{T}_h^n)$ satisfies the variational equation

$$\sum_{n=0}^{N_T} \sum_{j=1}^{N_n} \left\{ \int_{\mathcal{K}_j^n} W_h \cdot \operatorname{div} \mathcal{F}(U_h)\, d\mathcal{K} + \int_{\mathcal{K}_j^n} ((\operatorname{grad} W_h)^T \cdot \mathfrak{D}(U_h)) : \operatorname{grad} U_h \, d\mathcal{K} \right\} = 0, \quad (7)$$

where $N_T + 1$ is the total number of space–time slabs and N_n the number of elements in the tessellation $\mathcal{T}_h^n$. The second contribution in (7) is the stabilization operator and it is added to the weak formulation to prevent numerical oscillations around discontinuities and in regions with sharp gradients (for more details see Section 3.5).

3.2.2. *Transformation of the Space–Time Weak Formulation into ALE Form*

The weak formulation (7) can be transformed into an integrated-by-parts form using Gauss' theorem. This has as the main benefit that it does not result in loss of conservation under inexact quadrature (see, e.g., Hansbo [18]). This approach is, for instance, followed by Shakib *et al.* [26]. It is, however, possible to establish a relation between the arbitrary Lagrangian Eulerian (ALE) formulation, commonly used on moving and deforming meshes, and the space–time approach. This can be done either directly for the partial differential equations, as presented by Masud and Hughes [23], or for the weak formulation using Stokes' theorem (see Bottasso [6]). In this section we will establish the relation between the space–time and ALE formulation in a way which does not require the use of differential forms and which gives more insight into the origin of the various contributions.

If we introduce

$$W_h \cdot \operatorname{div} \mathcal{F}(U_h) = \operatorname{div}\big(W_h^T \mathcal{F}(U_h)\big) - (\operatorname{grad} W_h)^T : \mathcal{F}(U_h) \quad (8)$$

into the weak formulation (7) and apply Gauss' theorem to the contribution resulting from the first term on the right-hand side of (8), then we obtain

$$\int_{\mathcal{K}_j^n} \operatorname{div}\big(W_h^T \mathcal{F}(U_h)\big)\, d\mathcal{K} = \int_{\partial\mathcal{K}_j^n} n_{\mathcal{K}} \cdot ((W_h^-)^T \mathcal{F}(U_h^-))\, d(\partial\mathcal{K}), \quad (9)$$

where $n_{\mathcal{K}}$ is the unit outward normal vector at the boundary $\partial\mathcal{K}^n$. The ALE formulation can now be obtained by calculating the normal vector $n_{\mathcal{K}}$.

Given the parameterization $(\bar{x}, t) = G_K^n(\xi)$ for the space–time element, the normal vector $n_{\mathcal{K}}$ at the boundary surface components $\mathcal{S}_{2i-1}$ and $\mathcal{S}_{2i}$, $1 \le i \le 4$, defined in (4), is orthogonal to the tangential vectors t_{i_1}, t_{i_2}, and t_{i_3}, with the indices $\{i_1, i_2, i_3\} \subset \{1, \dots, 4\}$ complementary to the index $\{i\}$. The tangential vectors are defined as $t_j = \frac{\partial G_K^n}{\partial \xi_j}$ and are equal to (cf. (3))

$$t_j = \begin{pmatrix} \frac{1}{2}(1 - \xi_4)\frac{\partial F_K^n(\bar{\xi})}{\partial \xi_j} + \frac{1}{2}(1 + \xi_4)\frac{\partial F_K^{n+1}(\bar{\xi})}{\partial \xi_j} \\ 0 \end{pmatrix}, \quad j = 1, 2, 3,$$

$$t_4 = \frac{1}{2}\begin{pmatrix} F_K^{n+1}(\bar{\xi}) - F_K^n(\bar{\xi}) \\ t_{n+1} - t_n \end{pmatrix} = \frac{1}{2}\begin{pmatrix} \Delta\bar{x} \\ \Delta t \end{pmatrix}$$

(also see Fig. 1). The normal vectors at $\mathcal{S}_7$ and $\mathcal{S}_8$ are simply $(0, 0, 0, -1)$ and $(0, 0, 0, 1)$, respectively; hence the boundary integrals over the surfaces $\mathcal{S}_7$ and $\mathcal{S}_8$ are equal to

$$\sum_{m=7}^{8} \int_{\mathcal{S}_m} n_{\mathcal{K}} \cdot ((W_h^-)^T \mathcal{F}(U_h^-))\, d\mathcal{S} = \int_{K_j(t_{n+1}^-)} W_h^- \cdot U_h^-\, d\mathcal{S} - \int_{K_j(t_n^+)} W_h^- \cdot U_h^-\, d\mathcal{S}, \quad (10)$$

where we used the relations $U_h^- = \mathcal{F}_{i4}(U_h^-)$, $\mathcal{S}_7 = K_j(t_n^+)$, and $\mathcal{S}_8 = K_j(t_{n+1}^-)$.

For the remaining boundary terms remember that for each $\xi_4 \in (-1, 1)$, the element $K(t)$, such that $(K(t), \xi_4) = G_K^n(\bar{\xi}, \xi_4)$, is the space element defined by the interpolated vertices of the elements $K(t_n^+)$ and $K(t_{n+1}^-)$. Let $\bar{n}_{\mathcal{K}}^i(\bar{x}, t) \in \mathbb{R}^3$ $(1 \le i \le 6)$ be the space part of the normal vector at the boundary parts $\mathcal{S}_i \subset \mathcal{Q}_j^n$. By definition, $\bar{n}_{\mathcal{K}}^{2m-1}$ and $\bar{n}_{\mathcal{K}}^{2m}$ $(1 \le m \le 3)$ are perpendicular to the tangential vectors $t_{i_k} = \frac{1}{2}(1 - \xi_4)\frac{\partial F_K^n}{\partial \xi_{i_k}} + \frac{1}{2}(1 + \xi_4)\frac{\partial F_K^{n+1}}{\partial \xi_{i_k}}$, with $k = 1$ or 2, such that $\{i_1, i_2, m\} = \{1, 2, 3\}$. Hence, the vectors $(\bar{n}_{\mathcal{K}}^i, \alpha) \in \mathbb{R}^4$ are orthogonal to the tangential vectors t_{i_k}, if and only if the conditions

$$\frac{1}{2}\Delta\bar{x} \cdot \bar{n}_{\mathcal{K}}^i + \frac{1}{2}\alpha\,\Delta t = 0$$

are satisfied. The space–time normal vector $n_{\mathcal{K}}^i$ at $\mathcal{S}_i$ $(1 \le i \le 6)$ therefore is equal to

$$n_{\mathcal{K}}^i = \left(\bar{n}_{\mathcal{K}}^i, -v \cdot \bar{n}_{\mathcal{K}}^i\right),$$

with the grid velocity $v \in \mathbb{R}^3$ given by the relation $v = \Delta\bar{x}/\Delta t$. Since the space–time normal vector $n_{\mathcal{K}}$ has length one, the space normal vector $\bar{n}_{\mathcal{K}}$ has a length $|\bar{n}_{\mathcal{K}}| = 1/\sqrt{1 + v \cdot v}$. The boundary flux integral over $\mathcal{S}_i$ $(1 \le i \le 6)$ is now equal to

$$\sum_{i=1}^{6} \int_{\mathcal{S}_i} n_{\mathcal{K}} \cdot ((W_h^-)^T \mathcal{F}(U_h^-))\, d\mathcal{S} = \int_{\mathcal{Q}_j^n} [\bar{n}_{\mathcal{K}} \cdot ((W_h^-)^T \bar{\mathcal{F}}(U_h^-)) - \bar{n}_{\mathcal{K}} \cdot v(W_h^- \cdot U_h^-)]\, d\mathcal{Q}, \quad (11)$$

where the flux tensor $\bar{\mathcal{F}} : \mathbb{R}^5 \to \mathbb{R}^{5\times 3}$ has components $\bar{\mathcal{F}}_{ij} = \mathcal{F}_{ij}$ with $1 \le j \le 3$. If we replace the right-hand side of (9) with the sum of (10) and (11), using $\partial\mathcal{K}^n = \cup_{i=1}^8 \mathcal{S}_i$, and introduce this relation into (7), we obtain the weak formulation for the Euler equations of gas dynamics in ALE form.

Find a $U_h \in V_h^1(\mathcal{T}_h^n)$ such that for all $W_h \in V_h^1(\mathcal{T}_h^n)$, the following variational equation is satisfied:

$$\begin{aligned}\sum_{n=0}^{N_T}\sum_{j=1}^{N_n} \Bigg\{ &- \int_{\mathcal{K}_j^n} (\operatorname{grad} W_h)^T : \mathcal{F}(U_h)\, d\mathcal{K} + \int_{K_j(t_{n+1}^-)} W_h^- \cdot U_h^- dK \\ &- \int_{K_j(t_n^+)} W_h^- \cdot U_h^-\, dK + \int_{\mathcal{Q}_j^n} W_h^- \cdot (\bar{\mathcal{F}}(U_h^-)\bar{n}_{\mathcal{K}} - \bar{n}_{\mathcal{K}} \cdot v U_h^-)\, d\mathcal{Q} \\ &+ \int_{\mathcal{K}_j^n} ((\operatorname{grad} W_h)^T \cdot \mathfrak{D}(U_h)) : \operatorname{grad} U_h\, d\mathcal{K} \Bigg\} = 0. \end{aligned} \quad (12)$$

3.2.3. *Introduction of Numerical Flux*

In the summation over the space–time elements, the integrals over the internal faces of $\mathcal{Q}^n$, $K(t_n)$, and $K(t_{n+1})$ in the weak formulation (12) are counted twice, since two elements are connected to each side of the faces. (In case of mesh refinement this applies to subsets of these faces.) This results in a multivalued flux tensor at internal faces, since in general $U_h^- \neq U_h^+$ in the discontinuous Galerkin discretization, and this requires special care. If we use the fact that the normal flux through the boundary faces must be continuous, almost everywhere, to ensure conservation, then we obtain the relations

$$\begin{aligned}\int_{K_j(t_n^+)\cup K_j(t_{n+1}^-)} W_h^- \cdot U_h^- \, dK &= \int_{K_j(t_n^+)\cup K_j(t_{n+1}^-)} W_h^- \cdot U_h^+ \, dK, \quad \forall W_h \in V_h^1\left(\mathcal{T}_h^n\right),\\ \int_{\mathcal{Q}_j^n} W_h^- \cdot (\tilde{\mathcal{F}}(U_h^-)\bar{n}_{\mathcal{K}})\, d\mathcal{Q} &= \int_{\mathcal{Q}_j^n} W_h^- \cdot (\tilde{\mathcal{F}}(U_h^+)\bar{n}_{\mathcal{K}})\, d\mathcal{Q}, \qquad \forall W_h \in V_h^1\left(\mathcal{T}_h^n\right),\end{aligned} \tag{13}$$

with U_h^+ the trace of U_h at $\partial\mathcal{K}_j^n$ of elements connected to $\mathcal{K}_j^n$. The generalized flux tensor $\tilde{\mathcal{F}} : \mathbb{R}^5 \to \mathbb{R}^{5\times 3}$ is defined as

$$\tilde{\mathcal{F}}(U) = \bar{\mathcal{F}}(U) - v \otimes U,$$

where $v \otimes U = v_i U_j$. The integrals over internal faces $K_j(t_n^+)$ then transform into

$$\sum_{j=1}^{N_n} \int_{K_j(t_n^+)} W_h^- \cdot U_h^- \, dK = \sum_{j=1}^{N_n} \int_{K_j(t_n^+)} W_h^- \cdot \frac{1}{2}(U_h^- + U_h^+)\, dK,$$

with a similar relation for $K_j(t_{n+1}^-)$. The multivalued time flux is now replaced with a numerical flux H_T, which, in order to ensure the causality of the time flux, is defined as

$$\begin{aligned} H_T(U_h^-, U_h^+) &= U_h^+, \quad \text{at } K_j(t_n^+),\\ &= U_h^-, \quad \text{at } K_j(t_{n+1}^-).\end{aligned}$$

The numerical flux H_T can also be used at the boundary faces $K_j(t_0^+)$, where the external trace is provided by the initial condition at $t = t_0$. The numerical flux H_T makes it possible to drop the summation over the space–time slabs in the weak formulation (12), since each space–time slab only depends on the previous space–time slab. The introduction of the time flux is an alternative to the weak coupling between space–time slabs generally used in time-discontinuous Galerkin methods and results in a uniform treatment of the space–time flux in the DG discretization. Using (14), the integrals over $\mathcal{Q}_j^n$ in (12) can be transformed into

$$\sum_{j=1}^{N_n} \int_{\mathcal{Q}_j^n} W_h^- \cdot (\tilde{\mathcal{F}}(U_h^-)\bar{n}_{\mathcal{K}})\, d\mathcal{Q} = \sum_{j=1}^{N_n} \int_{\mathcal{Q}_j^n} W_h^- \cdot \frac{1}{2}(\tilde{\mathcal{F}}(U_h^-)\bar{n}_{\mathcal{K}} + \tilde{\mathcal{F}}(U_h^+)\bar{n}_{\mathcal{K}})\, d\mathcal{Q}. \tag{14}$$

The representation of the flux in (14) as the average between the left and right states at the element face results in a central discretization which suffers from numerical oscillations around discontinuities. Monotone solutions are obtained by adding the stabilization operator and introducing a Godunov-type upwind flux. The use of an upwind flux fits very well into a discontinuous Galerkin discretization, since the states U_h^- and U_h^+ can be considered the left and right states in a Riemann problem. We replace therefore the flux $\frac{1}{2}(\tilde{\mathcal{F}}(U_h^-(\bar{x},t))\bar{n}_{\mathcal{K}} + \tilde{\mathcal{F}}(U_h^+(\bar{x},t))\bar{n}_{\mathcal{K}})$ at the element faces $\mathcal{Q}_j^n$ in the time interval $[t, t+\Delta t)$ with a monotone upwind flux $H(U_h^-, U_h^+)$, which is consistent, $H(U_h, U_h) = \tilde{\mathcal{F}}(U_h)\bar{n}_{\mathcal{K}}$, and conservative, $H(U_h^-, U_h^+) = -H(U_h^+, U_h^-)$. At external boundary faces we apply the same procedure, but at these faces the external state U_h^+ is controlled by the boundary operator: $U_h^+ = \mathcal{B}(U_h^-, U_w)$.

Any of the well-known (approximate) Riemann solvers, such as those from Godunov, Roe, Lax–Friedrichs, or Osher (for a survey, see Toro [30]), can be used as upwind numerical flux. In earlier work [31], we used the Osher flux because of its good accuracy and nice mathematical foundation, but the Osher flux is computationally expensive and is replaced with the HLLC flux. The HLLC flux is introduced by Toro *et al.* [29] and further analyzed by Batten *et al.* [3, 5]. The HLLC flux provides solutions of at least the same quality as the Osher flux but at less than one quarter of the computational cost. The definition of the HLLC flux for moving interfaces is provided in Section 3.4. An important benefit of using an upwind numerical flux is that this ensures nearly monotone solutions without a stabilization operator. A relatively simple stabilization operator in comparison with, for instance, the one used by Shakib *et al.* [26] for the Galerkin least-squares finite element method then is sufficient to obtain monotone solutions. The weak formulation for the space–time discontinuous Galerkin finite element discretization of the Euler equations of gas dynamics now is equal to the following:

Find a $U_h \in V_h^1(\mathcal{T}_h^n)$ such that for all $W_h \in V_h^1(\mathcal{T}_h^n)$, the following variational equation is satisfied:

$$\sum_{j=1}^{N_n} \Bigg\{ -\int_{\mathcal{K}_j^n} (\text{grad}\, W_h)^T : \mathcal{F}(U_h)\, d\mathcal{K} + \int_{K_j(t_{n+1}^-)} W_h^- \cdot U_h^-\, dK \\ - \int_{K_j(t_n^+)} W_h^- \cdot U_h^+\, dK + \int_{\mathcal{Q}_j^n} W_h^- \cdot H(U_h^-, U_h^+)\, d\mathcal{Q} \\ + \int_{\mathcal{K}_j^n} ((\text{grad}\, W_h)^T \cdot \mathfrak{D}(U_h)) : \text{grad}\, U_h\, d\mathcal{K} \Bigg\} = 0. \tag{15}$$

3.3. Equations for the Flow Field Expansion Coefficients

An important element in the numerical discretization is the splitting of the test and trial functions into an element mean $\bar{U}_h : \mathcal{T}_h^n \to \mathbb{R}^5$ at the time level t_{n+1} and a fluctuating part $\tilde{U}_h : \mathcal{T}_h^n \to \mathbb{R}^5$; i.e.,

$$U_h(\bar{x}, t) = \bar{U}_h(K_j(t_{n+1}^-)) + \tilde{U}_h(\bar{x}, t), \quad \forall (\bar{x}, t) \in \mathcal{K}_j^n, \tag{16}$$

with

$$\bar{U}_h(K_j(t_{n+1}^-)) = \hat{U}_0, \tag{17}$$

$$\int_{K_j(t_{n+1}^-)} \tilde{U}_h(\bar{x}, t)\, dK = 0. \tag{18}$$

The flow field can now be represented as

$$U_h(\bar{x},t) = \bar{U}_h(K_j(t_{n+1}^-)) + \sum_{m=1}^{4} \hat{U}_m\left(\mathcal{K}_j^n\right)\psi_m(\bar{x},t), \quad \forall(\bar{x},t)\in\mathcal{K}_j^n.$$

This splitting is a direct consequence of the fact that the basis functions $\psi_m(\bar{x},t)$ are constructed such that

$$\int_{K_j(t_{n+1}^-)} \psi_m(\bar{x},t_{n+1}^-)\,dK = 0, \quad m\geq 1, \tag{19}$$

and it has several advantages. In the first place, the structure of the space–time discontinuous Galerkin discretization becomes more clear, because the equations for the element mean are identical to a finite volume discretization. A second advantage of the splitting is that it makes it easier to define the stabilization operator and the multigrid convergence acceleration procedure. The stabilization operator does not act on the element mean, but only on the fluctuating part. Any adjustment to the element fluctuations due to the stabilization operator will therefore not affect the element mean and will preserve a conservative scheme. The multigrid procedure also benefits from this splitting, since it only uses the equations for the element mean at the coarse grid levels. This results in a significant simplification of the multigrid algorithm while maintaining good multigrid performance.

If we introduce the polynomial expansions (6), for U_h and W_h into the weak formulation of the Euler equations (15) and use (16)–(19) and the fact that the coefficients $\hat{W}$ are arbitrary, then the following set of equations for the element mean $\bar{U}_i(K_j(t_{n+1}^-))$, $1\leq i\leq 5$, is obtained:

$$|K_j(t_{n+1}^-)|\bar{U}_i(K_j(t_{n+1}^-)) - \int_{K_j(t_n^+)} U_{h,i}(\bar{x},t_n^-)\,dK + \int_{\mathcal{Q}_j^n} H_i(U_h^-,U_h^+)\,d\mathcal{Q} = 0. \tag{20}$$

The coefficients for the fluctuating part of the flow field $\hat{U}_{mi}(\mathcal{K}_j^n)$ with $m=1,\ldots,4$, are equal to

$$\begin{aligned}
&\sum_{m=1}^{4}\hat{U}_{mi}\left(\mathcal{K}_j^n\right)\left(-\int_{\mathcal{K}_j^n}\frac{\partial\psi_l(\bar{x},t)}{\partial t}\psi_m(\bar{x},t)\,d\mathcal{K} + \int_{K_j(t_{n+1}^-)}\psi_l(\bar{x},t_{n+1}^-)\psi_m(\bar{x},t_{n+1}^-)\,dK\right)\\
&\quad-\int_{K_j(t_n^+)}U_{h,i}(\bar{x},t_n^-)\psi_l(\bar{x},t_n^+)\,dK - \bar{U}_i(K_j(t_{n+1}^-))\int_{\mathcal{K}_j^n}\frac{\partial\psi_l(\bar{x},t)}{\partial t}\,d\mathcal{K}\\
&\quad+\int_{\mathcal{Q}_j^n}\psi_l(\bar{x},t)H_i(U_h^-,U_h^+)\,d\mathcal{Q} - \int_{\mathcal{K}_j^n}\frac{\partial\psi_l(\bar{x},t)}{\partial x_k}\bar{\mathcal{F}}_{ik}(U_h)\,d\mathcal{K}\\
&\quad+\sum_{m=1}^{4}\hat{U}_{mi}\left(\mathcal{K}_j^n\right)\int_{\mathcal{K}_j^n}\frac{\partial\psi_l(\bar{x},t)}{\partial x_k}\mathfrak{D}_{kp}(U_h)\frac{\partial\psi_m(\bar{x},t)}{\partial x_p}\,d\mathcal{K}=0, \quad l=1,\ldots,4;\ i=1,\ldots,5.
\end{aligned} \tag{21}$$

The computational mesh can be discontinuous at the interface between two space–time slabs. This implies that more than one element in $\mathcal{T}_h^{n-1}$ can connect to the element $\mathcal{K}_j^n\in\mathcal{T}_h^n$. In that case the polynomial representation of $U_h(\bar{x},t_n^-)$ in the various elements in $\mathcal{T}_h^{n-1}$ which connect to the element $\mathcal{K}_j^n$ must be used in the evaluation of the integrals $\int_{K_j(t_n^+)} U_h(\bar{x}, t_n^-)\psi_l(\bar{x}, t_n^+)\, dK$. This is discussed in Sections 5.1 and 5.2. The different contributions in (21) are evaluated separately. Define the geometric coefficients A^1, A^2, $\mathcal{A} \in \mathbb{R}^{5\times 5}$ as

$$\begin{aligned}
A_{lm}^1 &= \int_{\mathcal{K}_j^n} \frac{\partial \psi_l(\bar{x}, t)}{\partial t} \psi_m(\bar{x}, t)\, d\mathcal{K},\\
A_{lm}^2 &= \int_{K_j(t_{n+1}^-)} \psi_l(\bar{x}, t_{n+1}^-)\psi_m(\bar{x}, t_{n+1}^-)\, dK,\\
\mathcal{A}_{lm} &= -A_{lm}^1 + A_{lm}^2,
\end{aligned}$$

and the coefficients $B \in \mathbb{R}^{5\times 5}$, which couple the space–time slabs, as

$$B_{il}\big(U_h^+|_{K_j(t_n^+)}\big) = \int_{K_j(t_n^+)} U_{h,i}(\bar{x}, t_n^-)\psi_l(\bar{x}, t_n^+)\, dK. \tag{22}$$

The element face and volume flux contributions $\mathcal{R}^1 \in \mathbb{R}^{5\times 5}$, $\mathcal{R}^2 \in \mathbb{R}^{5\times 4}$ are defined as

$$\mathcal{R}_{il}^1\big(U_h^-|_{\mathcal{Q}_j^n}, U_h^+|_{\mathcal{Q}_j^n}\big) = \int_{\mathcal{Q}_j^n} \psi_l(\bar{x}, t) H_i(U_h^-, U_h^+)\, d\mathcal{Q}, \tag{23}$$

$$\mathcal{R}_{il}^2\big(U_h|_{\mathcal{K}_j^n}\big) = \int_{\mathcal{K}_j^n} \frac{\partial \psi_l(\bar{x}, t)}{\partial x_k} \bar{\mathcal{F}}_{ik}(U_h)\, d\mathcal{K}, \tag{24}$$

and the integrals of the stabilization operator $\mathcal{D} \in \mathbb{R}^{4\times 4}$ are denoted as

$$\mathcal{D}_{lm}\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big) = \int_{\mathcal{K}_j^n} \frac{\partial \psi_l(\bar{x}, t)}{\partial x_k} \mathfrak{D}_{kp}\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big) \frac{\partial \psi_m(\bar{x}, t)}{\partial x_p}\, d\mathcal{K}, \tag{25}$$

with $U_h^*|_{\mathcal{K}_j^n}$ the solution in the elements $\mathcal{K} \in \mathcal{T}_h^n$, which connect to the element $\mathcal{K}_j^n$. The evaluation of the flux and stabilization operator integrals is discussed in Sections 3.4 and 3.5, respectively. The system of nonlinear equations (21) for the expansion coefficients $\hat{U}_{mi}(\mathcal{K}_j^n)$ can be expressed as

$$\mathcal{L}(\hat{U}^n, \hat{U}^{n-1}) = 0, \tag{26}$$

with $\hat{U}^n = \hat{U}_{mi}(\mathcal{K})$, $\mathcal{K} \in \mathcal{T}_h^n$, $0 \le m \le 4$, $1 \le i \le 5$, and $\mathcal{L} : \mathbb{R}^{5\times 5} \times \mathbb{R}^{5\times 5} \to \mathbb{R}^{5\times 5}$ having components in each space–time element:

$$\mathcal{L}_{i0} = |K_j(t_{n+1}^-)| \bar{U}_i(K_j(t_{n+1}^-)) - B_{i0}\big(U_h^+|_{K_j(t_n^+)}\big) + \mathcal{R}_{i0}^1\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big), \tag{27}$$

$$\begin{aligned}
\mathcal{L}_{il} &= \sum_{m=1}^{4} \big(\mathcal{A}_{lm}\big(\mathcal{K}_j^n\big) + \mathcal{D}_{lm}\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big)\big)\hat{U}_{mi}\big(\mathcal{K}_j^n\big) - B_{il}\big(\hat{U}_h^+|_{K_j(t_n^+)}\big)\\
&\quad - A_{l0}^1 \bar{U}_i(K_j(t_{n+1}^-)) + \mathcal{R}_{il}^1\big(U_h^-|_{\mathcal{Q}_j^n}, U_h^+|_{\mathcal{Q}_j^n}\big) - \mathcal{R}_{il}^2\big(U_h|_{\mathcal{K}_j^n}\big), \quad l = 1, \ldots, 4.
\end{aligned} \tag{28}$$

The space–time discontinuous Galerkin discretization results in a set of nonlinear equations (26) for the expansion coefficients $\hat{U}^n$. This set of nonlinear equations is solved with a full approximation storage (FAS) multigrid scheme, which is discussed in Section 4. Since the evaluation of the coupling terms B between space–time slabs is fairly complicated for general meshes, it is also useful to consider the equations for continuous grid motion. In this case these integrals are relatively simple; i.e.,

$$B_{il}\left(U_h^{+}|_{K_j(t_n^{+})}\right) = \mathcal{B}_{lm}\hat{U}_{mi}\left(\mathcal{K}_j^{n-1}\right),$$

with

$$\mathcal{B}_{lm} = \int_{K_j(t_n)} \psi_l(\bar{x}, t_n^{+})\psi_m(\bar{x}, t_n^{-})\, dK.$$

If we use the relations $\mathcal{B}_{00} = |K_j(t_n)|$ and $\hat{U}_{0i}(\mathcal{K}_j^{n-1}) = \bar{U}_i(K_j(t_n^{-}))$, then (27) is a standard finite volume discretization for the element mean.

Remark. It would have been more convenient to define the element mean flow field for the space–time element $\mathcal{K}$ instead of using the element mean flow field in $K_j(t_{n+1}^{-})$), but this would not result in a decoupling of the equations for the element mean from the equations for the fluctuations $\tilde{U}_h$ due to the weak coupling between the different time slabs in the weak formulation (15).

3.4. Flux Calculation

3.4.1. *Extension of the HLLC Scheme to Moving Meshes*

In Section 3.2.3 we introduced the HLLC flux into the weak formulation in order to prevent numerical oscillations around discontinuities. The formulation of the HLLC scheme discussed in Toro *et al.* [29, 30] and Batten *et al.* [3, 5] is, however, only valid for nonmoving meshes. In this section we will discuss the extension of the HLLC scheme to moving meshes. This extension is most easily accomplished by considering the structure of the wave pattern in the Riemann problem which is assumed in the HLLC scheme (see Fig. 2). The HLLC scheme assumes that 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

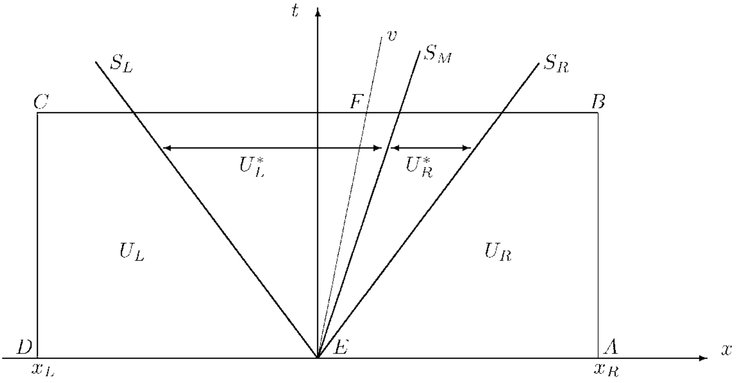


FIG. 2. Wave pattern used in the definition of the HLLC flux function for an element face moving with velocity v. Here S_L and S_R are the fastest left- and right-moving signal velocities. The solution in the star region U^* is divided by a wave with velocity S_M.

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

$$U_{HLLC} = \begin{cases} U_L \equiv U_h^-(t_m), & \text{if } S_L > v, \\ U_L^*, & \text{if } S_L \leq v < S_M, \\ U_R^*, & \text{if } S_M \leq v < S_R, \\ U_R \equiv U_h^+(t_m), & \text{if } S_R \leq v, \end{cases} \tag{29}$$

where depending on the grid velocity v we have to consider four different cases. The time interval Δt is chosen such that there is no interaction with waves coming from other Riemann problems.

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 $[t_m, t_m + \Delta t)$ by integrating the Euler equations over the control volumes $\square$ *DEFC* and $\square$ *EABF*, as shown in Fig. 2. Using Gauss' theorem we obtain for the control volume $\square$ *DEFC* the relation

$$\begin{aligned} &\int_{x_L}^{S_L \Delta t} U_L \, dx + \int_{S_L \Delta t}^{v \Delta t} U_h(x, t_m + \Delta t) \, dx \\ &\quad = \int_{x_L}^{0} U_h(x, t_m) \, dx + \int_{t_m}^{t_m + \Delta t} \hat{\mathcal{F}}(U_h(x_L, t)) \, dt - \int_{t_m}^{t_m + \Delta t} \hat{\mathcal{F}}(U_h^-(vt, t)) \, dt, \end{aligned} \tag{30}$$

and for the control volume $\square$ *EABF*

$$\begin{aligned} &\int_{v \Delta t}^{S_M \Delta t} U_h(x, t_m + \Delta t) \, dx + \int_{S_M \Delta t}^{S_R \Delta t} U_h(x, t_m + \Delta t) \, dx + \int_{S_R \Delta t}^{x_R} U_R \, dx \\ &\quad = \int_{0}^{x_R} U_h(x, t_m) \, dx + \int_{t_m}^{t_m + \Delta t} \hat{\mathcal{F}}(U_h^+(vt, t)) \, dt - \int_{t_m}^{t_m + \Delta t} \hat{\mathcal{F}}(U_h(x_R, t)) \, dt, \end{aligned} \tag{31}$$

with $\hat{\mathcal{F}}(U_h) = \bar{n}_{\mathcal{K}} \bar{\mathcal{F}}(U_h)$. If we introduce now the averaged solutions U_L^* and U_R^*, which are defined as

$$\begin{aligned} U_L^* &= \frac{1}{(S_M - S_L)\Delta t} \int_{S_L \Delta t}^{S_M \Delta t} U_h(x, t_m + \Delta t) \, dx, \\ U_R^* &= \frac{1}{(S_R - S_M)\Delta t} \int_{S_M \Delta t}^{S_R \Delta t} U_h(x, t_m + \Delta t) \, dx, \end{aligned}$$

and use the fact that $U_h^\pm$ is constant along the line $x = vt$ in the Riemann problem, then we obtain after subtracting (30) from (31) the following expression for the HLLC flux at the interface in the time interval $[t_m, t_m + \Delta t)$:

$$\begin{aligned} H_{HLLC}(U_L, U_R) &= \frac{1}{2}(\tilde{\mathcal{F}}(U_h^-(vt, t)) + \tilde{\mathcal{F}}(U_h^+(vt, t))) \\ &= \frac{1}{2}(\hat{\mathcal{F}}(U_L) + \hat{\mathcal{F}}(U_R) + ((S_L - v) + (S_M - v))U_L^* \\ &\quad + ((S_R - v) - (S_M - v))U_R^* - S_L U_L - S_R U_R). \end{aligned}$$

For the other three cases, $(S_L < v,\, S_R > v,\, S_M \le v)$, $(S_L < v,\, S_R < v,\, S_M < v)$, and $(S_L > v,\, S_R > v,\, S_M > v)$, a similar analysis can be made. If we combine the four cases, then we obtain the following expression for the HLLC flux at a moving interface in the time interval $[t_m, t_m + \Delta t)$:

$$\begin{aligned} H_{HLLC}(U_L, U_R) = \frac{1}{2}(&\hat{\mathcal{F}}(U_L) + \hat{\mathcal{F}}(U_R) - (|S_L - v| - |S_M - v|)U_L^* \\ &+ (|S_R - v| - |S_M - v|)U_R^* + |S_L - v|U_L \\ &- |S_R - v|U_R - v(U_L + U_R)). \end{aligned} \tag{32}$$

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 between the HLLC flux for moving and nonmoving meshes, we only state the final results. We will follow the approach of Batten *et al.* [3], who 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 with $\hat{u}^*$ 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; i.e.,

$$U_L^* = \frac{S_L - \hat{u}_L}{S_L - S_M} U_L + \frac{1}{S_L - S_M} \begin{pmatrix} 0 \\ (p^* - p_L)\bar{n}_{\mathcal{K}} \\ p^* S_M - p_L \hat{u}_L \end{pmatrix}, \tag{33}$$

with an identical relation for U_R^*, only with L replaced with R. The intermediate pressures are equal to

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

but 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 to Davis [14] as

$$S_L = \min(\hat{u}_L - a_L, \hat{u}_R - a_R), \quad S_R = \max(\hat{u}_L + a_L, \hat{u}_R + a_R), \tag{34}$$

with $a = \sqrt{\gamma p/\rho}$ the speed of sound. Batten *et al.* [5] showed that it is better to use wave velocities based on the Roe averaged velocities, but we did not notice any major difference with the simpler waves velocities defined in (34) for the simulations discussed in this article.

3.4.2. *Evaluation of Flux Integrals*

The flux integrals (23) and (24) are computed by transforming the integrals to the reference face $(-1, 1)^3$ and reference element $\hat{\mathcal{K}}$, respectively, after which the integrals are approximated according to product Gauss quadrature rules. For the element face flux integrals (23), a two-point product Gauss quadrature rule is used for the integration in the local coordinate directions ξ_1, ξ_2, and ξ_3, and a three-point Gauss quadrature rule is used for the integration in the local coordinate direction ξ_4. The volume flux integrals (24) are computed with a three-point product Gauss quadrature rule. In van der Ven and van der Vegt [34], Corollary 16, it is shown that these quadrature rules are sufficiently accurate to ensure that the discontinuous Galerkin discretization discussed in this article is second-order accurate in a suitable Sobolev norm. The product Gauss quadrature rules also evaluate the flux integrals sufficiently accurately to satisfy the geometric conservation law (GCL). The GCL, which was originally formulated by Thomas and Lombard [28], requires that a uniform flow field not be disturbed by the grid motion and that it be an essential condition in order to obtain at least first-order accuracy in time, as was proven by Guillard and Farhat [17].

The product Gauss quadrature rules are easy to implement but require 12 flux evaluations per element face integral and 81 flux evaluations per volume integral. This number can be slightly reduced using more sophisticated quadrature rules, as described by Stroud [27], but the number of flux evaluations remains large. In [34] we describe and analyze a technique to reduce the number of flux evaluations in the flux integration to one while maintaining the same second-order accuracy as obtained with the product Gauss quadrature rules. The discussion of this technique is, however, beyond the scope of this article.

3.5. Stabilization Operator

The discontinuous Galerkin finite element method without stabilization operator does not guarantee monotone solutions around discontinuities and sharp gradients. In these regions numerical oscillations develop when polynomials of degree one or higher are used. For the Runge–Kutta discontinuous Galerkin method Cockburn *et al.* [10] derived a local projection or slope limiter which guarantees monotone solutions for multidimensional scalar conservation laws. This approach was a major breakthrough for the numerical solution hyperbolic partial differential equations because initially discontinuous Galerkin finite element discretizations experienced severe stability limitations. The use of a slope limiter in combination with a DG method results in a robust numerical discretization and has become quite popular. We have used this technique to compute complex three-dimensional (unsteady) flows for aerodynamical applications in combination with local mesh refinement [31, 32]. Other applications of DG methods with limiters, including higher order discretizations, can be found in Cockburn and Shu [11], Cockburn *et al.* [13], and Kershaw *et al.* [20].

Despite its robustness the use of a slope limiter has serious disadvantages, since it may result in an unnecessary reduction in accuracy in smooth parts of the flow field and prevents convergence to steady state. The accuracy problem has been an important motivation for Cockburn and Shu to look at total variation bounded (TVB) DG discretizations [9], but these techniques are not easy to apply in multiple dimensions and contain problem-dependent constants which are difficult to estimate. Recently, Burbeau *et al.* [7] proposed limiters for second- and higher-order-accurate DG methods without problem-dependent constants which look promising, but they still need further testing on real applications.

The problems with the convergence to steady state caused by the limiter are more severe and originate from an inconsistency in the combination of a discontinuous Galerkin discretization and a limiter. Since the limited solution does not satisfy the steady-state discontinuous Galerkin equations, it is not possible to reduce the residual to machine accuracy. Instead, the scheme tries to converge to the unlimited solution, which suffers however from numerical oscillations, and the limiter must remain active to prevent this. This is particularly annoying for industrial applications, since it is unclear when to stop the calculations. Convergence to steady state is also important for unsteady problems. In Section 4 we solve the nonlinear equations for the DG expansions coefficients (26) by introducing a pseudo-time and marching the solution to steady state in pseudo-time with a FAS multigrid algorithm.

The problems in obtaining steady-state solutions with a limited DG method are well known but have received little attention, since most applications of DG methods have been to unsteady problems in combination with an explicit Runge–Kutta time integration method. After extensive testing we came to the conclusion that a better alternative is provided by stabilizing the discontinuous Galerkin method by adding artificial dissipation. This approach is also followed by Barth [2], Baumann [4], Cockburn and Gremaud [8], and Jaffre *et al.* [19] for the discontinuous Galerkin method and is standard in the streamline upwind Petrov Galerkin (SUPG) and Galerkin least-squares methods. In this section we will discuss new stabilization operators for the space–time discontinuous Galerkin method and in later sections we will demonstrate that this technique provides excellent shock capturing and convergence to steady state in pseudo-time. The stabilization operators use the jump in the polynomial representation at the element faces in the discontinuous Galerkin discretization and the element residual. In this way optimal use is made of the information contained in a DG discretization and we maintain the compact stencil of the discontinuous Galerkin discretization.

The effectiveness of the stabilization operator $\mathcal{D}$ defined in (25) strongly depends on the artificial viscosity matrix $\mathfrak{D}(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}) \in \mathbb{R}^{4\times 4}$. The definition of the artificial viscosity matrix is more straightforward when the stabilization operator acts independently in all computational coordinate directions. This is achieved by introducing the artificial viscosity matrix $\tilde{\mathfrak{D}} \in \mathbb{R}^{4\times 4}$ into computational space using the relation

$$\mathfrak{D}\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big) = R^T \tilde{\mathfrak{D}}\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big) R, \tag{35}$$

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

$$R = 2H^{-1}\,\mathrm{grad}\, G_K. \tag{36}$$

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

$$H = \mathrm{diag}(h_1, h_2, h_3, h_4),$$

with $h_i \in \mathbb{R}^+$ the leading terms of the expansion of the mapping G_K (3) in the computational coordinates ξ_i $(1 \le i \le 4)$. The multiplication with the factor two in (36) ensures that for orthogonal cells the matrix R is the rotation matrix from the computational space to the physical space. The integrals in the stabilization operator $\mathcal{D}_{nm}$ given by (25) can now be further evaluated, resulting in

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

(no summation on n), where 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.

The stabilization operator should act only in areas with discontinuities or when the mesh resolution is insufficient. This requirement can be directly coupled to the jump in the solution across element faces and the element residual, respectively, both of which are readily available in the discontinuous Galerkin discretization. In regions with smooth solutions these contributions are on the order of the truncation error and will therefore not reduce the accuracy in these regions. We have tested two models for the artificial coefficients.

Model I. In the artificial viscosity model I, only the jump in the pressure across the element faces influences the stabilization matrix. This technique works very well in subsonic and transonic flows with weak shocks. The artificial viscosity matrix is defined as

$$\begin{aligned}\tilde{\mathfrak{D}}_{qq}\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big) &= \frac{C'\lambda h_q}{\big|\mathcal{Q}_j^n\big|}\sum_{m=1}^{6}\frac{\big|p^+\big(x_{(m)}\big) - p^-\big(x_{(m)}\big)\big||\mathcal{S}_m|}{p^+\big(x_{(m)}\big) + p^-\big(x_{(m)}\big)}, \quad && q = 1, 2, 3,\\ &= 0, && \text{otherwise},\end{aligned}$$

with $p^\pm(x_{(m)}) = \gamma^\pm(p(x_{(m)}))$ the pressure at the centers of the faces $\mathcal{S}_m \subset \mathcal{Q}_j^n$ and γ the trace operator. The scaling factor λ is defined as $\lambda = |\bar{n}_{\mathcal{K}}\cdot(u - v)| + a$ and is the maximum of the eigenvalues of the flux Jacobians $\partial\bar{\mathcal{F}}/\partial U$ at the midpoints $x_{(m)}$ of the faces $\mathcal{S}_m$, with $\bar{n}_{\mathcal{K}}$ the space normal at $\mathcal{Q}_j^n$, u and v the fluid and grid velocity, and $a = \sqrt{\gamma p/\rho}$ the speed of sound. The constant C' is of order one. Other discontinuity sensors, based for instance on the density, have also been tested, but the difference with the pressure sensor generally was very small.

Model II. For problems with stronger discontinuities the artificial viscosity model proposed and analyzed by Jaffre *et al.* [19] is used. In this model, both the jumps at the element faces and the element residual are used to define the artificial viscosity; i.e.,

$$\begin{aligned}\tilde{\mathfrak{D}}_{qq}\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big) &= \max\Big(C_2 h_{\mathcal{K}}^{2-\beta} R_q\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big), C_1 h_{\mathcal{K}}^{\frac{3}{2}}\Big), \quad && q = 1, 2, 3,\\ &= 0, && \text{otherwise},\end{aligned}$$

with

$$\begin{aligned}R\big(U_h|_{\mathcal{K}_j^n}, U_h^*|_{\mathcal{K}_j^n}\big) &= \left|\sum_{k=0}^{3}\frac{\partial\mathcal{F}(U_h)}{\partial U_{h,i}}\frac{\partial U_{h,i}(G_K(0))}{\partial x_k}\right| + C_0\big|U_h^+\big(x_{(7)}\big) - U_h^-\big(x_{(7)}\big)\big|/h_{\mathcal{K}}\\ &\quad + \sum_{m=1}^{6}\frac{1}{h_{\mathcal{K}}}\big|\bar{n}_{\mathcal{K}}^T\tilde{\mathcal{F}}\big(U_h^+\big(x_{(m)}\big)\big) - \bar{n}_{\mathcal{K}}^T\tilde{\mathcal{F}}\big(U_h^-\big(x_{(m)}\big)\big)\big|, \qquad (37)\end{aligned}$$

with $h_{\mathcal{K}} = \sqrt{h_1^2 + h_2^2 + h_3^2 + h_4^2}$ and $U_{h,i}$ the components of U_h. The coefficients β, C_0, C_1, and C_2 are positive constants and set equal to $C_0 = 1.2$, $C_1 = 0.1$, $C_2 = 1.0$, and $\beta = 0.1$. For stronger shocks the addition of the quasilinear form of the conservation law in (37), which is the first contribution on the right-hand side of (37), significantly improves the robustness of the numerical scheme, since this contribution detects discontinuities very well. Numerical tests showed that the contributions of the element residual of the quasilinear equations and the contributions in the jump of the flux at the element faces are equally important.

4. SOLUTION OF THE NONLINEAR DG COEFFICIENT EQUATIONS

4.1. Multigrid Algorithm for Pseudo-Time Integration

The space–time discontinuous Galerkin discretization results in each element in a system of coupled nonlinear equations for the expansion coefficients $\hat{U}^n$. In this section we will describe an efficient multigrid technique to solve these nonlinear equations. The use of a multigrid scheme is motivated by the fact that it maintains the local, element-based structure of the discontinuous Galerkin discretization when a proper relaxation scheme is chosen. This greatly facilitates the use of a domain decomposition technique on parallel computers, which are our main target platforms. The multigrid technique has only been discussed for the linear advection–diffusion equation by Gopalakrishnan and Kanschat [16], which theoretically analyzed its performance. Until now multigrid techniques have not been used for DG discretizations of the Euler equations or on locally refined meshes. The development of an efficient technique has turned out to be nontrivial.

The nonlinear equations of the space–time discontinuous Galerkin discretization (26) are solved by augmenting them with a pseudo-time derivative; i.e.,

$$|K_j(t_{n+1}^-)| \frac{\partial \hat{U}\left(\mathcal{K}_j^n\right)}{\partial \tau} = -\frac{1}{\Delta t}\mathcal{L}(\hat{U}^n, \hat{U}^{n-1}), \tag{38}$$

where the right-hand side of (38) is divided by Δt to make it possible to obtain also steady-state solutions as $\Delta t \to \infty$, because $\frac{1}{\Delta t}\mathcal{L}$ is independent of Δt. The system (38) is integrated in pseudo-time using an optimized Runge–Kutta scheme in combination with a FAS multigrid algorithm to accelerate the convergence to steady state. On the coarse meshes only the equations for element mean are used. Depending on the type of artificial dissipation, we must, however, modify the Runge–Kutta scheme.

We define the following five-stage semiimplicit Runge–Kutta scheme, with $K^n = K_j(t_{n+1}^-)$, as relaxation operator for the multigrid procedure:

```
Procedure $S_I^k(k, \mathcal{L}^k, \mathcal{F}_O^k, \hat{W}^k)$:
    1. Initialize the first Runge–Kutta stage: $\hat{V}^{(0)} = \hat{W}^k$.
    2. Do for all stages $s = 1$ to 5:
$$\left(I + \frac{\alpha_s \bar{\lambda}}{|K^n|}\left(|K^n| I + \bar{\mathcal{D}}^k\left(\hat{V}^{(s-1)}\right)\right)\right)\hat{V}^{(s)}$$
$$= \hat{V}^{(0)} + \frac{\alpha_s \bar{\lambda}}{|K^n|}\left(\left(|K^n| I + \bar{\mathcal{D}}^k\left(\hat{V}^{(s-1)}\right)\right)\hat{V}^{(s-1)} - \mathcal{L}^k\left(\hat{V}^{(s-1)}, \hat{U}^k(\mathcal{K}^{n-1})\right) + \mathcal{F}_O^k\right). \quad (39)$$
    3. End do.
    4. Update solution: $\hat{W}^k = \hat{V}^{(5)}$.
End Procedure $S_I^k$.
```

In this procedure $\hat{W}^k$ are approximations to the expansion coefficients $\hat{U}(\mathcal{K}^n)$ at the different grid levels k, $\hat{U}^k(\mathcal{K}^{n-1})$ are the expansion coefficients of the restriction of $U_h(\mathcal{K}^{n-1})$ to the grid level k, and $\mathcal{F}_O^k$ represents the forcing function, which is defined in Procedure FAS. At the fine grid level $k = M$, the nonlinear operator $\mathcal{L}^M : \mathbb{R}^{5\times5} \times \mathbb{R}^{5\times5} \to \mathbb{R}^{5\times5}$ satisfies $\mathcal{L}^M = \mathcal{L}$, with $\mathcal{L}$ defined in (27)–(28), and we have $\hat{U}^M$, $\hat{V}^M$, $\hat{W}^M$, $\hat{\mathcal{F}}_O^M \in \mathbb{R}^{5\times5}$. At the coarse grid levels $1 \le k < M$, the components of the operators $\mathcal{L}^k : \mathbb{R}^5 \times \mathbb{R}^5 \to \mathbb{R}^5$ are equal to $\mathcal{L}_i^k = \mathcal{L}_{i0}$, and we have $\hat{V}^k$, $\hat{W}^k$, $\hat{\mathcal{F}}_O^k \in \mathbb{R}^5$. The coefficients $\hat{U}^k \in \mathbb{R}^5$ only consist of the coefficients of the mean flow field $\hat{U}_{0i}$ and the coefficient $\bar{\lambda}$ is defined as $\bar{\lambda} = \frac{\Delta\tau}{\Delta t}$, with $\Delta\tau$ the time step in the pseudo-time integration. The Runge–Kutta coefficients α_s are defined as $\alpha_1 = 0.0791451$, $\alpha_2 = 0.163551$, $\alpha_3 = 0.283663$, $\alpha_4 = 0.5$, and $\alpha_5 = 1.0$ and are optimized with a searching technique to improve the stability and smoothing properties of the Runge–Kutta scheme. The matrix $\bar{\mathcal{D}}^M \in \mathbb{R}^{5\times5}$ is defined as

$$\bar{\mathcal{D}}^M = \begin{pmatrix} 0 & 0 \\ 0 & \mathcal{D} \end{pmatrix}$$

at the fine grid level, with the dissipation matrix $\mathcal{D} \in \mathbb{R}^{4\times4}$ given by (25), and $\bar{\mathcal{D}}^k$ is zero at the coarse grid levels. Note, both dissipation operators discussed in Section 3.5 result in a diagonal matrix; hence the implicit treatment of this contribution is straightforward. The matrix $I \in \mathbb{R}^{5\times5}$ represents the identity matrix.

The Runge–Kutta scheme (39) is obtained from a second-order-accurate five-stage Runge–Kutta method,

$$\hat{V}^{(s)} = \hat{V}^{(0)} - \frac{\alpha_s\bar{\lambda}}{|K^n|}\mathcal{L}^k\big(\hat{V}^{(s-1)}, \hat{U}^k(\mathcal{K}^{n-1})\big), \quad \text{for } s = 1, \dots, 5, \tag{40}$$

by treating $\hat{V}$ in $\mathcal{L}^k(\hat{V}, \hat{U}^k(\mathcal{K}^{n-1}))$ semiimplicitly. This is accomplished by approximating $(\bar{\mathcal{A}} + \bar{\mathcal{D}})\hat{V}$ as $(|K^n|I + \bar{\mathcal{D}})\hat{V}^{(s)} + (\bar{\mathcal{A}} - |K^n|I)\hat{V}^{(s-1)}$. Here $\bar{\mathcal{A}}$ is the coefficient matrix multiplying $\hat{V}$ in (27) and (28). The contribution $\alpha_s\bar{\lambda}(|K^n|I + \bar{\mathcal{D}})\hat{V}^{(s-1)}/|K^n|$ then is added and subtracted to the right-hand side of (40) to restore the operator $\mathcal{L}^k$. This makes it possible to have a residual $\mathcal{L}^k(\hat{U}^n, \hat{U}^{n-1}) \approx 0$ when the solution converges to a steady state, which facilitates the definition of the multigrid algorithm.

The semiimplicit Runge–Kutta scheme is necessary because the pseudo-time integration would otherwise become unstable for values of $\bar{\lambda}$ on the order of one. The use of a semiimplicit Runge–Kutta scheme was proposed by Melson *et al.* [24] for time-accurate calculations with multigrid acceleration using a Jameson-type finite volume discretization of the compressible Navier–Stokes equations. In Section 4.2 we analyze this procedure and show that for small values of $\bar{\lambda}$, it also greatly enhances the stability of the pseudo-time-integration method for the space–time discontinuous Galerkin discretization.

The multigrid procedure also requires the definition of the coarse grid meshes and the restriction and prolongation operator. The unadapted mesh is generated such that it has a sufficient number of coarse grid levels. For most calculations at least three levels are used. In general the mesh is also locally refined, and starting at the leaves of the refinement tree, we traverse the tree backward until a sufficiently large number of cells is merged into coarse grid cells. The ratio of the number of cells between two grid levels is approximately eight in three dimensions. In Fig. 3 an example of this process is given. This process results in a

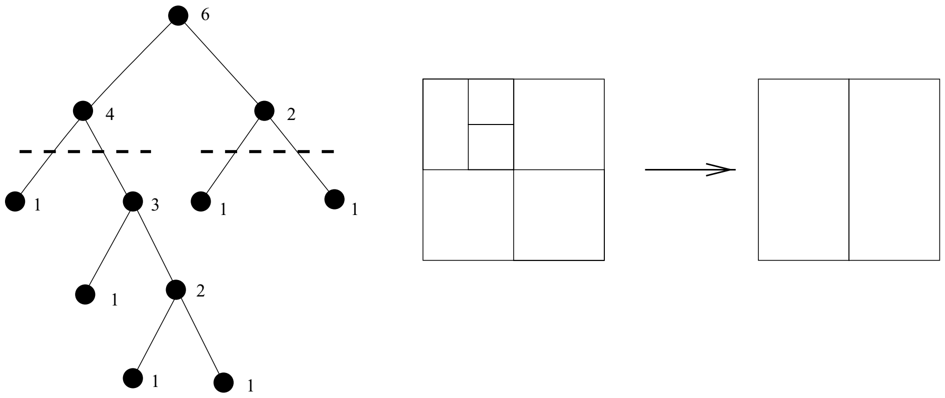


FIG. 3. Coarsening based on refinement tree. The numbers at the nodes of the tree refer to the number of leaves in the subtree. (Dashed lines) Where the tree is pruned. The fine grid cells and the resulting coarse grid cells are shown to the right.

number of tessellations $\mathcal{T}^n_{h,k}$, $1 \le k \le M$, for each grid level, which are defined as

$$\mathcal{T}^n_{h,k} := \left\{ \mathcal{K}^n_{j,k} \,\middle|\, \mathcal{K}^n_{j,k} = \bigcup_{j' \in I_{j,k}} \mathcal{K}^n_{j'}, \mathcal{K}^n_{j'} \in \mathcal{T}^n_h \right\},$$

with $I_{j,k}$ the indices of the elements $\mathcal{K}^n_{j'}$ which agglomerate into the coarse grid element $\mathcal{K}^n_{j,k}$. Note, at the fine grid level $k = M$ we have $\mathcal{T}^n_{h,M} = \mathcal{T}^n_h$. An example of three multigrid levels in a locally refined mesh is given in Fig. 4.

We also have to correct for the fact that the agglomerated coarse grid cells are not necessarily hexahedronal elements. This does not give serious problems, since at the coarse grid levels we only use equations for the element mean. These equations are identical to a first-order-accurate finite volume discretization for which it is straightforward to obtain a discretization on agglomerated elements. This is considerably more complicated for a second- or higher-order-accurate discontinuous Galerkin discretization, which also uses the equations for the flow field fluctuations, and is one of the main reasons for only using the equations for the element mean on the coarse grid levels.

For the discretization at the coarse grid levels we introduce the approximation spaces $V_{h,k}$, which are defined as

$$V_{h,k}\left(\mathcal{T}^n_{h,k}\right) := \begin{cases} \left\{ U_h \,|U_h|_{\mathcal{K}} = \bar{U}(K(t^-_{n+1})), \forall \mathcal{K} \in \mathcal{T}^n_{h,k} \right\}, & \text{if } 1 \le k < M, \\ V^1\left(\mathcal{T}^n_h\right), & \text{if } k = M. \end{cases}$$

The restriction operator $I^{k-1}_k : V_{h,k}(\mathcal{T}^n_{h,k}) \to V_{h,k-1}(\mathcal{T}^n_{h,k-1})$ is a volume-weighted average and is defined as

$$I^{k-1}_k U_h|_{\mathcal{K}^n_j} = \frac{\sum_{j' \in I_{j,k}} \hat{U}_{0i}\left(\mathcal{K}^n_{j'}\right)\left|\mathcal{K}^n_{j'}\right|}{\sum_{j' \in I_{j,k}} \left|\mathcal{K}^n_{j'}\right|}, \tag{41}$$

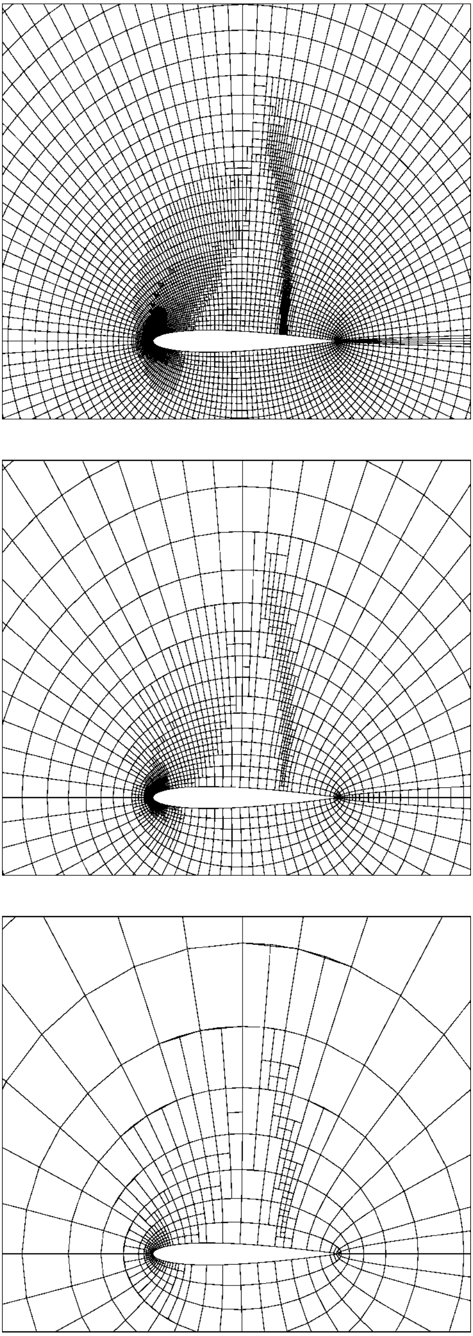


FIG. 4. Multigrid levels in an adapted mesh about the NACA 0012 airfoil.

with $\mathcal{K}_j^n \in \mathcal{T}_{h,k}^n$. The prolongation operator $I_{k-1}^k : V_{h,k-1}(\mathcal{T}_{h,k-1}^n) \rightarrow V_{h,k}(\mathcal{T}_{h,k}^n)$ is a pure injection and defined as

$$I_{k-1}^k U_h|_{\mathcal{K}_{j,k}^n} = \bar{U}(K_{j,k}(t_{n+1}^-)) \tag{42}$$

for all fine grid elements $\mathcal{K}_j \in \mathcal{T}_{h,k}^n$, which agglomerate into the coarse grid element $\mathcal{K}_{j,k}^n$.

We can now define a FAS multigrid algorithm for the space–time discontinuous Galerkin discretization on locally refined meshes:

```
Procedure FAS($k$, $\mathcal{L}^k$, $\mathcal{F}_O^k$, $\hat{W}^k$):
  1. Do $m_1$ Runge–Kutta steps $S^k(k, \mathcal{L}^k, \mathcal{F}_O^k, \hat{W}^k)$ at grid level $k$.
  2. Compute forcing function:
       $F_O^{k-1} = \mathcal{L}^{k-1}\big(I_k^{k-1}\hat{W}^k, I_k^{k-1}\hat{U}^k(\mathcal{K}^{n-1})\big) + I_k^{k-1}\big(\mathcal{F}_O^k - \mathcal{L}^k(\hat{W}^k, \hat{U}^k(\mathcal{K}^{n-1}))\big),$
     with $\mathcal{F}_O^M = 0$.
  3. If $k > 0$ do Procedure FAS($k-1$, $\mathcal{L}^{k-1}$, $\mathcal{F}_O^{k-1}$, $\hat{W}^{k-1}$).
  4. Update element mean solution at grid level $k$: $\hat{W}_{i0}^k = \hat{W}_{i0}^k + I_{k-1}^k(\hat{W}_{i0}^{k-1} - I_k^{k-1}\hat{W}_{i0}^k)$.
  5. Do $m_2$ Runge–Kutta steps $S^k(k, \mathcal{L}^k, \mathcal{F}_O^k, \hat{W}^k)$ at level $k$.
End Procedure FAS.
```

In the definition of the Procedure FAS we used (17)–(18), which allows us to apply the restriction and prolongation operator directly to the coefficients $\hat{U}$ without first projecting U_h to the basis functions ψ_m. The Procedure FAS uses a V-cycle multigrid strategy. Other cycling strategies, such as the W-cycle, can be obtained with minor changes to the Procedure FAS. The present multigrid algorithm makes rather crude assumptions at the coarse grid levels but has a good performance in practice. An example is given in Fig. 5 for calculations of the transonic flow about a NACA 0012 airfoil on a locally refined mesh. This figure shows that after each adaptation step, which results in the peaks in the residual, the residual

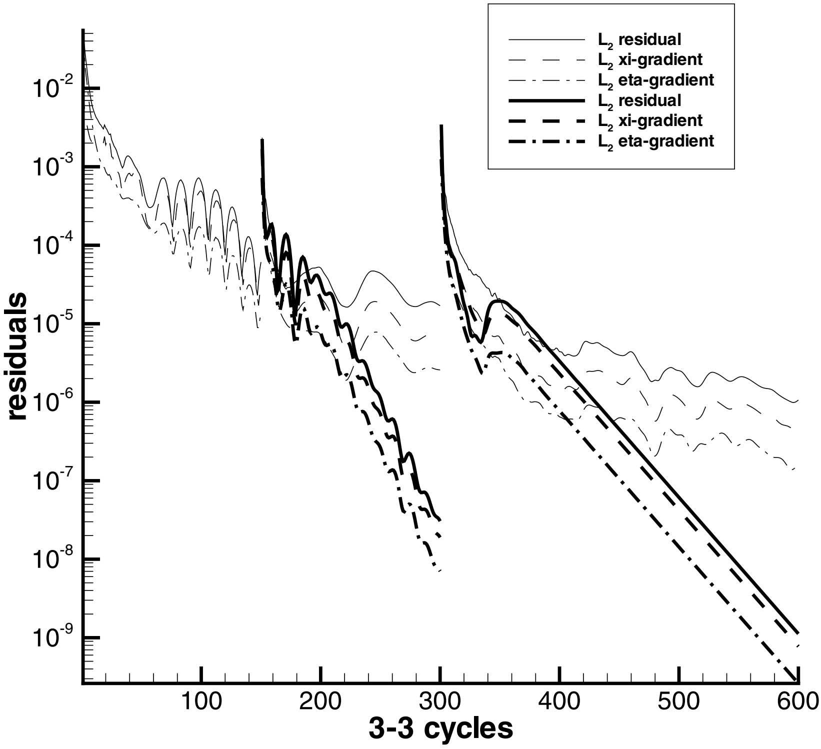


FIG. 5. Convergence rate comparison of the residual for the element mean and fluctuating DG coefficient equations using single and multigrid computations (dark lines) on a twice-adapted mesh of a NACA 0012 airfoil.

is efficiently reduced by the multigrid procedure, both for the equations of the element mean and for the fluctuations. We have extensively tested several other multigrid strategies, including solving the equations for the flow field fluctuations $\hat{U}_{mi}, m \geq 1$ on coarse meshes and more elaborate restriction and prolongation operators. Although some of these methods were promising in a two-level smoothing analysis and their theoretical performance was verified in calculations on simple model problems, none of these techniques came close to the performance of the multigrid algorithm for the solution of the Euler equations discussed in this section.

4.2. Stability Analysis of Pseudo-Time Integration

In this section we investigate the stability of the pseudo-time-integration method discussed in Section 4.1. As a model problem we use the linear advection equation $u_t + au_x = 0$, with a a positive constant. For this equation a relatively simple discretization is obtained, which is summarized in the Appendix. The Runge–Kutta scheme (39) is used for the pseudo-time integration.

If we assume that the time step, element size, and velocity remain constant, i.e., $\Delta t = \Delta t_n$, $\Delta x = \Delta x_j^{n+1} = \Delta x_j^n$, and $s = s^n_{j-\frac{1}{2}} = s^n_{j+\frac{1}{2}}$ for all j and n, and if we set the artificial viscosity coefficients equal to zero, then the operator $\mathcal{L}$ for the linear advection equation defined in the Appendix can be expressed as

$$\mathcal{L}(\hat{U}^n, \hat{U}^{n-1}) = \mathcal{A}\hat{U}\left(\mathcal{K}_j^n\right) - \mathcal{B}\hat{U}\left(\mathcal{K}_{j-1}^n\right) - \mathcal{C}\hat{U}\left(\mathcal{K}_j^{n-1}\right), \tag{43}$$

with the matrices $\mathcal{A}, \mathcal{B}, \mathcal{C} \in \mathbb{R}^{3\times 3}$ defined as

$$\mathcal{A} = \begin{pmatrix} 1+\delta & \delta & -\delta \\ -\delta & \frac{1}{3}+\delta & \delta \\ -2-\delta & -\delta & 2+\frac{4}{3}\delta \end{pmatrix}, \quad \mathcal{B} = \begin{pmatrix} \delta & \delta & -\delta \\ -\delta & -\delta & \delta \\ -\delta & -\delta & \frac{4}{3}\delta \end{pmatrix}, \quad \mathcal{C} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ -2 & 0 & 0 \end{pmatrix},$$

with $\delta = \Delta t(a-s)/\Delta x$ and $s \leq a$. Consider now the spatial Fourier mode

$$\hat{U}\left(\mathcal{K}_j^n\right) = e^{\imath\theta j}\hat{U}^F,$$

with $\theta \in [0, 2\pi)$ and $\imath = \sqrt{-1}$. Since the stability of the pseudo-time integration is determined by the transients we only consider the homogeneous part of the equation for the Fourier coefficient $\hat{U}^F$; i.e.,

$$\frac{d\hat{U}^F}{d\tau} = -\frac{1}{\Delta t}\mathcal{P}(\theta)\hat{U}^F, \tag{44}$$

with $\mathcal{P}(\theta) = \mathcal{A} - e^{-\imath\theta}\mathcal{B}$. The matrix $\mathcal{P} \in \mathbb{R}^{3\times 3}$ is nonsingular and we can write $\mathcal{P}$ as $\mathcal{P} = QMQ^{-1}$, with Q the matrix of right eigenvectors and M the diagonal matrix with the eigenvalues $\mu_m(\theta)$ $(m = 0, 1, 2)$ of $\mathcal{P}(\theta)$. If we introduce a new vector $\hat{V}^F = Q^{-1}\hat{U}^F$, then (44) becomes a system of uncoupled ordinary differential equations:

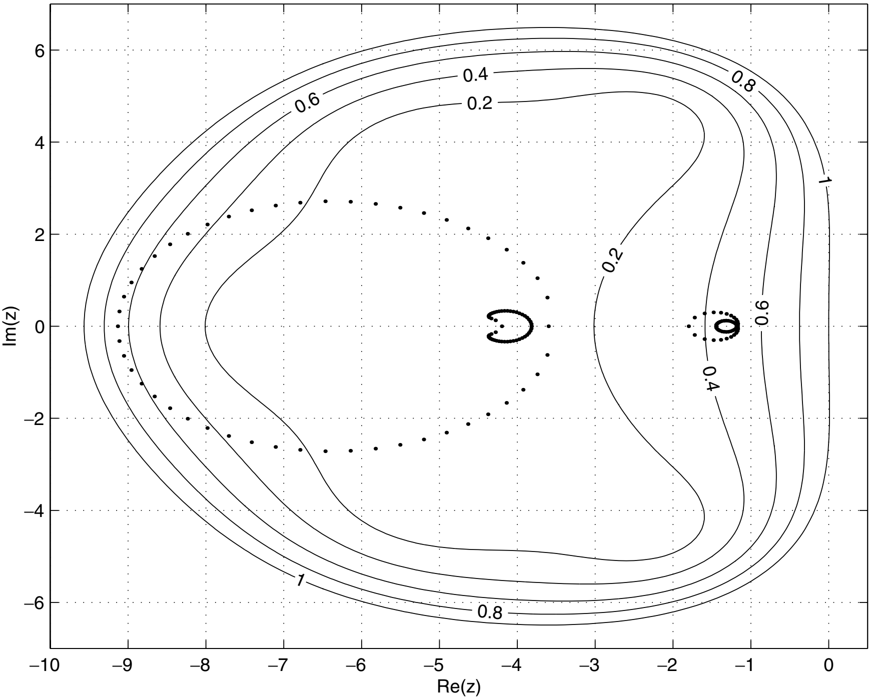
$$\frac{d\hat{V}_m^F}{d\tau} = -\frac{\mu_m(\theta)}{\Delta t}\hat{V}_m^F, \quad \text{for } m = 0, 1, 2.$$

This system of ordinary differential equations is solved with the semiimplicit Runge–Kutta scheme (39), which has an amplification factor $G(z)$, with $z \in \mathbb{C}$. The pseudo-time-integration method is stable if the amplification factor G satisfies the condition $|G(z_m(\theta))| \leq 1$, for $m = 0, 1, 2$ and $\theta \in [0, 2\pi)$, with $z_m(\theta)$ defined as $z_m(\theta) = -\frac{\Delta\tau}{\Delta t}\mu_m(\theta)$. The stability is analyzed for different values of the physical and pseudo-time step CFL numbers (defined as $\mathrm{CFL}_{\Delta t} = a\Delta t/\Delta x$ and $\mathrm{CFL}_{\Delta\tau} = a\Delta\tau/\Delta x$, respectively), and the ratio s/a. In Fig. 6 contour values of the stability domain $|G(z)| \leq 1$ for the five-stage semiimplicit Runge–Kutta scheme (39) with optimized coefficients are shown for the physical CFL numbers $\mathrm{CFL}_{\Delta t} = 1$ and 100. Also shown are the locus of the eigenvalues $z_m(\theta)$, $\theta \in [0, 2\pi)$, which must be inside the stability region to ensure the stability of the pseudo-time integration. For $\mathrm{CFL}_{\Delta t} = 1$ the Runge–Kutta scheme is stable for $\mathrm{CFL}_{\Delta\tau} \leq 1.94$, and for $\mathrm{CFL}_{\Delta t} = 100$ the pseudo-time-step CFL number must be less than $\mathrm{CFL}_{\Delta\tau} \leq 1.85$, which is unchanged for larger values of $\mathrm{CFL}_{\Delta t}$. The large stability domain and excellent smoothing properties of the semiimplicit Runge–Kutta method for small values of the physical-time-step CFL number is important for time-accurate simulations. In Fig. 7 the effect of the semiimplicit treatment of $\hat{V}$ in (39) is shown for $\mathrm{CFL}_{\Delta t} = 1$. For small physical-time-step CFL numbers the stabilizing effect of this technique is very large and the pseudo-time-step CFL number must be reduced to 1.08 to ensure stability when the semiimplicit technique is not used. For physical CFL numbers larger than 100 the effect of the semiimplicit Runge–Kutta scheme is, however, negligible. The effect of using optimized coefficients in the Runge–Kutta scheme (39) is also large, as can be seen in Fig. 7, where the stability contours for the semiimplicit Runge–Kutta scheme with coefficients $\alpha_s = \frac{1}{4}, \frac{1}{6}, \frac{3}{8}, \frac{1}{2}, 1$ for the stages $s = 1, \ldots, 5$ are shown. These are the coefficients for the Jameson Runge–Kutta scheme, which is a popular Runge–Kutta method in computational fluid dynamics and also is frequently used as a smoother in multigrid algorithms. For this Runge–Kutta scheme the pseudo-time CFL number must be reduced to $\mathrm{CFL}_{\Delta\tau} \leq 0.88$ when the physical CFL number is equal to $\mathrm{CFL}_{\Delta t} = 1$. When the physical CFL number is equal to $\mathrm{CFL}_{\Delta t} = 100$, the pseudo-time CFL number must be reduced to $\mathrm{CFL}_{\Delta\tau} \leq 0.95$ for the Jameson Runge–Kutta scheme. The effect of grid velocity is stabilizing if the grid velocity is in the range $0 \leq s \leq a$. This is a direct consequence of the relation $\delta = \mathrm{CFL}_{\Delta t}(1 - s/a)$. When the grid velocity is in this range, it reduces the effective physical-time-step CFL number and since the pseudo-time integration has a larger stability domain for smaller values of $\mathrm{CFL}_{\Delta t}$, this improves stability.

5. MESH ADAPTATION

In order to improve the accuracy of the discontinuous Galerkin discretization the computational mesh is adapted to provide more resolution in important flow structures. The mesh adaptation procedure is based on anisotropic refinement and coarsening of the mesh by subdividing and merging elements, independently in each of the local coordinate directions ξ_i $(1 \leq i \leq 4)$, of the reference element. The data structures and searching techniques for local mesh refinement and coarsening, which are suitable for the space–time discontinuous Galerkin finite element discretization, are essentially the same as that discussed in van der Vegt and van der Ven [31]. The mesh adaptation is controlled with a sensor function, which is based on the following quantities:

- shock sensor, which measures differences in flow quantities and total pressure loss across cell faces.



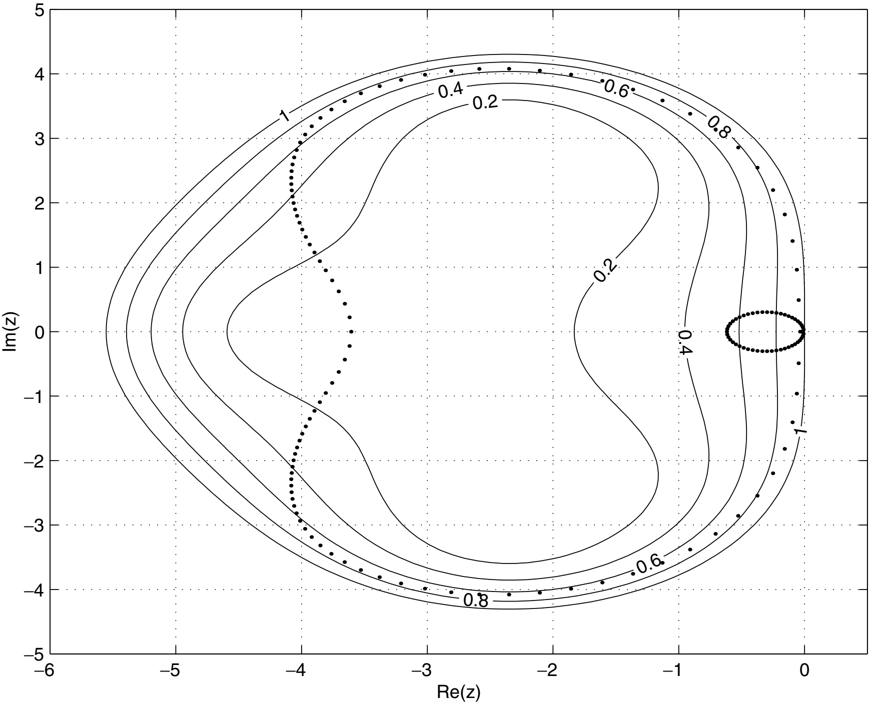
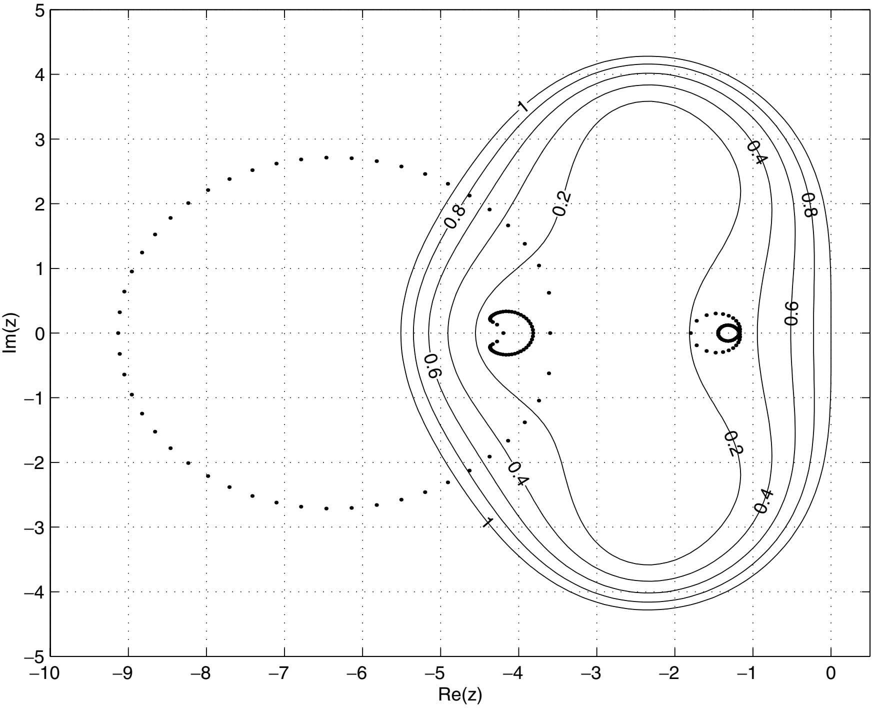


FIG. 6. Locus of the eigenvalues $z_m(\theta)$, $\theta \in [0, 2\pi)$ (dots), of the DG discretization of $u_t + au_x = 0$ and the stability domain of the five-stage semiimplicit Runge–Kutta method with optimized coefficients. $\mathrm{CFL}_{\Delta t} = 1.0$, $\mathrm{CFL}_{\Delta\tau} = 6.0$ (top); $\mathrm{CFL}_{\Delta t} = 100.0$, $\mathrm{CFL}_{\Delta\tau} = 2.2$ (bottom); no grid velocity.



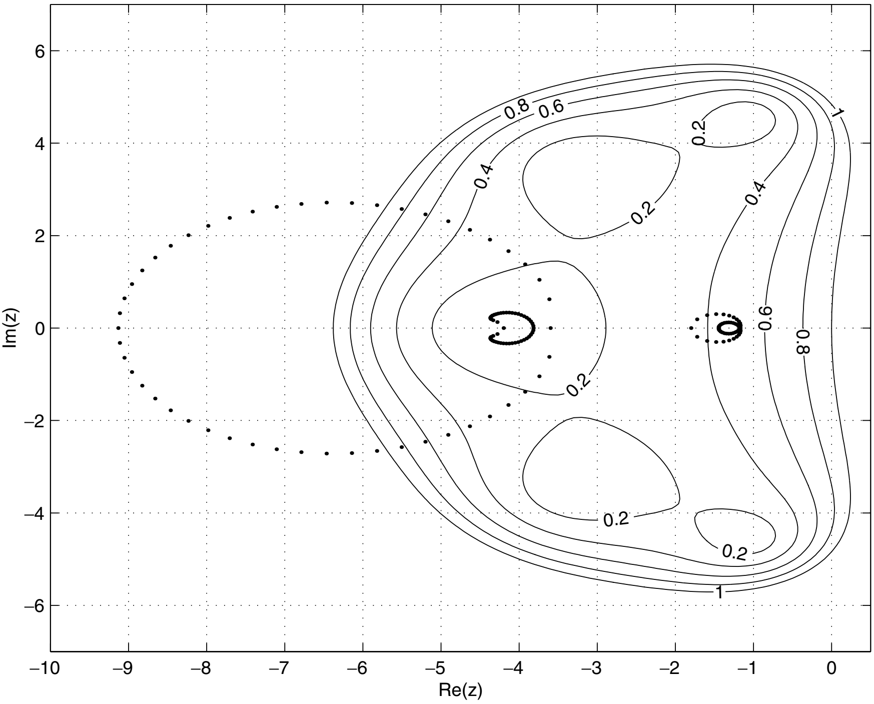


FIG. 7. Locus of the eigenvalues $z_m(\theta)$, $\theta \in [0, 2\pi)$ (dots), of the DG discretization of $u_t + au_x = 0$ and the stability domain of the explicit five-stage Runge–Kutta method (40) with optimized coefficients (top) and the five-stage semiimplicit Jameson Runge–Kutta scheme (bottom). $\text{CFL}_{\Delta t} = 1.0$; $\text{CFL}_{\Delta\tau} = 6.0$. No grid velocity.

- vorticity sensor, which measures the vorticity within an element.
- grid sensor, which either measures the anisotropy of the mesh or the mesh width of a cell.

We do not control the adaptation procedure using a posteriori error estimates, since this technique presently is not sufficiently well developed for the Euler equations. After the mesh adaptation, the coupling coefficients (22), which link the old and new space–time slabs, have to be computed in order to preserve time accuracy. In the next two sections we will discuss the evaluation of this contribution for element refinement and coarsening.

5.1. Space–Time Slab Coupling for Element Refinement

Given a refinement between two space–time slabs, where an element is divided into half in one of the computational coordinate directions, let $\mathcal{K}_j^{n-1}$ be an element in the space–time slab $\mathcal{T}_h^{n-1}$ and $\mathcal{K}_{j_0}^n$, $\mathcal{K}_{j_1}^n$ be two space–time elements in $\mathcal{T}_h^n$ such that $K_j^n := \bar{K}_j(t_n^-) = \bar{K}_{j_0}(t_n^+) \cup \bar{K}_{j_1}(t_n^+)$. The solution $U_h(\bar{x}, t_n^-)$ in element $\mathcal{K}_j^{n-1}$ is approximated as

$$U_{h,j}(\bar{x}, t_n^-) = \sum_{m=0}^{4} \hat{U}_m\big(\mathcal{K}_j^{n-1}\big)\psi_{m,j}(\bar{x}, t_n^-),$$

where the element index j is added to U_h and the basis functions ψ_m to indicate to which element they belong. The space–time slab coupling coefficients (22) for the elements $\mathcal{K}_{j_k}^n$, $k = 0$ or 1, can now be evaluated as

$$\begin{aligned} B_l\big(U_h^+|_{K_{j_k}(t_n^+)}\big) &= \int_{K_{j_k}(t_n^+)} U_{h,j}(\bar{x}, t_n^-)\psi_{l,j_k}(\bar{x}, t_n^+)\, dK \\ &= \sum_{m=0}^{3} \hat{U}_m\big(\mathcal{K}_j^{n-1}\big) \int_{K_{j_k}(t_n^+)} \psi_{m,j}(\bar{x}, t_n^-)\psi_{l,j_k}(\bar{x}, t_n^+)\, dK, \quad l = 0, \ldots, 4. \end{aligned} \tag{45}$$

The summation over the DG expansion coefficients is from zero to three, since $\psi_{4,j}(\bar{x}, t_n^-) = 0$ (cf. (5)). The evaluation of the integrals on the right-hand side of (45) requires an explicit expression for $\psi_{m,j}(\bar{x}, t_n^-)$ in the element K_{j_k}. Since the basis functions ψ_m are defined in the reference element $\hat{\mathcal{K}}$ using the basis functions $\hat{\psi}_m$, we must link $\psi_{m,j}(\bar{x}, t_n^-)$ to its representation in the reference element. Introduce the mappings L_i^k, with $1 \leq i \leq 3$ and $k = 0$ or 1, which are defined as

$$L_i^k : \hat{\mathcal{K}} \to \hat{\mathcal{K}}' : \xi_m \mapsto \begin{cases} \frac{1}{2}\xi_m - \frac{1}{2} + k, & \text{if } i = m, \\ \xi_m, & \text{if } i \neq m, \end{cases}$$

with

$$\hat{\mathcal{K}}' = (-1, 1) \times \cdots \times \underset{i\text{th entry}}{(-1 + k, k)} \times \cdots \times (-1, 1).$$

Here the subscript i denotes the coordinate direction in which the element is refined. Note that $G_{K_{j_k}}$ and $G_{K_j} \circ L_i^k$ are identical isoparametric mappings of $\mathcal{K}_{j_k}$. We can use this property to relate the basis function in the element $\mathcal{K}_j$ to the basis functions in its children $\mathcal{K}_{j_0}$ and $\mathcal{K}_{j_1}$. The basis functions $\psi_{m,j}$, restricted to K_{j_0} and K_{j_1} in (45), transform to

$$\psi_{m,j} = \hat{\psi}_m \circ G_{K_j}^{-1} = \hat{\psi}_m \circ L_i^k \circ G_{K_{j_k}}^{-1} = \begin{cases} \frac{1}{2}\psi_{m,j_k} - \frac{1}{2} + k, & \text{if } i = m, \\ \psi_{m,j_k}, & \text{if } i \neq m, \end{cases} \tag{46}$$

and we can use (46) to define the basis functions $\psi_{m,j}$ in the elements K_{j_k}. If we introduce (46) into (45) and transform back to the reference element $\hat{K}$, then we obtain simple expressions for the element integrals, which can be evaluated with a product Gauss quadrature rule with three points in each coordinate direction.

5.2. Space–Time Slab Coupling for Element Coarsening

Given a derefinement between two space–time slabs, let $\mathcal{K}_{j_0}^{n-1}$ and $\mathcal{K}_{j_1}^{n-1}$ be two elements in the space–time slab $\mathcal{T}_h^{n-1}$ and $\mathcal{K}_j^n \in \mathcal{T}_h^n$ be the space–time element such that $K_j^n := \bar{K}_j(t_n^+) = \bar{K}_{j_0}(t_n^-) \cup \bar{K}_{j_1}(t_n^-)$. The integral for the coupling coefficients (22) then can be evaluated as

$$\begin{aligned}
B_l\left(U_h^+|_{K(t_n^+)}\right) &= \int_{K_{j_0}(t_n^-)} U_{h,j_0}(\bar{x}, t_n^-)\psi_{l,j}(\bar{x}, t_n^+)\, dK + \int_{K_{j_1}(t_n^-)} U_{h,j_1}(\bar{x}, t_n^-)\psi_{l,j}(\bar{x}, t_n^+)\, dK \\
&= \sum_{m=0}^{3} \left(\hat{U}_m\left(\mathcal{K}_{j_0}^{n-1}\right) \int_{K_{j_0}(t_n^-)} \psi_{m,j}(\bar{x}, t_n^-)\psi_{l,j}(\bar{x}, t_n^+)\, dK \right. \\
&\quad \left. + \hat{U}_m\left(\mathcal{K}_{j_1}^{n-1}\right) \int_{K_{j_1}(t_n^-)} \psi_{m,j}(\bar{x}, t_n^-)\psi_{l,j}(\bar{x}, t_n^+)\, dK \right), \qquad (47)
\end{aligned}$$

with $\psi_{m,j}$ restricted to $\mathcal{K}_{j_0}$ and $\mathcal{K}_{j_1}$ given by (46). After transformation to the reference element it is straightforward to calculate the integrals in (47) with a product Gauss quadrature rule with three points in each coordinate direction.

6. DISCUSSION AND RESULTS

The space–time discontinuous Galerkin finite element method has been tested on a number of problems with increasing complexity. In this section we discuss results of simulations aimed at verifying and validating the algorithm. In [34] we present three-dimensional simulations, including a deforming wing, which demonstrate the applicability of the space–time DG method to unsteady aerodynamics.

6.1. Sod's Shock Tube Problem

Sod's problem is one of the classical shock tube problems (see Toro [30]). Its solution consists of a left-moving rarefaction wave and a right-moving contact discontinuity and shock. Two simulations have been performed, one with and one without mesh adaptation. Both simulations start on a uniform mesh with 100 cells in space. The time step is chosen such that the physical CFL number $\text{CFL}_{\Delta t}$ is less than or equal to 0.9. For the simulation with mesh adaptation, in the first time step two refinements have been carried out, resulting in 21 extra cells. The minimum mesh width is now one-quarter of the mesh width of the original mesh. The mesh adaptation on the initial solution is crucial because the error generated in the first time step cannot be recovered with adaptation during the simulation and a result similar to the uniform mesh solution would be obtained. In the subsequent time steps as many cells were added as removed, so the total number of cells remained constant in time. The maximum number of refinement levels has been restricted to one, which implies that no new cells with mesh widths less than half the mesh width of the original mesh are created.

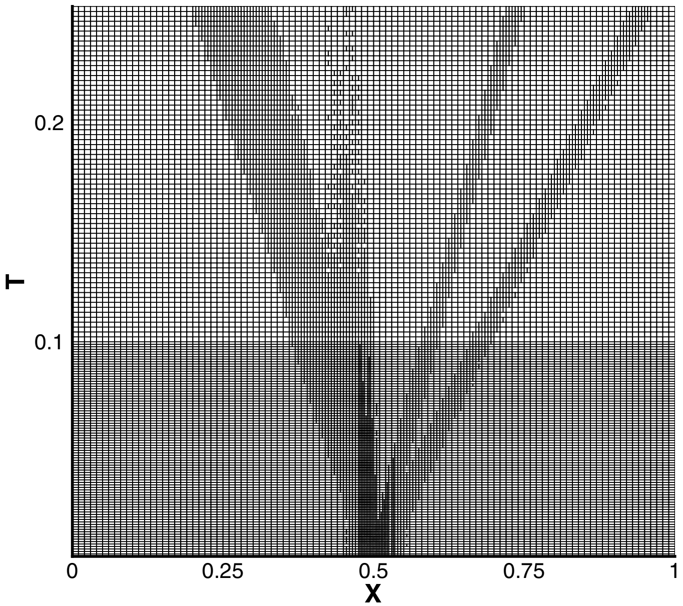


FIG. 8. Space–time mesh for the adaptive solution of Sod's shock tube problem.

The adapted space–time mesh is shown in Fig. 8. The space–time mesh clearly shows that the structure of the solution and the adaptation based on coarsening and refinement of elements follows the discontinuities without smearing. The flow solutions on the uniform and adapted mesh at $t = 0.2531$ are shown in Fig. 9. Clearly, the solution on the adapted mesh compares better with the exact solution. The flow solutions for Sod's problem have been obtained with dissipation model II, which results in nearly monotone solutions around the discontinuities. For all other subsonic and transonic problems the simpler dissipation model I is sufficient.

6.2. Accuracy Study of the Discontinuous Galerkin Discretization

The local elementwise discretization obtained with discontinuous Galerkin methods combines well with local mesh refinement and the discretization does not strongly depend on the mesh smoothness. Whether the DG method maintains its accuracy on nonsmooth meshes resulting from h-refinement must, however, be verified. In order to verify this an accuracy study was conducted using different meshes and comparing the numerical solution with the exact solution. For this purpose the subsonic two-dimensional flow through a channel with a $\sin^3$ bump is simulated on a sequence of meshes with 800, 3200, and 12800 elements. The coarsest mesh is shown in Fig. 10. At the inflow boundary total pressure, total temperature, and the velocity direction (normal to the inflow plane) are prescribed. At the outflow boundary the free-stream pressure is prescribed. Since the entropy p/ρ^γ should be conserved in subsonic isentropic flow, the L^2-norm of the difference between the computed entropy and the free-stream value is taken as a measure for the discretization error. In Fig. 11 the L^2-norm of the error is plotted for uniformly refined meshes. The L^2-error is proportional with $h^{5/2}$, which is better than the theoretical results presented by Cockburn [12] for the linear advection equation. This can be attributed to the

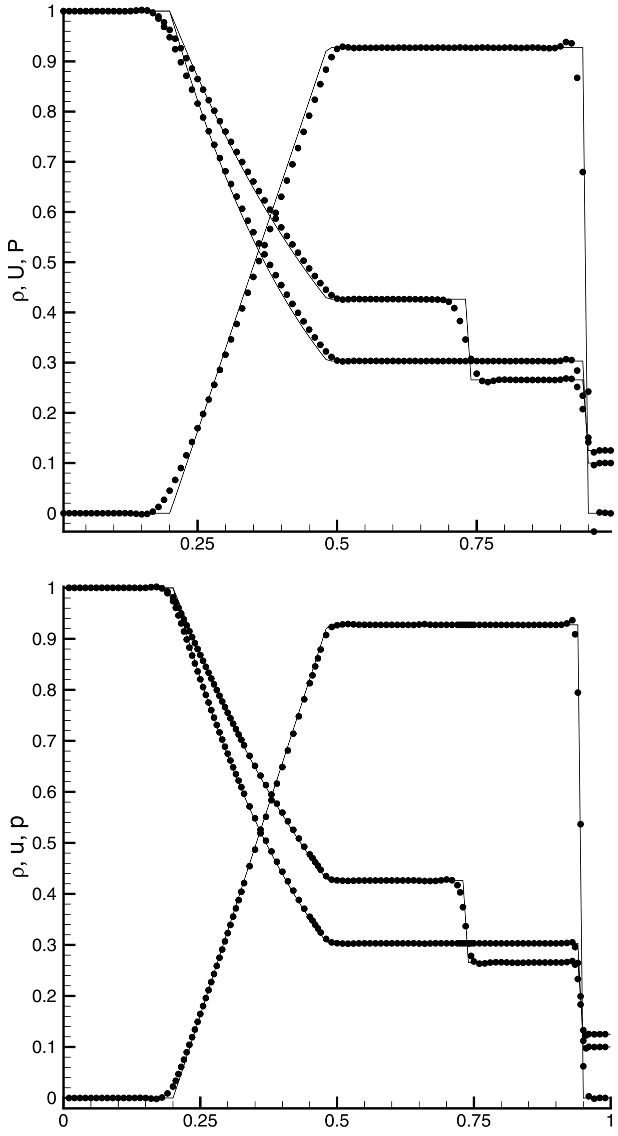


FIG. 9. Results of Sod's shock tube problem at $t = 0.2531$ on a uniform (top) and adapted (bottom) mesh. Computed results plotted as circles; exact solution plotted as lines.

fact that we use the data in the element center at t_{n+1}^{-}, which can be shown with a simple wave analysis for the linear advection equation to be $O(h)$ more accurate than the data at the element faces.

Each of the three meshes is also locally refined in two steps in order to test the accuracy of the method on nonsmooth meshes with hanging nodes. At each adaptation step, the mesh size is increased by 40%. Since the mesh adaptation parameters are the same for all three grids, the fine-to-coarse meshes have the following property: for an arbitrary region of the mesh, the average mesh width is halved with respect to the average mesh width in the next coarser mesh for the same region. Hence the series is suited for a grid convergence study to obtain the discretization error of the DG scheme on hanging nodes. A survey of the number

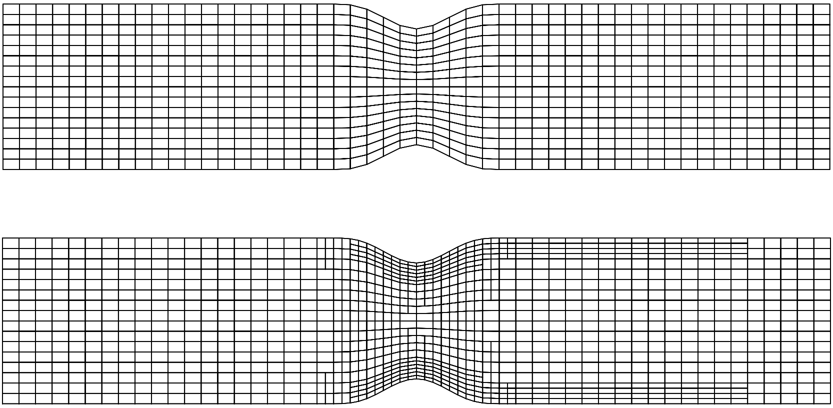


FIG. 10. Original and one-time-adapted mesh for converging–diverging channel.

of mesh points is given in Table I. In the adaptation the correct geometry of the bump is preserved. A view of the one-time-adapted mesh, which initially had 800 mesh points, is shown in Fig. 10. The L^2-norm of the error on the adapted meshes is shown in Fig. 11, which clearly demonstrates that the L^2-error on locally refined meshes in the discontinous Galerkin discretization has the same mesh dependence $h^{5/2}$ as on the uniformly refined meshes, despite the fact that the adapted mesh contains hanging nodes and is nonsmooth.

6.3. Oscillating NACA 0012 Airfoil in Transonic Flow

The performance of the space–time discretization and mesh adaptation algorithm on unsteady transonic flows was investigated with the simulation of the flow field about an oscillating NACA 0012 airfoil. The free-stream Mach number is 0.8, the pitching angle

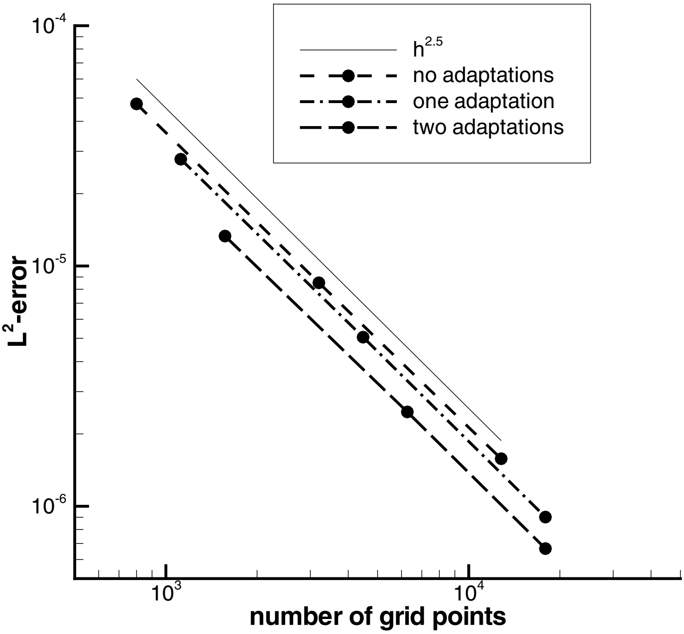


FIG. 11. L^2-error in flow calculations for converging–diverging channel on uniform and adapted meshes.

TABLE I
Number of Mesh Points in Uniformly and Adaptively Refined Meshes

	Coarse grid	Medium grid	Fine grid
Original	800	3200	12800
One adaptation	1120	4480	17920
Two adaptations	1568	6272	25088

range is between -0.5 and 4.5 degrees, and the oscillation period is $T = 20$ (normalized with L/a_∞, where L is the chord length and a_∞ is the free-stream speed of sound), which results in a circular frequency $\omega = \pi/10$. The flow field is computed both on a fine mesh with 32,768 elements and on an adapted mesh, which has approximately 9400 elements during the simulation. During each time step the coarse mesh is adapted, with first coarsening

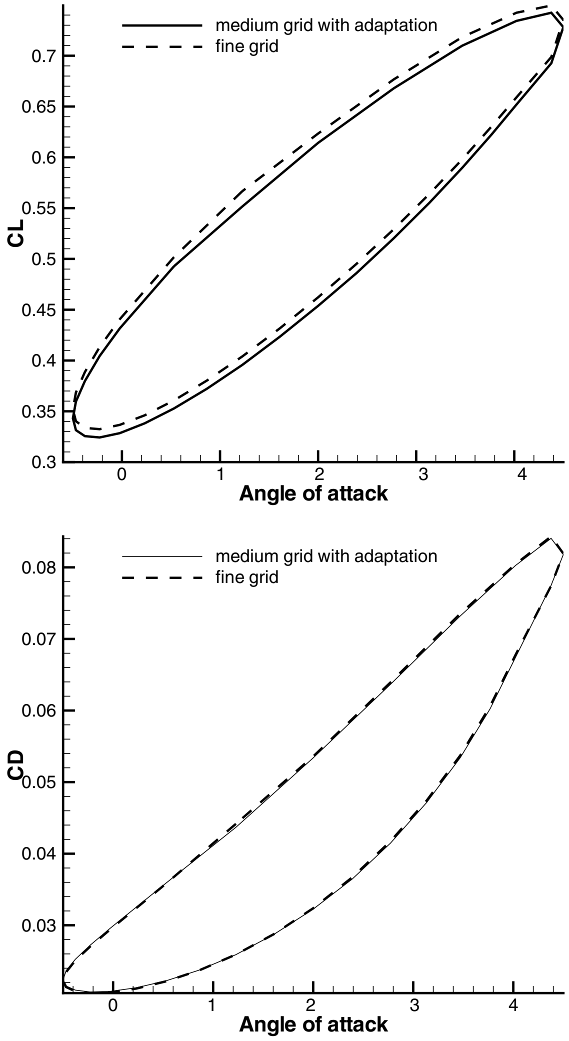


FIG. 12. Lift and drag coefficient on oscillating NACA 0012 airfoil ($M_\infty = 0.8$, $\omega = \pi/10$).

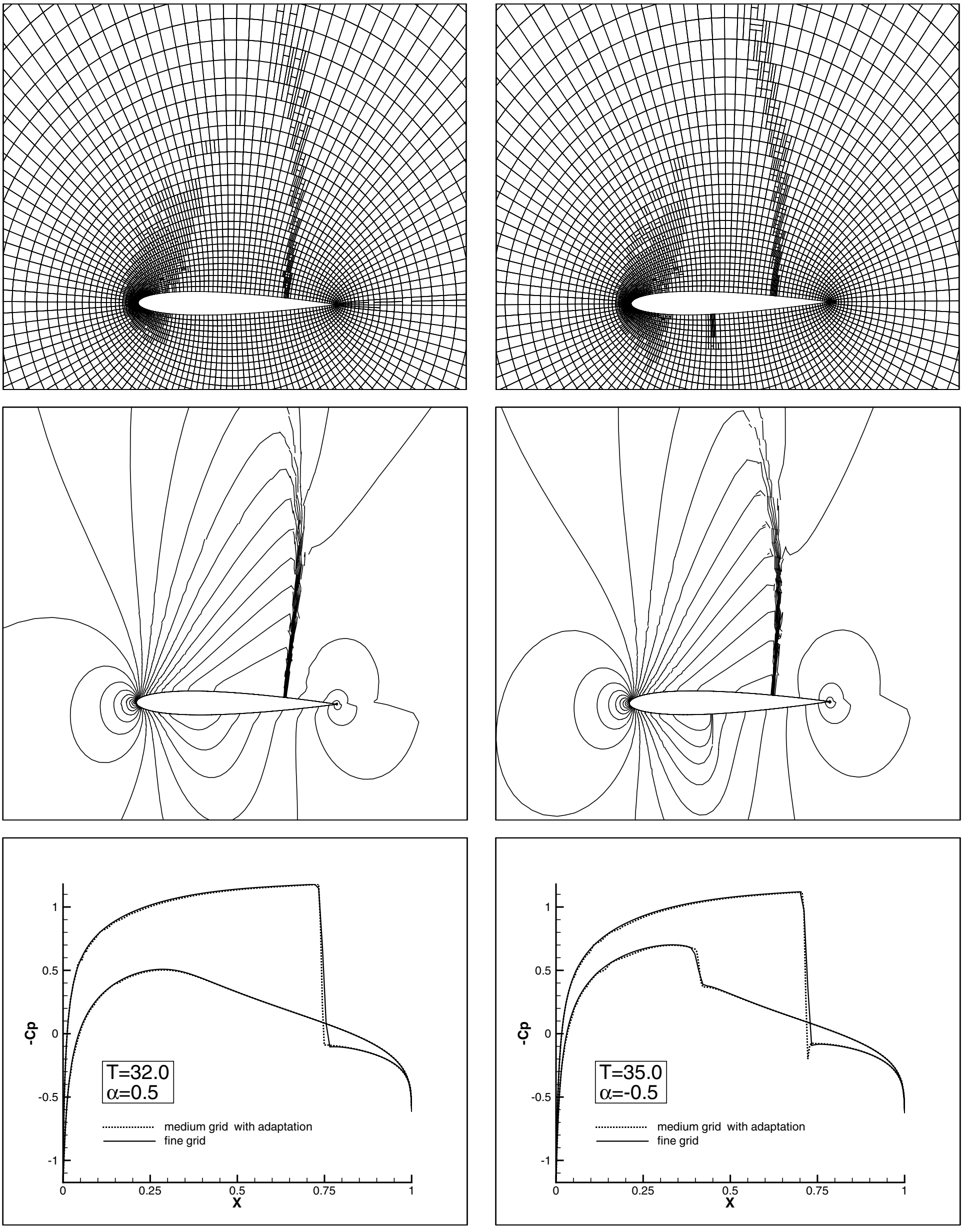


FIG. 13. Adapted mesh around oscillating NACA 0012 airfoil, contours of density, and pressure coefficient C_p on the airfoil surface for $\alpha = 0.5^\circ$ (pitching downward) and $\alpha = -0.5^\circ$ ($M_\infty = 0.8$, $\omega = \pi/10$).

followed by refinement. Both simulations used a time step of 1.0 for the interval [3.0, 13.0] of a period, and a time step of 0.5 in the remaining part of the period. The smaller time steps during this part of the oscillation period are necessary, since the shock at the lower side of the airfoil has a greater velocity than the shock at the upper side. If the shock moves through several cells during a time step this will result in numerical oscillations, since no artificial dissipation or limiting is applied in the time direction. In Fig. 12 the hysteresis curves of the lift and drag force coefficients C_L and C_D are shown. The results on the fine and adapted mesh are nearly identical, where the difference in the lift coefficient can be attributed to the

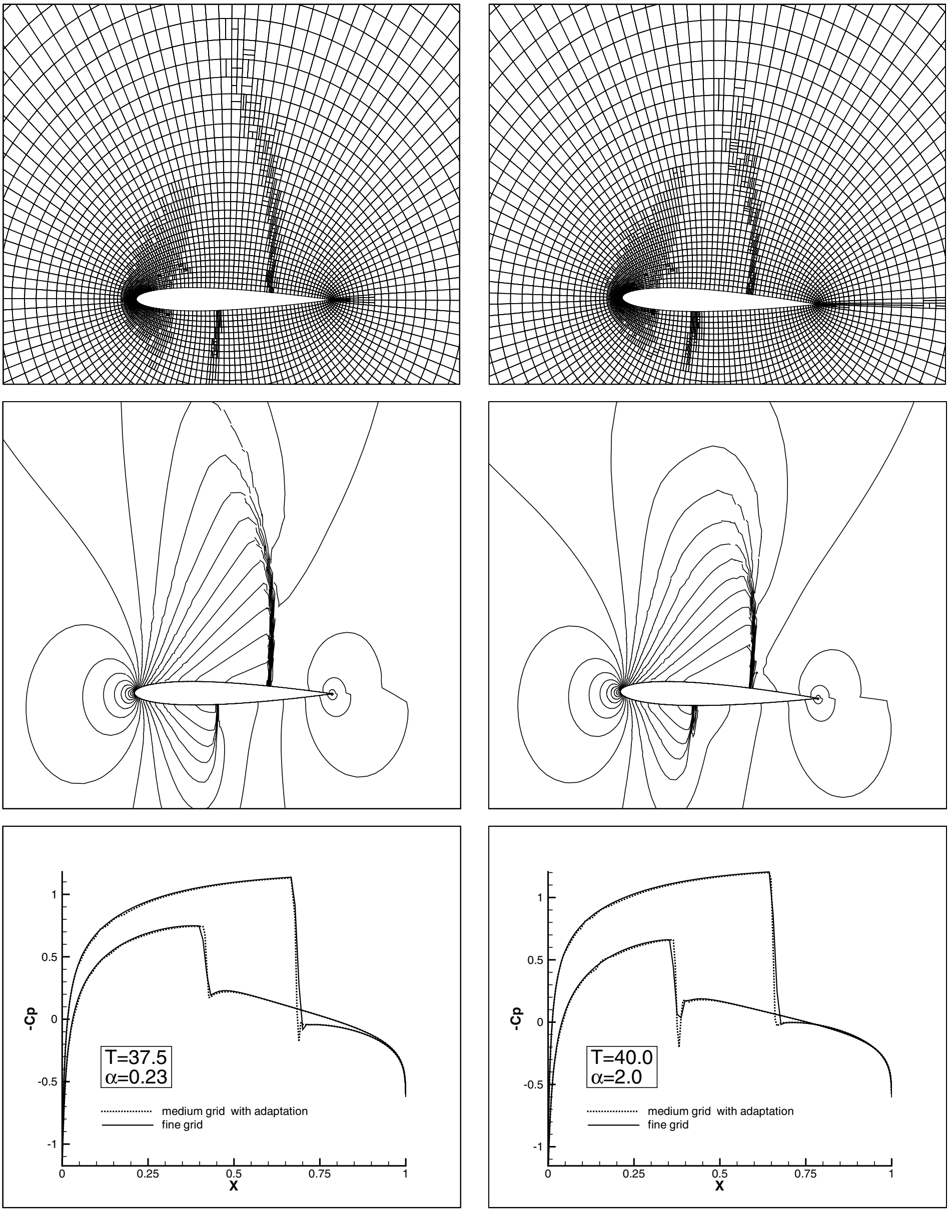


FIG. 14. Adapted mesh around oscillating NACA 0012 airfoil, contours of density, and pressure coefficient C_p on the airfoil surface for $\alpha = 0.23^\circ$ (pitching upward) and $\alpha = 2.0^\circ$ (pitching upward) ($M_\infty = 0.8, \omega = \pi/10$).

improved accuracy in the shock due to the mesh adaptation. This can be inferred from the pressure coefficients C_p at the wing shown in Figs.13 to 15, where the pressure coefficient is defined as $C_p = (p - p_\infty)/\frac{1}{2}\rho V_\infty^2$, with p_∞ and V_∞ the free-stream pressure and velocity, respectively. The pressure coefficients for the fine and adapted mesh are nearly identical, except in the shock, where the adapted mesh captures the discontinuity better. The physical interpretation of the flow phenomena shown in Fig. 15 at time $t = 42.5$, which appear at the lower side of the airfoil when the shock dissappears, is not clear. However, both the adapted mesh and the fine mesh flow results predict the same phenomena.

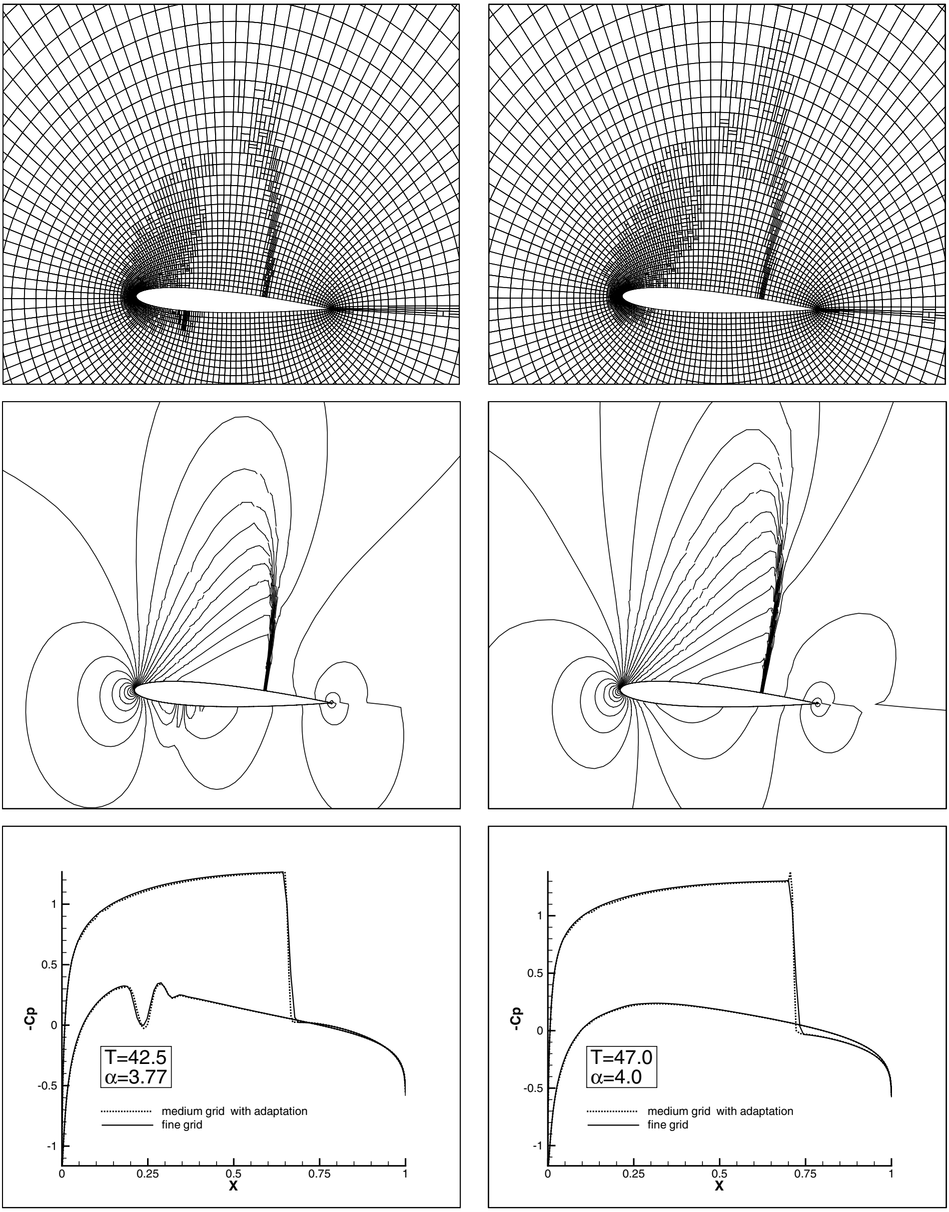


FIG. 15. Adapted mesh around oscillating NACA0012 airfoil, contours of density, and pressure coefficient C_p on the airfoil surface for $\alpha = 3.77°$ (pitching upward) and $\alpha = 4.0°$ (pitching downward) ($M_\infty = 0.8, \omega = \pi/10$).

Figures 13 to 15 also show that the mesh adaptation does not negatively influence the time accuracy and is very efficient in capturing the flow discontinuities, also for the weak shock at the lower side of the wing, which periodically disappears.

7. CONCLUDING REMARKS

In this article we have presented a new space–time discontinuous Galerkin finite element method for the time-accurate solution of inviscid compressible flows on dynamic, hexahedron-type meshes. The accuracy is improved using local mesh refinement and we have presented an efficient pseudo-time-integration technique with multigrid convergence acceleration to solve the nonlinear equations for the expansion coefficients in the DG discretization. The space–time DG method has been demonstrated to combine well with local mesh refinement in various simulations and maintains accuracy on nonsmooth meshes. This makes the space–time DG method an interesting technique for complex aerodynamic and aeroelastic problems. In a separate article [34], we address the issue of improving the computational efficiency of the space–time discontinuous Galerkin discretization and demonstrate the ability of the DG method when used in three-dimensional unsteady flows.

APPENDIX

Discontinuous Galerkin Discretization for Linear Advection Equation

In this appendix we summarize the space–time discontinuous Galerkin finite element discretization for the linear advection equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0,$$

with $a > 0$. This results in a relatively simple linear system, which is useful for analyzing the properties of the numerical discretization (see for instance Section 4.2). The space–time discontinuous Galerkin discretization for the linear advection equation using a mesh with grid velocities $s_j \leq a$, $j = 1, \dots N$, with N the number of mesh points, can be represented in matrix form as

$$\mathcal{A}\hat{U}\left(\mathcal{K}_j^n\right) - \mathcal{B}\hat{U}\left(\mathcal{K}_{j-1}^n\right) = \mathcal{C}\hat{U}\left(\mathcal{K}_j^{n-1}\right),$$

with

$$\mathcal{A} = \begin{pmatrix} \Delta x_j^{n+1} + c^n_{j+\frac{1}{2}} & c^n_{j+\frac{1}{2}} & -c^n_{j+\frac{1}{2}} \\ 2a_1 + c^n_{j+\frac{1}{2}} - 2a\Delta t_n & \frac{1}{3}a_2 + c^n_{j+\frac{1}{2}} + d_{11} & -2a_1 - c^n_{j+\frac{1}{2}} + 2a\Delta t_n \\ -\Delta x_j^n - \Delta x_j^{n+1} - c^n_{j+\frac{1}{2}} & -c^n_{j+\frac{1}{2}} & \frac{2}{3}a_3 + \frac{4}{3}c^n_{j+\frac{1}{2}} + d_{22} \end{pmatrix},$$

$$\mathcal{B} = \begin{pmatrix} c^n_{j-\frac{1}{2}} & c^n_{j-\frac{1}{2}} & -c^n_{j-\frac{1}{2}} \\ -c^n_{j-\frac{1}{2}} & -c^n_{j-\frac{1}{2}} & c^n_{j-\frac{1}{2}} \\ -c^n_{j-\frac{1}{2}} & -c^n_{j-\frac{1}{2}} & \frac{4}{3}c^n_{j-\frac{1}{2}} \end{pmatrix}, \quad \mathcal{C} = \begin{pmatrix} \Delta x_j^n & 0 & 0 \\ 0 & \frac{1}{3}\Delta x_j^n & 0 \\ -2\Delta x_j^n & 0 & 0 \end{pmatrix},$$

with $\Delta x_j^n = x_{j+1}^n - x_j^n$, $\bar{x}_j^n = \frac{1}{2}(x_j^n + x_{j+1}^n)$, $a_1 = \bar{x}_j^{n+1} - \bar{x}_j^n$, $a_2 = 2\Delta x_j^{n+1} - \Delta x_j^n$, $a_3 = 2\Delta x_j^n + \Delta x_j^{n+1}$, $c^n_{j\pm\frac{1}{2}} = \Delta t_n(a - s_{j\pm\frac{1}{2}})$, and $s^n_{j+\frac{1}{2}} = (x_{j+1}^{n+1} - x_{j+1}^n)/\Delta t_n$. Here x_j^n and x_{j+1}^n denote the beginning and end points of the element at time t_n, respectively. The terms d_{11} and d_{22} are determined by the artificial dissipation operator.

ACKNOWLEDGMENTS

The advice and continued support of Dr. B. Oskam (NLR) during the course of this project is greatly appreciated, especially his essential contributions to the development of the artificial dissipation operator. Sincere thanks are also due to O. J. Boelens (NLR) for conducting many simulations with the program HEXADAP, which significantly contributed to the validation and improvements of the numerical scheme. We also thank Y. Stalder (UT) for assistance with the stability analysis of the Runge–Kutta scheme and the testing of the HLLC scheme, and J. J. Sudirham (UT) for conducting the accuracy tests.

REFERENCES

1. F. Bassi and S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, *J. Comput. Phys.* **138**, 251 (1997).
2. T. J. Barth, *Numerical Methods for Gasdynamic Systems on Unstructured Meshes*, Lecture Notes in Computational Science and Engineering, Vol. 5 (Springer-Verlag, Berlin, 1998).
3. P. Batten, N. Clarke, C. Lambert, and D. Causon, On the choice of wave speeds for the HLLC Riemann solver, *SIAM J. Sci. Stat. Comput.* **18**, 1553 (1997).
4. C. E. Baumann, *An hp-Adaptive Discontinuous Finite Element Method for Computational Fluid Dynamics*, Ph.D. dissertation (The University of Texas at Austin, 1997).
5. P. Batten, M. A. Leschziner, and U. C. Goldberg, Average-state Jacobians and implicit methods for compressible viscous and turbulent flows, *J. Comput. Phys.* **137**, 38 (1997).
6. C. L. Bottasso, On the computation of the boundary integral of space-time deforming finite elements, *Commun. Numer. Methods Eng.* **13**, 53 (1997).
7. A. Burbeau, P. Sagaut, and Ch.-H. Bruneau, A problem-independent limiter for high-order Runge-Kutta discontinuous Galerkin methods, *J. Comput. Phys.* **169**, 111 (2001).
8. B. Cockburn and P. A. Gremaud, Error estimates for finite element methods for nonlinear conservation laws, *SIAM J. Numer. Anal.* **33**, 522 (1996).
9. B. Cockburn and C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II: General framework, *Math. Comput.* **52**, 411 (1989).
10. B. Cockburn, S. Hou, and C.-W. Shu, The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV: The multidimensional case, *Math. Comput.* **54**, 545 (1990).
11. B. Cockburn and C.-W. Shu, The Runge-Kutta discontinuous Galerkin method for conservation laws, V, *J. Comput. Phys.* **141**, 199 (1998).
12. B. Cockburn, *Discontinuous Galerkin Methods for Convection-Dominated Problems*, Lecture Notes in Computational Science and Engineering, Vol. 9 (Springer-Verlag, Berlin, 1999).
13. B. Cockburn, G. E. Karniadakis, and C.-W. Shu, Eds., *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11 (Springer-Verlag, Berlin, 2000).
14. S. F. Davis, Simplified second-order Godunov-type methods, *SIAM J. Sci. Stat. Comput.* **9**, 445 (1988).
15. M. Giles and E. Süli, Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality. *Acta Numer.* 145 (2002).
16. J. Gopalakrishnan and G. Kanschat, A multilevel discontinuous Galerkin method, *Numer. Math.*, in press.
17. H. Guillard and C. Farhat, *On the Significance of the GCL for Flow Computations on Moving Meshes*, AIAA Paper 99-0793 (1999).
18. P. Hansbo, Aspects of conservation in finite element flow computations, *Comput. Methods Appl. Mech. Eng.* **117**, 423 (1994).
19. J. Jaffre, C. Johnson, and A. Szepessy, Convergence of the discontinuous Galerkin finite element method for hyperbolic conservation laws, *Math. Models Meth. Appl. Sci.* **5**, 367 (1995).
20. D. S. Kershaw, M. K. Prasad, M. J. Shaw, and J. L. Milovich, 3D unstructured mesh ALE hydrodynamics with the upwind discontinuous finite element method, *Comput. Methods Appl. Mech. Eng.* **158**, 81 (1998).
21. M. Lesoinne and 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**, 71 (1996).

22. R. B. Lowrie, P. L. Roe, and B. van Leer, space-time methods for hyperbolic conservation laws, in *Barriers and Challenges in Computational Field Dynamics*, *ICASE/LaRC Interdiscip. Ser. Sci. Eng. 6*, edited by V. Venkatakrishnan *et al.*, p. 79 (Kluwer Acad. Publ., Dordrecht, 1998).

23. A. Masud and T. J. R. Hughes, A space-time Galerkin/least-squares finite element formulation of the Navier-Stokes equations for moving domain problems, *Comput. Methods Appl. Mech. Eng.* **146**, 91 (1997).

24. N. D. Melson, M. D. Sanetrik, and H. L. Atkins, Time-accurate Navier-Stokes calculations with multigrid acceleration, in *Proc. 6th Copper Mountain Confer. on Multigrid Methods*, pp. 423–437 (NASA Langley Research Center, 1993).

25. Ch. Schwab, *hp-FEM for Fluid Flow Simulation*, Lecture Notes in Computational Science and Engineering, Vol. 9 (Springer-Verlag, Berlin, 1999).

26. F. Shakib, T. J. R. Hughes, and Z. Johan, A new finite element method for computational fluid dynamics. X. The compressible Euler and Navier-Stokes equations, *Comput. Methods Appl. Mech. Eng.* **89**, 141 (1991).

27. A. H. Stroud, *Approximate Calculation of Multiple Integrals* (Prentice–Hall, Englewood Cliffs, NJ, 1971).

28. P. D. Thomas and C. K. Lombard, Geometric conservation law and its application to flow computations on moving grids, *AIAA J.* **17**, 1030 (1979).

29. E. F. Toro, M. Spruce, and W. Speares, Restoration of the contact surface in the HLL-Riemann solver, *Shock Waves* **4**, 25 (1994).

30. E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics. A Practical Introduction*, 2nd ed. (Springer-Verlag, Berlin, 1999).

31. 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 (1998).

32. J. J. W. van der Vegt, H. van der Ven, and O. J. Boelens, Discontinuous Galerkin finite element methods for hyperbolic partial differential equations, in *Godunov Methods: Theory and Applications*, edited by E. F. Toro, p. 985 (Kluwer Academic/Plenum, New York/London, 2001).

33. J. J. W. van der Vegt and H. van der Ven, Slip flow boundary conditions in discontinuous Galerkin discretizations of the Euler equations of gas dynamics, in *Proceedings Fifth World Congress on Computational Mechanics, July 7–12, 2002, Vienna, Austria*, edited by H. A. Mang, F. G. Rammerstorfer, and J. Eberhardsteiner, http://wccm.tuwien.ac.at.

34. 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. Methods Appl. Mech. Eng.* **191**, 4747–4780 (2002).