Interface Velocity Consistency in time-accurate flow simulations on dynamic meshes

D. Muffo, G. Quaranta, A. Guardone and P. Mantegazza

Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano,
Via La Masa, 34, 20158 Milano, Italy

Abstract

An Interface Velocity Consistency (IVC) condition for compressible flow computations on moving grid is presented, which relates the local interface velocity to the time derivative of the portion of the cell volume pertaining to the interface itself, thus allowing a consistent evaluation of the finite volume integrals of the fluxes at cell interfaces. The IVC condition is demonstrated to be a sufficient condition for fulfilling the well-known Geometric Conservation Law (GCL), which guarantees that a uniform flow is correctly computed by the flow solver. Indeed, the IVC condition generalizes the GCL to possibly nonuniform flows and it amounts to solve an additional system of ODE for the interface velocities together with the ODE system representing the space-discrete flow equations over moving domains. The flow solver is presented in a general form by introducing a suitable transformation matrix that allows for virtually any numerical flux to be readily available within the Arbitrary Lagrangian Eulerian (ALE) framework. The correct treatment of boundary conditions on dynamic meshes is also thoroughly detailed. A simple but effective approach to mesh movement, based on a description of the grid elements as deformable bodies is presented; the new grid at each time step is obtained as the solution of a constrained elastic continuum problem. A suitable treatment of boundary conditions guarantees a high grid quality as the simulation progresses. Both two- and three-dimensional cases are thoroughly studied and numerical simulations are presented to investigate the accuracy of the present approach for different time discretization techniques including Backward Differences Formulae (BDF) schemes with nonuniform time-step. Numerical results for a three-dimensional aeroelastic analysis of the Agard 445.6 wing are found to agree fairly well with available experimental data.

Key words: Euler equations, Arbitrary Lagrangian Eulerian, Dynamic meshes, Interface velocity, Geometric Conservation Law, Computational Aeroelasticity

1 Introduction

The interaction between a deformable structure and the fluid flowing around it may possibly lead to instability phenomena, which are potentially dangerous for the integrity of the structure itself [1]. For example, so-called flutter instabilities may present themselves in airplane flight and strongly limit the flight envelopes. In order to predict the behavior of complex aeronautic structures in connection with diverse aerodynamic phenomena such as for example transonic range shock waves, high fidelity Computational Fluid Dynamics (CFD) models must be adopted and coupled with the proper structural model of the airplane itself. This branch of continuum mechanics—usually termed Computational Aeroelasticity (CA)—is by definition highly interdisciplinary and since the past two decades it has been a very active and fruitful research area [2–4]. Indeed, computational aeroelasticity is not the only application in which the flow equations are to be solved in a domain whose shape continuously varies with time due to the movement of the domain boundaries, which in aeroelastic application are represented by the deformable surface of the structure itself.

In moving domains, the standard Eulerian formulation of the flow equations is to be dismissed in favor of the so-called Arbitrary Lagrangian–Eulerian (ALE) approach, in which the control volumes are allowed to change their shape in time [5]. Moreover, a suitable mesh movement strategy is to be adopted to preserve the quality of the computational grid as the underlying geometry changes its shape. Even in the case of prescribed boundary movement, namely, for uncoupled fluid-structure problems, the computational burden can be enormous.

To reduce the computational costs, it is therefore mandatory to develop time-accurate integration schemes for the flow equations in the ALE formulation. However, the issue of preserving the time-accuracy in the ALE extension of existing Eulerian codes has been investigated in details only recently [6–9]. In this respect, Nobile [6] showed that a naïve extension of fixed-grid methods to flows in moving domains does not preserve numerical accuracy and may possibly lead to numerical instabilities. Therefore, care is to be taken in both the evaluation of the local grid velocities and the definition of the geometric quantities, which are necessary to compute the fluxes across a given portion of the domain, which cannot be chosen independently [10]. Thomas and Lombard [11] proposed to supply the discrete statement of the problem with the additional constraint of reproducing a uniform flow field exactly. This condition, which is known as the Geometric Conservation Law (GCL), is demonstrated to be sufficient to achieve a first order time accuracy, but it is neither necessary nor sufficient for higher order accuracy [7]. At the same time, satisfying the GCL is a necessary and sufficient condition to guarantee the nonlinear stability of the integration scheme [12].

Following Mavriplis and Yang [13], it is possible to argue that the GCL condition is simply an additional conservation law, which should not be violated to reduce the source of errors, and must be investigated when the time integration scheme is evaluated. In this report the idea of considering a additional conservation laws for the geometric quantities is further expanded. However, differently from previous studies, the present analysis is not limited to the simple situation of uniform flow fields. To this purpose, a suitable Interface Velocity Consistency (IVC) condition is introduced which links the velocity at the interface between adjacent control volumes to the time derivatives of the volume themselves and results in an additional conservation law for these quantities, which is to be solved together with the flow equations. The time-accuracy of the resulting schemes is thoroughly investigated in both the two- and three-dimensional case for Backward Euler (BE) and the Backward Differences Formulae (BDF) time integration schemes, including variable time step discretizations, in two and three spatial dimensions.

The structure of the present report is as follows. In section 2, the flow solver is presented, both for static and dynamic grid applications. The edge-based node-centered finite volume scheme is written in a generalized form to allow for virtually any numerical flux approximation to be easily translated into the Arbitrary Lagrangian Eulerian (ALE) framework (section 2.3). In section 2.3.2, the procedure for imposing the boundary conditions for dynamic meshes in a weak form is thoroughly detailed. The Interface Velocity Condition is discussed in section 3, where the coupled system made of the flow equations together with the equations resulting from the IVC is given. In section 4, time integration techniques are discussed, for both constant and variable time step schemes. Section 5 gives some details on the elastic analogy strategy adopted to move the grid. Two- and three-dimensional application are solved to demonstrate the temporal accuracy of the proposed scheme in section 6. A fully three-dimensional aeroelastic analysis of the Agard 455.6 wing is performed in section 6.2, where numerical results are compared to available experimental data. Final remarks are then given in section 7, together with a summary of the obtained results.

2 Edge-based ALE solver for compressible inviscid flows

In the present section, the main features of the numerical scheme used in the present study are described. For simplicity, the discrete form of the Euler equations is presented first for a static triangulation, in which the shape and position of the grid elements do not vary with time. In section 2.3, the treatment is extended to dynamic meshes under the Arbitrary Lagrangian Eulerian approach.

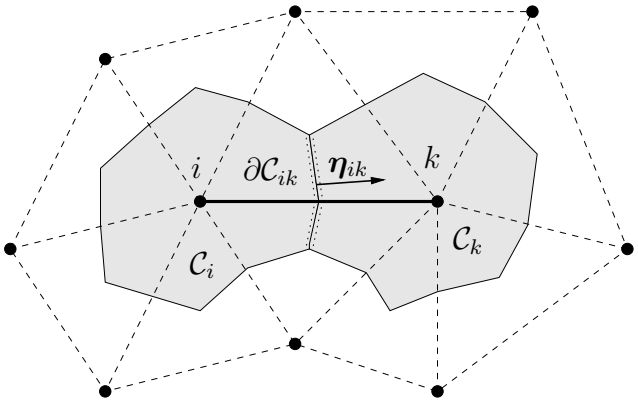


Fig. 1. Edge associated with the finite volume interface $\partial\mathcal{C}_{ik} = \partial\mathcal{C}_i \cap \partial\mathcal{C}_k$ and metric vector $\boldsymbol{\eta}_{ik}$ (integrated normal) in two spatial dimensions. The two shaded regions are the finite volumes $\mathcal{C}_i$ and $\mathcal{C}_k$; dashed lines indicate the underlying triangulation.

2.1 Governing equations

The model describing the dynamic of a compressible inviscid fluid in three spatial dimensions is provided by the well-known Euler equations, namely,

$$\frac{d}{dt}\int_{\mathcal{C}} \mathsf{u} + \oint_{\partial\mathcal{C}} \mathbf{f}(\mathsf{u}) \cdot \boldsymbol{n} = 0, \qquad \forall \mathcal{C} \subseteq \Omega, \tag{1}$$

where u, $\mathsf{u} = (\rho, \boldsymbol{m}, E^{\mathrm{t}})^{\mathrm{T}} \in \mathbb{R}^+ \times \mathbb{R}^4$, is the vector unknown of the density ρ, momentum vector $\boldsymbol{m}$ and total energy per unit volume E^{t}. The solution is sought for in the spatial domain $\Omega \in \mathbb{R}^3$, with boundary $\partial\Omega$ for all times $t \in \mathbb{R}^+$. System (1) is to be made complete by specifying suitable initial boundary conditions, see e.g. [14]. In (1), $\mathbf{f} = (\mathsf{f}_x, \mathsf{f}_y, \mathsf{f}_z)^{\mathrm{T}} \in \mathbb{R}^5 \times \mathbb{R}^3$ is the flux function defined as

$$\mathbf{f}(\mathsf{u}) = \left(\boldsymbol{m},\ \frac{\boldsymbol{m} \otimes \boldsymbol{m}}{\rho} + P(\mathsf{u})\ \boldsymbol{I},\ \left[E^{\mathrm{t}} + P(\mathsf{u})\right] \frac{\boldsymbol{m}}{\rho}\right)^{\mathrm{T}}, \tag{2}$$

where $\boldsymbol{I}$ is the 3×3 identity matrix and P is the pressure. For a polytropic ideal gas, namely, a perfect gas, one has $P(T, \rho) = RT\rho$ and $e(T) = RT/(\gamma - 1)$, with T temperature and $R = \mathcal{R}/M$, $\mathcal{R}$ universal gas constant and M molecular weight. The constant $\gamma = c_P/c_v$ is the ratio of the specific heat at constant pressure and volume, respectively. Hence,$P(\mathsf{u}) = (\gamma - 1)(E^{\mathrm{t}} - \frac{1}{2}|\boldsymbol{m}|^2/\rho)$ in all the present simulations, although the solver is capable of treating a more general, i.e., nonideal, thermodynamic model. The vector $\boldsymbol{n} = (n_x, n_y, n_z)^{\mathrm{T}}$ is the outward unit vector normal to the surface $\partial\mathcal{C}$ of $\mathcal{C}$; the scalar product in (1) is computed as $\mathbf{f}(\mathsf{u}) \cdot \boldsymbol{n} = \mathsf{f}_x n_x + \mathsf{f}_y n_y + \mathsf{f}_z n_x$.

2.2 Edge-based finite volume solver

The discrete counterpart of the Euler equation (1) is obtained by selecting a finite number of non overlapping volumes $\mathcal{C}_i \subset \Omega$, with boundary $\partial\mathcal{C}_i$, such that $\bigcup_i \mathcal{C}_i \equiv \Omega$. In the present section, both the shape and the location of the finite volumes $\mathcal{C}_i$ are assumed to not depend on time: the extension of the present method to dynamic meshes, in which $\mathcal{C}_i = \mathcal{C}_i(t)$, is dealt with in section 2.3. According to the node-centered finite volume approach considered here, each finite volume $\mathcal{C}_i$ surrounds a single node i of the triangulation of Ω, so that

$$V_i \frac{d\mathsf{u}_i}{dt} = -\oint_{\partial\mathcal{C}_i} \boldsymbol{n}_i \cdot \mathbf{f}(\mathsf{u}), \qquad \forall i \in \mathcal{K}, \tag{3}$$

where V_i is the volume of $\mathcal{C}_i$, $\mathcal{K}$ is the set of all nodes of the triangulation and $\boldsymbol{n}_i$ denotes the outward normal with respect to the volume $\mathcal{C}_i$, see Fig. 1. The unknown u is approximated over $\mathcal{C}_i$ by its average value $\mathsf{u}_i = \mathsf{u}_i(t)$. The right hand side of (3) is now rearranged to put into evidence the boundary contribution, namely,

$$\oint_{\partial\mathcal{C}_i} \boldsymbol{n}_i \cdot \mathbf{f}(\mathsf{u}) = \sum_{k \in \mathcal{K}_{i,\neq}} \int_{\partial\mathcal{C}_{ik}} \boldsymbol{n}_i \cdot \mathbf{f}(\mathsf{u}) + \int_{\partial\mathcal{C}_i \cap \partial\Omega} \boldsymbol{n}_i \cdot \mathbf{f}(\mathsf{u}), \tag{4}$$

where $\mathcal{K}_{i,\neq} = \{k \in \mathcal{K}, k \neq i | \partial\mathcal{C}_i \cap \partial\mathcal{C}_k \neq \emptyset\}$ is the set of the indexes k of the finite volumes $\mathcal{C}_k$ sharing a portion of their boundary with $\mathcal{C}_i$, $\mathcal{C}_i$ excluded. In the finite volume jargon, the set $\partial\mathcal{C}_i \cap \partial\mathcal{C}_k$ is often referred to as the cell interface between the volumes $\mathcal{C}_i$ and $\mathcal{C}_k$ (Fig. 1). A suitable integrated numerical flux $\mathbf{\Phi}_{ik} \in \mathbb{R}^5$, representing the flux across the cell interface $\partial\mathcal{C}_i \cap \partial\mathcal{C}_k$, is now introduced to give

$$\int_{\partial\mathcal{C}_{ik}} \boldsymbol{n}_i \cdot \mathbf{f}(\mathsf{u}) \simeq \mathbf{\Phi}_{ik}. \tag{5}$$

A second-order scheme is obtained from the following centered approximation [15],

$$\mathbf{\Phi}^{\mathrm{II}}_{ik} = \mathbf{\Phi}^{\mathrm{II}}(\mathsf{u}_i, \mathsf{u}_k, \boldsymbol{\eta}_{ik}) = \frac{\mathbf{f}(\mathsf{u}_i) + \mathbf{f}(\mathsf{u}_k)}{2} \cdot \boldsymbol{\eta}_{ik}, \tag{6}$$

where $\boldsymbol{\eta}_{ik}$ is the integrated outward normal, namely,

$$\boldsymbol{\eta}_{ik} = \int_{\partial\mathcal{C}_{ik}} \boldsymbol{n}_i.$$

Under the centered approximation (6), system (3) reads

$$V_i \frac{d\mathsf{u}_i}{dt} = \sum_{k \in \mathcal{K}_{i,\neq}} \frac{\mathbf{f}(\mathsf{u}_i) + \mathbf{f}(\mathsf{u}_k)}{2} \cdot \boldsymbol{\eta}_{ik} + \mathbf{\Phi}^{\partial}_i, \tag{7}$$

where $\mathbf{\Phi}^{\partial}_i \in \mathbb{R}^5$,

$$\mathbf{\Phi}^{\partial}_i = \int_{\partial\mathcal{C}_i \cap \partial\Omega} \boldsymbol{n}_i \cdot \mathbf{f}(\mathsf{u}), \tag{8}$$

represents the contribution of the boundary integral, to be detailed in section 2.2.1.

It is well known, see e.g. [15], that the use of the above second-order integrated numerical flux may lead to the appearance of spurious oscillations in advection dominated flows and in particular near discontinuities of the flow variables; in this case, the following first-order upwind approximation due to Roe [16], namely,

$$\Phi^{\mathrm{I}}_{ik} = \frac{\mathbf{f}(\mathsf{u}_i) + \mathbf{f}(\mathsf{u}_k)}{2} \cdot \boldsymbol{\eta}_{ik} - \frac{1}{2}|\widetilde{\mathsf{A}}|\,(\mathsf{u}_k - \mathsf{u}_i), \tag{9}$$

is to be preferred, where $\widetilde{\mathsf{A}}$, $\widetilde{\mathsf{A}} \in \mathbb{R}^5$, is the Roe matrix. $|\widetilde{\mathsf{A}}| = \widetilde{\mathsf{R}}|\widetilde{\Lambda}|\widetilde{\mathsf{L}}$, with $\widetilde{\mathsf{R}}$ and $\widetilde{\mathsf{L}}$ matrices of the right and left eigenvectors of $\widetilde{\mathsf{A}}$, respectively, and $|\widetilde{\Lambda} = \mathrm{diag}\,(|\widetilde{\lambda}_1|, \ldots, |\widetilde{\lambda}_5|)$ where $\widetilde{\lambda}_p, p = 1, \ldots, 5$, are the eigenvalues of $\widetilde{\mathsf{A}}$. The Roe matrix can be obtained as the Jacobian matrix $\mathbf{A} = \partial\mathbf{f}(\mathsf{u})/\partial\mathsf{u}$ of the flux function evaluated in the Roe-averaged state $\widetilde{\mathsf{u}}$, $\widetilde{\mathsf{u}} = \widetilde{\mathsf{u}}(\mathsf{u}_i, \mathsf{u}_k)$ times the integrated normal, namely, $\widetilde{\mathsf{A}} = \mathbf{A}(\widetilde{\mathsf{u}}) \cdot \boldsymbol{\eta}_{ik}$.

Following [17], a high-resolution expression for the integrated numerical flux is now obtained by resorting to the Total Variation Diminishing (TVD) approach, in which the second order approximation Φ^{II}_{ik} is replaced by its first order counterpart Φ^{I}_{ik} near flow discontinuities; the switch is controlled by a suitable flux limiter Υ, $\Upsilon = \mathrm{diag}\,(\Upsilon_1, \ldots, \Upsilon_5)$. The resulting high-resolution integrated numerical flux reads

$$\Phi^{\mathrm{HR}}_{ik} = \Phi^{\mathrm{I}}_{ik} + \Upsilon\left[\Phi^{\mathrm{II}}_{ik} - \Phi^{\mathrm{I}}_{ik}\right] = \Phi^{\mathrm{II}}_{ik} + \frac{1}{2}\widetilde{\mathsf{R}}|\widetilde{\Lambda}|\,(\widetilde{\mathsf{w}} - \widetilde{\mathsf{v}}), \tag{10}$$

where $\widetilde{\mathsf{v}} = \widetilde{\mathsf{L}}\,(\mathsf{u}_k - \mathsf{u}_i)$ and $\widetilde{\mathsf{w}} = \Upsilon\,\widetilde{\mathsf{L}}\,(\mathsf{u}_k - \mathsf{u}_i)$. By substituting the expression of the limiter of [17], the p-th component of the vector of the limited characteristic jumps $\widetilde{\mathsf{w}}$ reads

$$\widetilde{\mathsf{w}}_{(p)} = \frac{\widetilde{\mathsf{v}}_{(p)}|\widetilde{\mathsf{q}}_{(p)}| + |\widetilde{\mathsf{v}}_{(p)}|\widetilde{\mathsf{q}}_{(p)}}{|\widetilde{\mathsf{v}}_{(p)}| + |\widetilde{\mathsf{q}}_{(p)}| + \epsilon},$$

where ϵ is a small positive parameter introduced here to avoid division by zero of a zero quantity ($\epsilon = 10^{-12}$ here) and where the p-th component of the vector of "upwind" jumps $\widetilde{\mathsf{q}}$ is given by

$$\widetilde{\mathsf{q}}_{(p)} = \begin{cases} \widetilde{\mathsf{L}}_{(p)}(\mathsf{u}_{k^\star} - \mathsf{u}_k) & \text{if} \quad \widetilde{\lambda}_p > 0, \\ \widetilde{\mathsf{L}}_{(p)}(\mathsf{u}_i - \mathsf{u}_{i^\star}) & \text{if} \quad \widetilde{\lambda}_p \leq 0. \end{cases}$$

In the above definition, the nodes $i^\star$ and $k^\star$ are the extension nodes belonging to edge i-k of the triangulation and $\widetilde{\mathsf{L}}_{(p)}$ is the p-th row of matrix $\widetilde{\mathsf{L}}$. Note the the above high-resolution version of the scheme requires the definition of an extended edge data structure that includes also the extension nodes $i^\star$ and $k^\star$, that are needed in the evaluation of the limiter function Υ. Following [18], the extension nodes belong to the two edges best aligned with i-k, see Fig. 2.

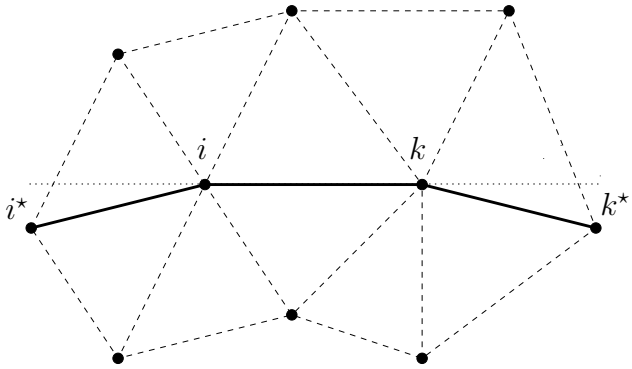


Fig. 2. Extended edge structure $i^\star$-i-k-$k^\star$ for high-resolution.

The computer implementation of the finite volume scheme described above is straightforward and very efficient, see e.g. [19]. All computations are performed only over the edges of the mesh: edges are present in one-, two- and three-dimensional grids and therefore the extension to different spatial dimension requires only few modifications to the code, that are limited to the definition of the vector unknown and to the associated flux function.

2.2.1 Boundary conditions

The evaluation of the boundary integral (8) is now detailed. According to the piecewise constant representation of the unknown in the finite volume framework, $\mathsf{u} = \mathsf{u}_i$ over the boundary portion $\partial\mathcal{C}_i \cap \Omega$ and therefore

$$\Phi_i^\partial = \int_{\partial\mathcal{C}_i\cap\partial\Omega} \mathbf{f}(\mathsf{u})\cdot\boldsymbol{n}_i \simeq \mathbf{f}(\mathsf{u}_i)\cdot\int_{\partial\mathcal{C}_i\cap\partial\Omega}\boldsymbol{n}_i = \mathbf{f}(\mathsf{u}_i)\cdot\boldsymbol{\xi}_i,$$

where the definition of the integrated boundary normal $\boldsymbol{\xi}_i$,

$$\boldsymbol{\xi}_i = \int_{\partial\mathcal{C}_i\cap\partial\Omega}\boldsymbol{n}_i$$

has been introduced.

Slip boundary conditions at solid surfaces and far-field boundary conditions are enforced here in a weak form, namely, by evaluating the boundary flux in a suitable boundary state variable $\overline{\mathsf{u}}$ as follows

$$\Phi_i^\partial = \mathbf{f}(\overline{\mathsf{u}}_i)\cdot\boldsymbol{\xi}_i, \tag{11}$$

where the boundary state $\overline{\mathsf{u}} = \overline{\mathsf{u}}(\mathsf{u}_i, \mathsf{a}_i)$, is a function of the boundary value u_i of the unknown and of the boundary data a_i. For example, slip boundary conditions are imposed by choosing

$$\begin{aligned}\overline{\mathsf{u}}_i &= \overline{\mathsf{u}}_\mathrm{s}(\mathsf{u}_i, \boldsymbol{n}_i)\\ &= \Big(\rho_i, \boldsymbol{m}_i - (\boldsymbol{m}_i\cdot\boldsymbol{n}_i)\boldsymbol{n}_i, E_i^\mathrm{t} - |\boldsymbol{m}_i\cdot\boldsymbol{n}_i|^2/(2\rho_i)\Big)^\mathrm{T},\end{aligned} \tag{12}$$

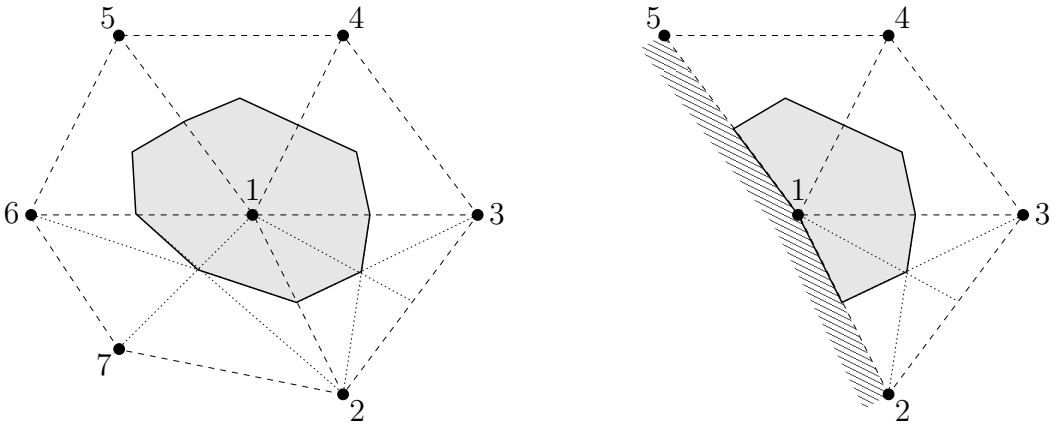


Fig. 3. Definition of the finite volume $\mathcal{C}_1$ (shaded region) from the underlying triangulation in two spatial dimensions. The nodes of the triangulation are indicated with the symbol •; the grid elements are indicated by dashed lines. a) Node 1 belongs to the interior of the computational domain Ω. b) Node 1 lies on the boundary of Ω, here represented by the segments 2–1 and 1–5. The medians of the triangle $(1, 2, 3)$ and of the quadrilateral element $(1, 6, 7, 2)$ are also indicated with dotted lines, cf. Fig. 3.

with $\boldsymbol{n}_i = \boldsymbol{\xi}_i/|\boldsymbol{\xi}_i|$ nodal averaged normal unit vector. In other words, $\overline{\mathsf{u}}_i$ is equal to u_i minus the component of the momentum vector normal to the surface.

The boundary state $\overline{\mathsf{u}}_i$ at the far-field is computed via characteristic reconstruction from the nodal state vector u_i and the far-field state u_∞ as follows [20]

$$\overline{\mathsf{u}}_i = \overline{\mathsf{u}}_{\mathrm{f}}(\mathsf{u}_i, \mathsf{u}_\infty, \boldsymbol{n}_i) = \mathsf{u}_i + \mathsf{R}_i^{\boldsymbol{n}} \mathsf{N}_i^{\boldsymbol{n}} \mathsf{L}_i^{\boldsymbol{n}} \Big[\mathsf{u}_\infty - \mathsf{u}_i\Big], \tag{13}$$

where $\mathsf{R}_i^{\boldsymbol{n}}$ and $\mathsf{L}_i^{\boldsymbol{n}}$ are the matrices of the right and left eigenvectors of $\boldsymbol{A}_i^{\boldsymbol{n}} = \mathbf{A}(\mathsf{u}_i) \cdot \boldsymbol{n}_i \in \mathbb{R}^5 \times \mathbb{R}^5$, with $\mathbf{A} = \partial \mathbf{f}(\mathsf{u})/\partial \mathsf{u}$, respectively, and where $\mathsf{N}_i^{\boldsymbol{n}}$ is the diagonal matrix $\mathrm{diag}\Big(\min\left(\lambda_p^{\boldsymbol{n}}, 0\right)\Big)$, $p = 1, \ldots, 5$, with $\lambda_p^{\boldsymbol{n}}$ the p-th eigenvalue of $\boldsymbol{A}_i^{\boldsymbol{n}}$. Note that the expression above simplifies to $\overline{\mathsf{u}}_i = \mathsf{u}_i$ for $\lambda_p^{\boldsymbol{n}} < 0 \ \forall p$, namely, for a supersonic outflow boundary, and to $\overline{\mathsf{u}}_i = \mathsf{u}_\infty$ for $\lambda_p^{\boldsymbol{n}} > 0 \ \forall p$ (supersonic inflow).

2.2.2 Node-centered finite volumes and metric vectors

In the present section, the expression for the volumes (area in two spatial dimensions) V_i and the metric quantities $\boldsymbol{\eta}_{ik}$ and $\boldsymbol{\xi}_i$, are given in terms of geometrical entities defined over the underlying triangulation of the domain Ω. The latter is possibly a so-called hybrid triangulation, namely, it could be made of elements of different types. For example, in two spatial dimensions, the grid elements can be either triangles or quadrilaterals, whereas in the more variegated three-dimensional case, one can have tetrahedra, prisms with triangular basis, pyramids or hexahedra.

For simplicity, the two-dimensional case is considered first. With reference to Fig. 3, the finite volume $\mathcal{C}_i$ is first split into the subsets $\mathcal{C}_{i,e} = \mathcal{C}_i \cap \Omega_e$, where Ω_e is the e-th element of the triangulation, so that $\mathcal{C}_i = \bigcup_{e\in\mathcal{E}_i} \mathcal{C}_i \cap \Omega_e$, with $\mathcal{E}_i$ the set of the elements sharing node i. The set $\mathcal{E}_i$ is often referred to as the element bubble of node i. Over each subset $\mathcal{C}_{i,e}$, the boundary $\partial\mathcal{C}_{i,e} = \partial\mathcal{C}_i \cap \Omega_e$ is made of the two segments connecting the center of gravity $\boldsymbol{x}_e$ of the element to the midpoints $\boldsymbol{x}_{ik}$ of the two edges from node i, as shown in Fig. 4. The elemental contribution $\boldsymbol{\eta}_{ik,e}$, see Fig. 4a, is therefore computed as

$$\boldsymbol{\eta}_{ik,e} = \int_{\partial\mathcal{C}_{ik,e}} \boldsymbol{n}_i = (\boldsymbol{x}_{ik} - \boldsymbol{x}_e) \times \hat{\boldsymbol{z}}, \tag{14}$$

where $\hat{\boldsymbol{z}}$ is the unit vector normal to the plane x-y. The metric vector $\boldsymbol{\eta}_{ik}$ is then computed as

$$\boldsymbol{\eta}_{ik} = \sum_{e\in\mathcal{E}_i} \boldsymbol{\eta}_{ik,e}. \tag{15}$$

Similarly, for the boundary metric vector $\boldsymbol{\xi}_i$, $\boldsymbol{\xi}_i = \sum_{e^\partial\in\mathcal{E}_i^\partial} \boldsymbol{\xi}_{i,e^\partial}$, where $\mathcal{E}_i^\partial$ is the set of the boundary elements having node i in common, one has

$$\boldsymbol{\xi}_{i,e^\partial} = \int_{\partial\mathcal{C}_{i,e}} \boldsymbol{n}_i = (\boldsymbol{x}_{e^\partial} - \boldsymbol{x}_i) \times \hat{\boldsymbol{z}}. \tag{16}$$

Note that in two spatial dimensions the boundary elements are the edges of the domain elements themselves, so that the center of gravity of the e^∂-th boundary element is the midpoint of the segment of the e-th element which lies on the boundary, as shown in Fig. 4b. The contribution $V_{i,e}$ of element e to the volume of $\mathcal{C}_i$ (area in the present two-dimensional case) is given by sum of the the area of the two triangles having as vertices node i, $\boldsymbol{x}_e$ and each of the two midpoints $\boldsymbol{x}_{ik}$, see Fig. 4, or, in terms of edge contributions

$$V_i = \sum_{e\in\mathcal{E}_i} V_{i,e} = \sum_{e\in\mathcal{E}_i} \sum_{k\in\mathcal{K}_{i,\neq}} V_{ik,e} \ ,$$

where

$$V_{ik,e} = \frac{1}{2}(\boldsymbol{x}_e - \boldsymbol{x}_i) \cdot \int_{\partial\mathcal{C}_{ik,e}} \boldsymbol{n}_i = \frac{1}{2}(\boldsymbol{x}_e - \boldsymbol{x}_i) \cdot \boldsymbol{\eta}_{ik,e} \, , \tag{17}$$

In three spatial dimensions, see Fig. 5, the boundary $\partial\mathcal{C}_i$ of $\mathcal{C}_i$ is the union of triangular faces which are defined as follows. As in the two-dimensional case, the definition is given for the elemental subsets $\mathcal{C}_{i,e} = \mathcal{C}_i \cap \Omega_e$. For each element e and for each edge ik belonging to e, a triangular facet is defined for each face f of the element which includes edge ik. These triangles have vertices $(\boldsymbol{x}_e, \boldsymbol{x}_f, \boldsymbol{x}_{ik})$, with $\boldsymbol{x}_f$ center of gravity of the f-th face. The integrated normal $\boldsymbol{\eta}_{ik}$ is computed as the sum of the contributions pertaining to each triangles $\partial\mathcal{C}_{ik,ef}$ as $\boldsymbol{\eta}_{ik} = \sum_{e\in\mathcal{E}_i\cap\mathcal{E}_k} \sum_{f\in\mathcal{F}_{ik,e}} \boldsymbol{\eta}_{ik,ef}$, where $\mathcal{F}_{ik,e}$ is the set of the faces of

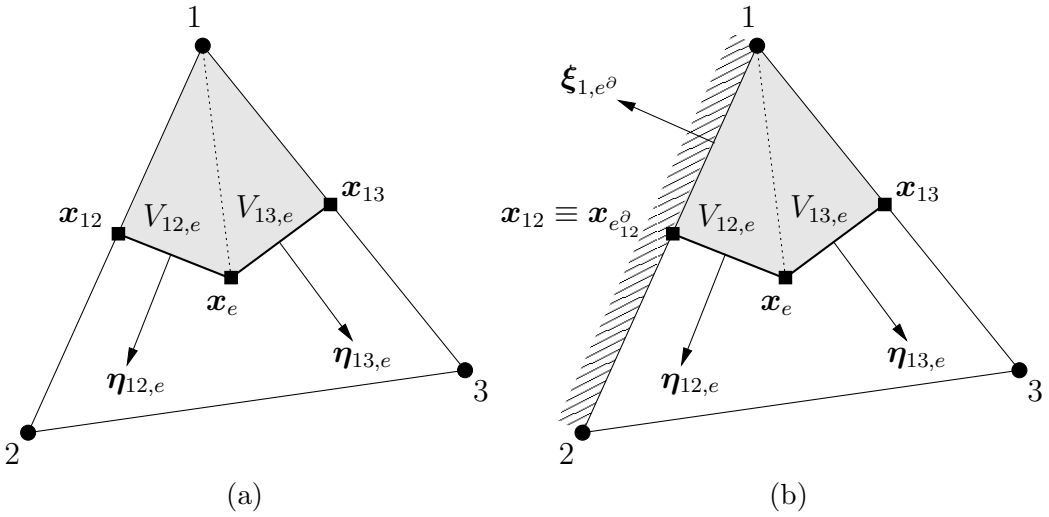


Fig. 4. Definition of the finite volume from the underlying triangulation in two spatial dimensions. The nodes of the triangulation are indicated with the symbol $\bullet$, the barycenter $\boldsymbol{x}_{ik}$ and $\boldsymbol{x}_e$ of the edges and of the element, respectively, are indicated with $\blacksquare$. a) The portion of the perimeter of the finite volume $\mathcal{C}_1$ pertaining to the triangular element e with nodes $(1,2,3)$ is made of the two segments $(\boldsymbol{x}_{12}, \boldsymbol{x}_e)$ and $(\boldsymbol{x}_e, \boldsymbol{x}_{13})$. The contribution to the area of the finite volume is given by the sum of the area of the two triangles $(\boldsymbol{x}_1, \boldsymbol{x}_{12}, \boldsymbol{x}_e)$ and $(\boldsymbol{x}_1, \boldsymbol{x}_{13}, \boldsymbol{x}_e)$. b) Elemental contributions for triangular element lying on the domain boundary.

element e sharing edge ik and

$$\boldsymbol{\eta}_{ik,ef} = \int_{\partial\mathcal{C}_{ik,ef}} \boldsymbol{n}_i = \frac{1}{2}(\boldsymbol{x}_f - \boldsymbol{x}_e) \times (\boldsymbol{x}_{ik} - \boldsymbol{x}_e). \tag{18}$$

The boundary metric vector $\boldsymbol{\xi}_i$ is computed as $\boldsymbol{\xi}_i = \sum_{e^\partial \in \mathcal{E}_i^\partial} \sum_{f^\partial \in \mathcal{F}_{i,e^\partial}} \boldsymbol{\xi}_{i,e^\partial f^\partial}$, where $\mathcal{F}^\partial_{i,e^\partial}$ is the set of faces of the boundary element e^∂ sharing node i and where

$$\boldsymbol{\xi}_{i,e^\partial f^\partial} = \int_{\partial\mathcal{C}_{i,ef}} \boldsymbol{n}_i = \frac{1}{2}(\boldsymbol{x}_{f^\partial} - \boldsymbol{x}_{e^\partial}) \times (\boldsymbol{x}_i - \boldsymbol{x}_{e^\partial}). \tag{19}$$

The contribution $V_{i,e}$ of element e to the volume of $\mathcal{C}_i$ is split again in contributions pertaining to each face and to each edge as follows, see Fig. 5,

$$V_i = \sum_{e \in \mathcal{E}_i} V_{i,e} = \sum_{e \in \mathcal{E}_i} \sum_{k \in \mathcal{K}_{i,\neq}} \sum_{f \in \mathcal{F}_{ik,e}} V_{ik,ef},$$

where $V_{ik,ef}$ is the volume of the tetrahedra $(\boldsymbol{x}_i, \boldsymbol{x}_e, \boldsymbol{x}_f, \boldsymbol{x}_{ik})$, namely,

$$V_{ik,ef} = \frac{1}{3}(\boldsymbol{x}_e - \boldsymbol{x}_i) \cdot \int_{\partial\mathcal{C}_{ik,ef}} \boldsymbol{n}_i = \frac{1}{3}(\boldsymbol{x}_e - \boldsymbol{x}_i) \cdot \boldsymbol{\eta}_{ik,ef}. \tag{20}$$

All the definitions given above—in both two and three spatial dimensions—are applicable to elements of any kind, i.e., to any given hybrid triangulation made of e.g. triangles and quadrilaterals (2D) and prisms, pyramids and hexahedra

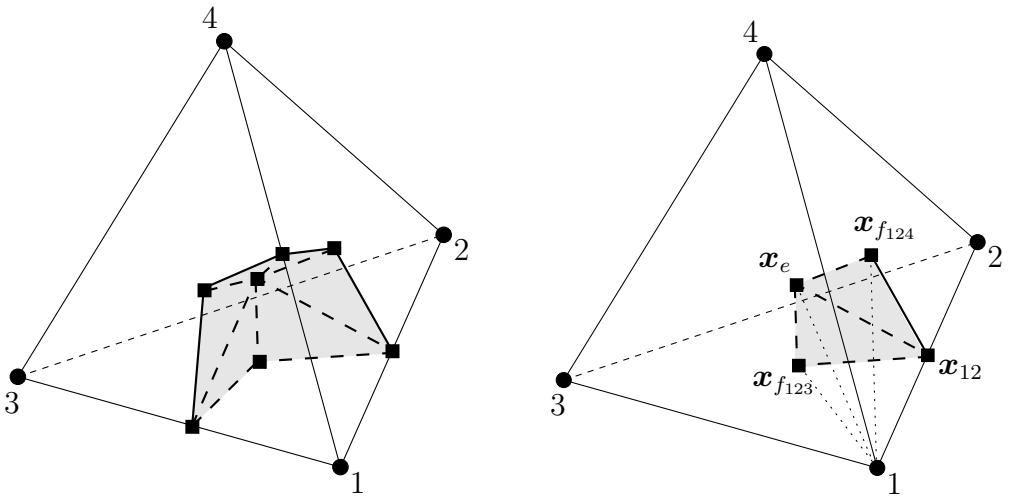


Fig. 5. Definition of the finite volume from the underlying triangulation in three spatial dimensions. The nodes of the triangulation are indicated with the symbol $\bullet$, the barycenter $\boldsymbol{x}_e$, $\boldsymbol{x}_f$ and $\boldsymbol{x}_g$ of the edges, of the element faces and of the element, respectively, are indicated with $\blacksquare$. a) The portion of the finite volume boundary $\partial\mathcal{C}_i$ pertaining to the tetrahedron e with nodes $(1, 2, 3, 4)$ is given by the six shaded triangles. The contribution to the volume is the sum of the volumes of the six tetrahedra obtained by connecting each of the six shaded triangles with vertex 1. Note that differently from the two-dimensional case, the shaded regions indicate the finite volume boundary and not its internal volume. b) Contribution of edge 12 only. The portion of the surface of the finite volume 1 pertaining to the tetrahedron 1234 and to the edge 12 is made of the two shaded triangles $(\boldsymbol{x}_{12}, \boldsymbol{x}_{f_{123}}, \boldsymbol{x}_e)$ and $(\boldsymbol{x}_{12}, \boldsymbol{x}_{f_{124}}, \boldsymbol{x}_e)$. The contribution to the volume of $\mathcal{C}_1$ is given by the sum of the volume of the two tetrahedron $(\boldsymbol{x}_1, \boldsymbol{x}_{12}, \boldsymbol{x}_{f_{123}}, \boldsymbol{x}_e)$ and $(\boldsymbol{x}_1, \boldsymbol{x}_{12}, \boldsymbol{x}_{f_{124}}, \boldsymbol{x}_e)$.

(3D) and guarantee that each finite volume is closed, namely,

$$\sum_{k \in \mathcal{K}_{i,\neq}} \boldsymbol{\eta}_{ik} = \mathbf{0} \qquad \text{on domain nodes,}$$

$$\sum_{k \in \mathcal{K}_{i,\neq}} \boldsymbol{\eta}_{ik} + \boldsymbol{\xi}_i = \mathbf{0} \qquad \text{on boundary nodes.}$$

To conclude, we notice in passing that Selmin [19] and Selmin and Formaggia [21] proved that a centered finite volume discretization built according to the above prescriptions is equivalent to a linear finite element approach in the case of two-dimensional grids of triangles or three-dimensional grids of tetrahedra, but for the boundary terms.

In the present section, the finite volume scheme is extended to deal with possibly moving grid elements, under the well-known Arbitrary Lagrangian Eulerian (ALE) approach [22,23]. To this purpose, the Euler equations (1) are now rewritten for a moving control volume $\mathcal{C} = \mathcal{C}(t)$ as follows

$$\frac{d}{dt}\int_{\mathcal{C}(t)} \mathsf{u} + \oint_{\partial\mathcal{C}(t)} \Big[\mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v}\Big]\cdot\boldsymbol{n} = \mathsf{0}, \qquad \forall \mathcal{C}(t) \subseteq \Omega(t), \tag{21}$$

where $-\mathsf{u}\,\boldsymbol{v}$ is the ALE additional flux due to the cell boundary movement, namely,

$$\mathsf{u}\,\boldsymbol{v} = \Big(\rho\boldsymbol{v}, \boldsymbol{m}\otimes\boldsymbol{v}, E^{\mathrm{t}}\boldsymbol{v}\Big)^{\mathrm{T}},$$

with $\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{x}, t)$ local velocity of the control volume. Note that the outward normal unit vector $\boldsymbol{n}$ appearing in (21) is a function of space and time as well.

An even more compact expression for (21) can be obtained by introducing the following 5×5 matrix,

$$\mathsf{T}(\boldsymbol{v}) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -v_x & 1 & 0 & 0 & 0 \\ -v_y & 0 & 1 & 0 & 0 \\ -v_z & 0 & 0 & 1 & 0 \\ \frac{1}{2}|\boldsymbol{v}|^2 & -v_x & -v_y & -v_z & 1 \end{pmatrix} \tag{22}$$

where v_x, v_y, v_z are the x, y and z component of the velocity $\boldsymbol{v}$, respectively, which allows to compute the state vector u in a local reference frame moving with velocity $\boldsymbol{v}$. Note that $\det(\mathsf{T}) = 1$ and therefore T^{-1} is always defined; moreover, $\mathsf{T}^{-1}(\boldsymbol{v}) = \mathsf{T}(-\boldsymbol{v})$. The following splitting of the ALE flux function $\mathbf{f}_{\mathrm{ALE}}(\mathsf{u}, \boldsymbol{v})$ is now introduced, cf. definition (2),

$$\mathbf{f}_{\mathrm{ALE}}(\mathsf{u}, \boldsymbol{v}) = \mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v} = \mathsf{u}\left[\frac{\boldsymbol{m}}{\rho} - \boldsymbol{v}\right] + \mathbf{p}(\mathsf{u}),$$

where $\mathbf{p}(\mathsf{u}) \in \mathbb{R}^5 \times \mathbb{R}^3$, $\mathbf{p}(\mathsf{u}) = (\mathbf{0}, P(\mathsf{u})\boldsymbol{I}, P(\mathsf{u})\,\boldsymbol{m}/\rho)^{\mathrm{T}}$, is the contribution to the flux function (2) due to the action of the pressure only. According to the assumption of local thermodynamic equilibrium, the pressure P does not depend on the local value of the velocity and therefore $P(\mathsf{u}) \equiv P(\mathsf{T}(\boldsymbol{v})\,\mathsf{u}), \forall \boldsymbol{v} \in \mathbb{R}^3$. By recalling definition (22) of the transformation matrix T, one easily obtains $\mathbf{p}(\mathsf{u}) = \mathsf{T}^{-1}(\boldsymbol{v})\,\mathbf{p}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u}),\ \forall \boldsymbol{v} \in \mathbb{R}^3$. Then, by multiplying the term $\mathsf{u}\,[\frac{\boldsymbol{m}}{\rho} - \boldsymbol{v}]$ of the ALE flux function by the identity matrix $\mathsf{T}^{-1}\mathsf{T}$ and by substituting the

above relation, one immediately obtains

$$\mathbf{f}_{\mathrm{ALE}}(\mathsf{u}, \boldsymbol{v}) = \mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v} = \mathsf{T}^{-1}(\boldsymbol{v})\,\mathbf{f}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u}),$$

namely, the ALE flux function is $\mathsf{T}^{-1}(\boldsymbol{v})$ times the flux (2) evaluated at the state u in the local reference frame moving at the velocity $\boldsymbol{v}$. The ALE formulation of the Euler equations (21) finally reads

$$\frac{d}{dt}\int_{\mathcal{C}(t)} \mathsf{u} + \oint_{\partial\mathcal{C}(t)} \mathsf{T}^{-1}(\boldsymbol{v})\,\mathbf{f}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u})\cdot\boldsymbol{n} = 0, \qquad \forall \mathcal{C}(t) \subseteq \Omega(t). \tag{23}$$

Note that matrix T reduces to the 5×5 identity matrix at $\boldsymbol{v} = 0$, namely, when the control volume is not moving, and therefore $\mathbf{f}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u}) \equiv \mathbf{f}(\mathsf{u})$ and the Euler equations (1) for static control volumes are recovered. To conclude, it is remarkable that the ALE formulation is obtained as a two step transformation from the Eulerian (fixed volume) point of view to the Lagrangian one and then back into the absolute (Eulerian) reference frame via the reverse transformation $\mathsf{T}^{-1}(\boldsymbol{v})$.

2.3.1 Finite volume discretization and Interface Velocity Consistency

The finite volume discretization of (23), is now derived. First, the surface integral in (23) is split into domain and boundary contributions as follows

$$\begin{aligned}\oint_{\partial\mathcal{C}(t)} \mathsf{T}^{-1}(\boldsymbol{v})\,\mathbf{f}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u})\cdot\boldsymbol{n} = \sum_{k\in\mathcal{K}_{i,\neq}} \int_{\partial\mathcal{C}_{ik}} \mathsf{T}^{-1}(\boldsymbol{v})\,\mathbf{f}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u})\cdot\boldsymbol{n}_i \\ + \int_{\partial\mathcal{C}_i\cap\partial\Omega} \mathsf{T}^{-1}(\boldsymbol{v})\,\mathbf{f}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u})\cdot\boldsymbol{n}_i.\end{aligned}$$

Considering now a centered approximation at the cell interfaces, the domain contributions read

$$\begin{aligned}\int_{\partial\mathcal{C}_{ik}} \mathsf{T}^{-1}(\boldsymbol{v})\,\mathbf{f}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u})\cdot\boldsymbol{n}_i &= \int_{\partial\mathcal{C}_{ik}} \Big[\mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v}\Big]\cdot\boldsymbol{n}_i\,, \\ &\simeq \frac{\mathbf{f}(\mathsf{u}_i)+\mathbf{f}(\mathsf{u}_k)}{2}\cdot\int_{\partial\mathcal{C}_{ik}}\boldsymbol{n}_i - \frac{\mathsf{u}_i+\mathsf{u}_k}{2}\int_{\partial\mathcal{C}_{ik}}\boldsymbol{v}\cdot\boldsymbol{n}_i\,, \\ &= \left[\frac{\mathbf{f}(\mathsf{u}_i)+\mathbf{f}(\mathsf{u}_k)}{2} - \frac{\mathsf{u}_i+\mathsf{u}_k}{2}\,\boldsymbol{\nu}_{ik}\right]\cdot\boldsymbol{\eta}_{ik}\,,\end{aligned}$$

where $\boldsymbol{\nu}_{ik}$ is the *average interface velocity*, which satisfies the following *interface velocity consistency* (IVC) condition

$$\boldsymbol{\nu}_{ik}\cdot\boldsymbol{\eta}_{ik} = \int_{\partial\mathcal{C}_{ik}}\boldsymbol{v}\cdot\boldsymbol{n}_i. \tag{24}$$

After rearranging the relations above, one finally obtains

$$\int_{\partial\mathcal{C}_{ik}} \mathsf{T}^{-1}(\boldsymbol{v})\,\mathbf{f}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u})\cdot\boldsymbol{n}_i \simeq \frac{1}{2}\left[\mathbf{f}(\mathsf{u}_i)+\mathbf{f}(\mathsf{u}_k)-(\mathsf{u}_i+\mathsf{u}_k)\boldsymbol{\nu}_{ik}\right]\cdot\boldsymbol{\eta}_{ik}\,,$$
$$= \frac{1}{2}\mathsf{T}_{ik}^{-1}\left[\mathbf{f}(\mathsf{T}_{ik}\mathsf{u}_i)+\mathbf{f}(\mathsf{T}_{ik}\mathsf{u}_k)\right]\cdot\boldsymbol{\eta}_{ik}\,,$$

where $\mathsf{T}_{ik}=\mathsf{T}(\boldsymbol{\nu}_{ik})$. Correspondingly, the boundary contribution to the surface integral reads

$$\begin{aligned}\int_{\partial\mathcal{C}_i\cap\partial\Omega} \mathsf{T}^{-1}(\boldsymbol{v})\,\mathbf{f}(\mathsf{T}(\boldsymbol{v})\,\mathsf{u})\cdot\boldsymbol{n}_i &= \int_{\partial\mathcal{C}_i\cap\partial\Omega}\Big[\mathbf{f}(\mathsf{u})-\mathsf{u}\,\boldsymbol{v}\Big]\cdot\boldsymbol{n}_i\,,\\ &\simeq \mathbf{f}(\mathsf{u}_i)\cdot\int_{\partial\mathcal{C}_{ik}}\boldsymbol{n}_i-\mathsf{u}_i\int_{\partial\mathcal{C}_{ik}}\boldsymbol{v}\cdot\boldsymbol{n}_i\,,\\ &= \left[\mathbf{f}(\mathsf{u}_i)-\mathsf{u}_i\,\boldsymbol{\nu}_{ik}\right]\cdot\boldsymbol{\xi}_i = \mathsf{T}_i^{-1}\,\mathbf{f}(\mathsf{T}_i\,\mathsf{u}_i)\cdot\boldsymbol{\xi}_i\,,\end{aligned} \tag{25}$$

where $\mathsf{T}_i=\mathsf{T}(\boldsymbol{\nu}_i)$, with $\boldsymbol{\nu}_i$ average interface velocity of the i-th boundary node, namely,

$$\boldsymbol{\nu}_i\cdot\boldsymbol{\xi}_i=\int_{\partial\mathcal{C}_i\cap\partial\Omega}\boldsymbol{v}\cdot\boldsymbol{n}_i\,, \tag{26}$$

Under the centered finite volume approximation considered so far, the spatially discrete form of (23) reads

$$\frac{d}{dt}\left[V_i\,\mathsf{u}_i\right]+\sum_{k\in\mathcal{K}_{i,\neq}}\mathsf{T}_{ik}^{-1}\frac{\mathbf{f}(\mathsf{T}_{ik}\mathsf{u}_i)+\mathbf{f}(\mathsf{T}_{ik}\mathsf{u}_k)}{2}\cdot\boldsymbol{\eta}_{ik}+\mathsf{T}_i^{-1}\,\mathbf{f}(\mathsf{T}_i\,\mathsf{u}_i)\cdot\boldsymbol{\xi}_i=\mathsf{0}. \tag{27}$$

For a general, namely, not centered approximation of the numerical fluxes, one has

$$\frac{d}{dt}\left[V_i\,\mathsf{u}_i\right]+\sum_{k\in\mathcal{K}_{i,\neq}}\mathsf{T}_{ik}^{-1}\,\Phi(\mathsf{T}_{ik}\,\mathsf{u}_i,\mathsf{T}_{ik}\,\mathsf{u}_k,\boldsymbol{\eta}_{ik})+\mathsf{T}_i^{-1}\Phi^{\partial}(\mathsf{T}_i\,\mathsf{u}_i,\boldsymbol{\xi}_i)=\mathsf{0}, \tag{28}$$

The above relations are formally equivalent to those obtained in the previous section in the case of static meshes; the integrated numerical flux is evaluated here in a transformed state $\mathsf{T}(\boldsymbol{v})\,\mathsf{u}$ and then transformed back into the absolute reference frame via the reverse transformation $\mathsf{T}^{-1}(\boldsymbol{v})$. Thanks to the general form of the ALE flux function presented here, the inclusion of different approximations for the flux function already available for static meshes is very simple. Note also that the following relation exists between the Roe intermediate state $\widetilde{\mathsf{u}}$ computed using the transformed variable $\mathsf{T}(\boldsymbol{v})\,\mathsf{u}$ and that obtained from the state vector u,

$$\widetilde{\mathsf{u}}(\mathsf{T}_{ik}\,\mathsf{u}_i,\mathsf{T}_{ik}\,\mathsf{u}_k)=\mathsf{T}_{ik}\,\widetilde{\mathsf{u}}(\mathsf{u}_i,\mathsf{u}_k)$$

thanks to Galilean invariance of the Euler equation, see e.g. [14]. Naturally, due care is exercised to avoid computations of null terms due to T sparsity in the coded perations.

2.3.2 Boundary conditions

The evaluation of the boundary integral (25) and the imposition of slip and far-field boundary conditions for system (23) in the case of dynamic meshes are now discussed.

The computation of the boundary value u_i satisfying the slip boundary condition is easily achieved by recalling that, in the case of moving boundary, the slip conditions requires that the component of the fluid velocity normal to the boundary is equal to the boundary velocity along the normal direction. In other words, the normal component of the fluid velocity is to be zero when evaluated in a reference frame that is moving at the velocity $\boldsymbol{v}$ of the boundary itself. Therefore, slip boundary conditions are imposed via the same function $\overline{\mathsf{u}}_{\mathrm{s}}$ defined in (12) as $\overline{\mathsf{u}}_i = \overline{\mathsf{u}}_{\mathrm{s}}(\mathsf{T}_i\,\mathsf{u}_i, \boldsymbol{n}_i)$. Similarly, the value of $\overline{\mathsf{u}}$ at the far-field boundary is immediately computed as $\overline{\mathsf{u}}_i = \overline{\mathsf{u}}_{\mathrm{f}}(\mathsf{T}_i\,\mathsf{u}_i, \mathsf{T}_i\,\mathsf{u}_\infty, \boldsymbol{n}_i)$.

3 The Interface Velocity Consistency condition

The introduction of suitable average interface velocities defined in (24) and (26), namely,

$$\boldsymbol{\nu}_{ik}(t) \cdot \boldsymbol{\eta}_{ik}(t) = \int_{\partial\mathcal{C}_{ik}(t)} \boldsymbol{v}(\boldsymbol{s},t) \cdot \boldsymbol{n}_i(\boldsymbol{s},t). \tag{29}$$

and

$$\boldsymbol{\nu}_i(t) \cdot \boldsymbol{\xi}_i(t) = \int_{\partial\mathcal{C}_i(t)\cap\partial\Omega} \boldsymbol{v}(\boldsymbol{s},t) \cdot \boldsymbol{n}_i(\boldsymbol{s},t)\,, \tag{30}$$

where $\boldsymbol{s}$ is the boundary coordinate and where $\boldsymbol{\eta}_{ik}(t)$ and $\boldsymbol{\xi}_i(t)$ are defined in section 2.2.2, is key to the finite volume discretization presented above. In the present section, an explicit expression of the average interface velocities $\boldsymbol{\nu}_{ik}(t)$ and $\boldsymbol{\nu}_i(t)$ is derived in both two- and three-dimensional cases. The coupled system consiting of the flow equations and of the IVC condition is given at the end of the present section.

3.1 Two-dimensional case

In two-dimensional problems, since grid edges remain rectilinear during the grid movement, one easily obtains

$$\int_{\partial\mathcal{C}_{ik}} \boldsymbol{v}\cdot\boldsymbol{n} = \sum_{e\in\mathcal{E}_i\cap\mathcal{E}_k} \int_{\partial\mathcal{C}_{ik,e}} \boldsymbol{v}\cdot\boldsymbol{n}_i = \sum_{e\in\mathcal{E}_i\cap\mathcal{E}_k} \frac{\boldsymbol{v}_i+\boldsymbol{v}_e}{2} \cdot \int_{\partial\mathcal{C}_{ik,e}} \boldsymbol{n}_i = \sum_{e\in\mathcal{E}_i\cap\mathcal{E}_k} \frac{\boldsymbol{v}_i+\boldsymbol{v}_e}{2} \cdot \boldsymbol{\eta}_{ik,e}\,,$$

where $\boldsymbol{v}_i$ and $\boldsymbol{v}_e$ are the velocities of node i and of the barycenter of the e-th element, respectively, and where the fact that the interface normal $\boldsymbol{n}$ is

constant on the interface portion $\mathcal{C}_{ik,e}$ has been accounted for. Note that the definition (29) imposes a constraint on the component of $\boldsymbol{\nu}_{ik}$ parallel to $\boldsymbol{\eta}_{ik}$ only. Hence, $\boldsymbol{\nu}_{ik}$ is chosen to be parallel to $\boldsymbol{\eta}_{ik}$ itself and reads

$$\boldsymbol{\nu}_{ik}(t) = \frac{\boldsymbol{\eta}_{ik}(t)}{|\boldsymbol{\eta}_{ik}(t)|} \sum_{e \in \mathcal{E}_i \cap \mathcal{E}_k} \frac{\boldsymbol{v}_i(t) + \boldsymbol{v}_e(t)}{2} \cdot \boldsymbol{\eta}_{ik,e}(t). \tag{31}$$

Similarly, the average interface velocity $\boldsymbol{\nu}_i(t)$ associated with a boundary node i reads

$$\boldsymbol{\nu}_i(t) = \frac{\boldsymbol{\xi}_i(t)}{|\boldsymbol{\xi}_i(t)|} \sum_{e^\partial \in \mathcal{E}_i^\partial} \frac{\boldsymbol{v}_i(t) + \boldsymbol{v}_e^\partial(t)}{2} \cdot \boldsymbol{\xi}_{i,e^\partial}(t), \tag{32}$$

where $\boldsymbol{v}_e^\partial$ is the velocity of the barycenter of the boundary element (segment in 2D) e^∂.

3.2 Three-dimensional case

Moving now to the three-dimensional case, by definition (see section 2.2.2), each portion $\mathcal{C}_{ik,ef}$ of the finite volume boundary is a triangle, thus it always remains plane during movements, and therefore

$$\begin{aligned} \int_{\partial \mathcal{C}_{ik}} \boldsymbol{v} \cdot \boldsymbol{n}_i = \sum_{e \in \mathcal{E}_i \cap \mathcal{E}_k} \sum_{f \in \mathcal{F}_{ik,e}} \int_{\partial \mathcal{C}_{ik,ef}} \boldsymbol{v} \cdot \boldsymbol{n}_i &= \sum_{e \in \mathcal{E}_i \cap \mathcal{E}_k} \sum_{f \in \mathcal{F}_{ik,e}} \frac{\boldsymbol{v}_i + \boldsymbol{v}_e + \boldsymbol{v}_f}{3} \cdot \int_{\partial \mathcal{C}_{ik,ef}} \boldsymbol{n} \\ &= \sum_{e \in \mathcal{E}_i \cap \mathcal{E}_k} \sum_{f \in \mathcal{F}_{ik,e}} \frac{\boldsymbol{v}_i + \boldsymbol{v}_e + \boldsymbol{v}_f}{3} \cdot \boldsymbol{\eta}_{ik,ef}\,, \end{aligned}$$

where $\boldsymbol{v}_f$ is the velocity of barycenter of the f-th element face. Therefore, $\boldsymbol{\nu}_i k(t)$ reads

$$\boldsymbol{\nu}_{ik}(t) = \frac{\boldsymbol{\eta}_{ik}(t)}{|\boldsymbol{\eta}_{ik}(t)|} \sum_{e \in \mathcal{E}_i \cap \mathcal{E}_k} \sum_{f \in \mathcal{F}_{ik,e}} \frac{\boldsymbol{v}_i(t) + \boldsymbol{v}_e(t) + \boldsymbol{v}_f(t)}{3} \cdot \boldsymbol{\eta}_{ik,e}(t), \tag{33}$$

and the average interface velocity $\boldsymbol{\nu}_i(t)$ associated with a boundary node i reads

$$\boldsymbol{\nu}_i(t) = \frac{\boldsymbol{\xi}_i(t)}{|\boldsymbol{\xi}_i(t)|} \sum_{e^\partial \in \mathcal{E}_i^\partial} \sum_{f^\partial \in \mathcal{F}_{ik,e^\partial}^\partial} \frac{\boldsymbol{v}_i(t) + \boldsymbol{v}_{e^\partial}(t) + \boldsymbol{v}_{f^\partial}(t)}{3} \cdot \boldsymbol{\xi}_{i,e^\partial f^\partial}(t) \tag{34}$$

3.3 The flow equations and the IVC condition

The expressions of the average interface and boundary velocity satisfying the IVC conditions derived in the previous section are now used to complete system (27) expressing the conservation of mass, momentum and total energy in

the ALE framework, to give

$$\begin{cases} \dfrac{d}{dt}[V_i\,\mathsf{u}_i] = \displaystyle\sum_{k\in\mathcal{K}_{i,\neq}} \Psi(\mathsf{u}_i,\mathsf{u}_k,\boldsymbol{\nu}_{ik},\boldsymbol{\eta}_{ik}), + \Psi^{\partial}(\mathsf{u}_i,\boldsymbol{\nu}_i,\boldsymbol{\xi}_i), \\ \boldsymbol{\nu}_{ik}\cdot\boldsymbol{\eta}_{ik} = \displaystyle\int_{\partial\mathcal{C}_{ik}} \boldsymbol{v}\cdot\boldsymbol{n}_i, \\ \boldsymbol{\nu}_i\cdot\boldsymbol{\xi}_i = \displaystyle\int_{\partial\mathcal{C}_i\cap\partial\Omega} \boldsymbol{v}\cdot\boldsymbol{n}_i, \end{cases} \tag{35}$$

where the shorthands $\Psi(\mathsf{u}_i,\mathsf{u}_k,\boldsymbol{\nu}_{ik},\boldsymbol{\eta}_{ik}) = -\mathsf{T}^{-1}(\boldsymbol{\nu}_{ik})\,\Phi(\mathsf{T}(\boldsymbol{\nu}_{ik})\,\mathsf{u}_i,\mathsf{T}(\boldsymbol{\nu}_{ik})\,\mathsf{u}_k,\boldsymbol{\eta}_{ik})$ and $\Psi^{\partial}((\mathsf{u}_i,\boldsymbol{\nu}_i,\boldsymbol{\xi}_i) = -\mathsf{T}^{-1}(\boldsymbol{\nu}_i)\Phi^{\partial}(\mathsf{T}(\boldsymbol{\nu}_i)\,\mathsf{u}_i,\boldsymbol{\xi}_i)$ have been introduced. System (35) is a system of Differential Algebraic Equations (DAE) consisting in $N_{\text{dof}} = N\times(d+2)$ Ordinary Differential Equations (ODE), with N_{dof} total number of degrees of freedom, N total number of grid points and $d = 1,2,3$ number of spatial dimensions, and $N_{ik} + N_{i,\partial}$ algebraic relations, with N_{ik} and $N_{i,\partial}$ number of grid edges and of boundary nodes, respectively. Note that the algebraic equations for the interface velocities $\boldsymbol{\nu}_{ik}$ and $\boldsymbol{\nu}_i$ are not coupled to the ODE fluid dynamics subsystem, provided that the coordinates of the grid nodes are known. As a consequence, in this case the system (35) can be reduced to simple ODEs by substitution. This is not the case for example in fluid-structure interaction problems in which the node velocity is obtained by solving the structural problem with loads from the flow field, thus resulting in a coupled system representing the structural, the fluid dynamics and the mesh deformation problem as well. Moreover, the coupled form of system (35) is preferred here to stress the existence of a consistency constraint on the interface velocities which leads to an additional system of algebraic (or ODE, cf. system (38)) to be solved together with the ODE system describing the flow dynamics.

The interface velocities are linked to the derivative of the volume V_i in time by the following relation

$$\frac{dV_i}{dt} = \oint_{\partial\mathcal{C}_i} \boldsymbol{v}\cdot\boldsymbol{n}_i = \sum_{k\in\mathcal{K}_{i,\neq}} \int_{\partial\mathcal{C}_{ik}} \boldsymbol{v}\cdot\boldsymbol{n}_i + \int_{\partial\mathcal{C}_i\cap\partial\Omega} \boldsymbol{v}\cdot\boldsymbol{n}_i = \sum_{k\in\mathcal{K}_{i,\neq}} \frac{dV_{i,ik}}{dt} + \frac{dV_{i,\partial}}{dt},$$

where the following definitions have been introduced

$$\frac{dV_{i,ik}}{dt} = \int_{\partial\mathcal{C}_{ik}} \boldsymbol{v}\cdot\boldsymbol{n}_i = \boldsymbol{\nu}_{ik}\cdot\boldsymbol{\eta}_{ik} \tag{36}$$

$$\frac{dV_{i,\partial}}{dt} = \int_{\partial\mathcal{C}_i\cap\partial\Omega} \boldsymbol{v}\cdot\boldsymbol{n}_i = \boldsymbol{\nu}_i\cdot\boldsymbol{\xi}_i \tag{37}$$

which allows to recast system (35) as a system of $N_{\text{dof}} + N_{ik} + N_{i,\partial}$ ODE as

follows

$$\begin{cases} \dfrac{d}{dt}[V_i\,\mathsf{u}_i] = \displaystyle\sum_{k\in\mathcal{K}_{i,\neq}} \Psi(\mathsf{u}_i,\mathsf{u}_k,\boldsymbol{\nu}_{ik},\boldsymbol{\eta}_{ik}) + \Psi^{\partial}(\mathsf{u}_i,\boldsymbol{\nu}_i,\boldsymbol{\xi}_i), \\ \dfrac{dV_{i,ik}}{dt} = \boldsymbol{\nu}_{ik}\cdot\boldsymbol{\eta}_{ik}\,, \\ \dfrac{dV_{i,\partial}}{dt} = \boldsymbol{\nu}_i\cdot\boldsymbol{\xi}_i\,, \end{cases} \tag{38}$$

which can be integrated in time by means of standard integration techniques, as detailed in section 4. Note that the quantities defined in (36) and (37), which sum up to the derivative in time of the volume V_i, are not related to the edge contributions V_{ik} defined in section 2.2.2. Differently from system (35), the grid velocity $\boldsymbol{v}$ is not longer present in system (38) and therefore no consistency problem arises in the fulfillment of the volume conservation constraint

$$\frac{dV_i}{dt} = \oint_{\partial\mathcal{C}_i} \boldsymbol{v}\cdot\boldsymbol{n}_i,$$

as it is often the case in other approaches due to the different discrete representation of the finite volume and of the grid velocities.

4 Time integration

In the present section, time integration of system (38) is detailed according to the Backward Euler (BE) and the Backward Differences Formulae (BDF) schemes. The method is also extended to a general multi-step scheme in section 4.2. The relationship between the IVC and the GCL conditions is discussed at the end of the present section.

4.1 Backward Euler scheme

The Backward Euler (BE) time discrete counterpart of system (38) is easily obtained as

$$\begin{cases} V_i^{n+1}\,\mathsf{u}_i^{n+1} - V_i^{n}\,\mathsf{u}_i^{n} = \Big[\displaystyle\sum_{k\in\mathcal{K}_{i,\neq}} \Psi(\mathsf{u}_i^{n+1},\mathsf{u}_k^{n+1},\boldsymbol{\nu}_{ik}^{n+1},\boldsymbol{\eta}_{ik}^{n+1}) \\ \qquad\qquad + \Psi^{\partial}(\mathsf{u}_i^{n+1},\boldsymbol{\nu}_i^{n+1},\boldsymbol{\xi}_i^{n+1})\Big]\,\Delta t^n \\ V_{i,ik}^{n+1} - V_{i,ik}^{n} = \boldsymbol{\nu}_{ik}^{n+1}\cdot\boldsymbol{\eta}_{ik}^{n+1}\,\Delta t^n \\ V_{i,\partial}^{n+1} - V_{i,\partial}^{n} = \boldsymbol{\nu}_{i}^{n+1}\cdot\boldsymbol{\xi}_{i}^{n+1}\,\Delta t^n \end{cases} \tag{39}$$

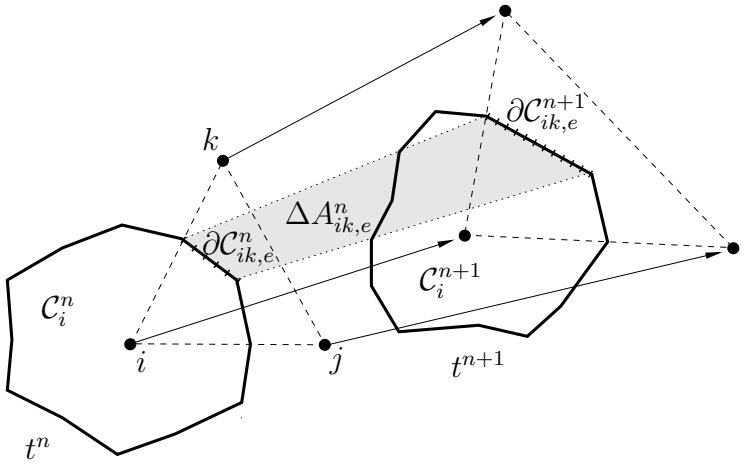


Fig. 6. Area swept by the interface $\partial\mathcal{C}_{ik,e}$ pertaining to edge i-k and to the e-th element from time level t^n to t^{n+1}.

where all quantities are assumed to be known at time level n and the grid-dependent quantities $V_i^{n+1}, V_{i,ik}^{n+1}, V_{i,\partial}^{n+1}, \boldsymbol{\eta}_{ik}^{n+1}$ and $\boldsymbol{\xi}_i^{n+1}$ are computed from the (known) positions of the grid nodes at time level $n+1$. System (39) is therefore a nonlinear system of $N_{\mathrm{dof}} + N_{ik} + N_{i,\partial}$ DOE in the N_{dof} flow variables u_i, and the $N_{ik} + N_{i,\partial}$ average interface velocities $\boldsymbol{\nu}_{ik}$ and $\boldsymbol{\nu}_i$. Note that the following relation holds

$$V_i^{n+1} - V_i^n = \sum_{k \in \mathcal{K}_{i,\neq}} \left[V_{i,ik}^{n+1} - V_{i,ik}^n \right] + V_{i,\partial}^{n+1} - V_{i,\partial}^n,$$

where the expressions of the contributions $V_{i,ik}^{n+1} - V_{i,ik}^n$ and $V_{i,\partial}^{n+1} - V_{i,\partial}^n$ in terms of the coordinates of the grid nodes still remains to be derived. The nonlinear system for the fluid variables u at time level $n+1$ is solved here by means of a modified Newton method, in which the Jacobian of the integrated flux function is approximated by that of the first-order scheme (9), and by resorting to a dual time-stepping technique [24], to improve the conditioning number of the Jacobian matrix.

4.1.1 Two spatial dimensions

In two spatial dimensions, the difference $V_{i,ik}^{n+1} - V_{i,ik}^n$ can be expressed in terms of elemental contributions as follows

$$\Delta V_{i,ik}^n = V_{i,ik}^{n+1} - V_{i,ik}^n = \sum_{e \in \mathcal{E}_i \cap \mathcal{E}_k} \Delta V_{i,ik,e}^n, \tag{40}$$

where, with reference to Fig. 6 and 7, $\Delta V_{i,ik,e}^n$ is the area swept by the portion $\partial\mathcal{C}_{ik,e}$ of the finite volume interface $\partial\mathcal{C}_i$ pertaining to the edge i-k and to the

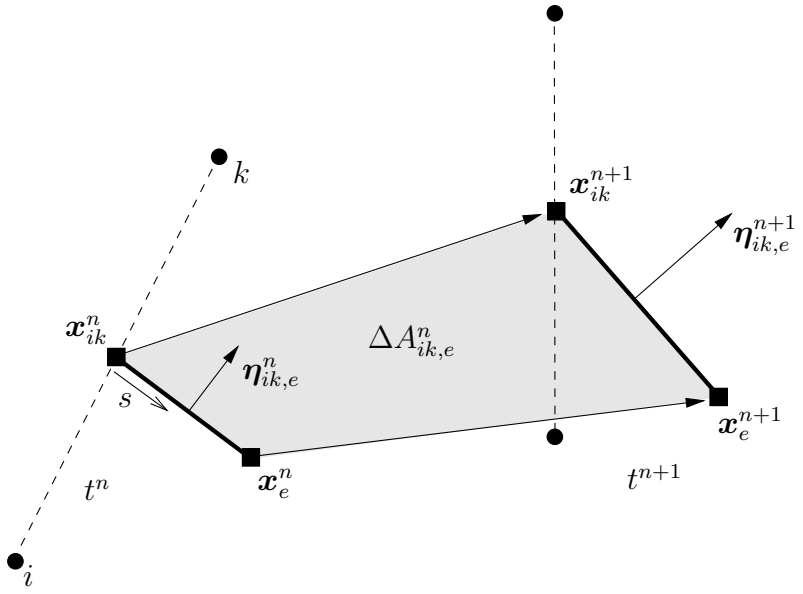


Fig. 7. The interface $\partial\mathcal{C}_{ik,e}$ is in fact the segment $(\boldsymbol{x}_{ik}, \boldsymbol{x}_e)$. The velocities $\boldsymbol{v}_{ik}$ and $\boldsymbol{v}_e$ of points $\boldsymbol{x}_{ik}$ and $\boldsymbol{x}_e$, respectively, are assumed to be constant over $t^n < t < t^{n+1}$.

e-th element, see section 2.2.2. From definition (36), one immediately obtains,

$$\Delta V_{i,ik,e}^{n} = \int_{t^n}^{t^{n+1}} \int_{\partial\mathcal{C}_{ik,e}(t)} \boldsymbol{v}\cdot\boldsymbol{n}_i = \int_{t^n}^{t^{n+1}} \frac{\boldsymbol{v}_{ik}(t)+\boldsymbol{v}_e(t)}{2}\cdot\boldsymbol{\eta}_{ik,e}(t)\,, \tag{41}$$

By assuming now that the velocities $\boldsymbol{v}_{ik}$ and $\boldsymbol{v}_e$ remain constant during the time step [25,9], an expression for $\boldsymbol{\eta}_{ik,e}(t)$ is easily obtained from (14) and by noticing that under the aforementioned assumption of constant $\boldsymbol{v}_{ik}$ and $\boldsymbol{v}_e$ for $t^n < t < t^{n+1}$ one has

$$\boldsymbol{x}_{ik}(t) \simeq \boldsymbol{x}_{ik}^n + \boldsymbol{v}_{ik}^n(t-t^n), \qquad \text{with } \boldsymbol{v}_{ik}^n = \frac{\boldsymbol{x}_{ik}^{n+1}-\boldsymbol{x}_{ik}^n}{\Delta t^n}$$

$$\boldsymbol{x}_{e}(t) \simeq \boldsymbol{x}_{e}^n + \boldsymbol{v}_{e}^n(t-t^n) \qquad \text{with } \boldsymbol{v}_{e}^n = \frac{\boldsymbol{x}_{e}^{n+1}-\boldsymbol{x}_{e}^n}{\Delta t^n}$$

and therefore from (14)

$$\begin{aligned}\boldsymbol{\eta}_{ik,e}(t) &= (\boldsymbol{x}_{ik}(t)-\boldsymbol{x}_e(t))\times\hat{\boldsymbol{z}}\\ &= (\boldsymbol{x}_{ik}^n-\boldsymbol{x}_e^n)\times\hat{\boldsymbol{z}} + (\boldsymbol{v}_{ik}^n-\boldsymbol{v}_e^n)\times\hat{\boldsymbol{z}}\,(t-t^n)\\ &= (\boldsymbol{x}_{ik}^n-\boldsymbol{x}_e^n)\times\hat{\boldsymbol{z}} + \left[\boldsymbol{x}_{ik}^{n+1}-\boldsymbol{x}_e^{n+1}-(\boldsymbol{x}_{ik}^n-\boldsymbol{x}_e^n)\right]\times\hat{\boldsymbol{z}}\,\frac{t-t^n}{\Delta^n}\end{aligned}$$

Namely, $\boldsymbol{\eta}_{ik,e}(t)$ is a linear function of time, or

$$\boldsymbol{\eta}_{ik,e}(t) = \boldsymbol{\eta}_{ik,e}^n + (\boldsymbol{\eta}_{ik,e}^{n+1}-\boldsymbol{\eta}_{ik,e}^n)\frac{t-t^n}{\Delta t^n}$$

By substituting the expression above into (41) and integrating over Δt^n, one has

$$\Delta V^n_{i,ik,e} \simeq (\boldsymbol{v}^n_{ik} + \boldsymbol{v}^n_e) \cdot (\boldsymbol{\eta}^n_{ik,e} + \boldsymbol{\eta}^{n+1}_{ik,e}) \frac{\Delta t^n}{4}$$

The average interface velocity $\boldsymbol{\nu}^{n+1}_{ik}$ satisfying the IVC condition (29) is therefore computed as

$$\boldsymbol{\nu}^{n+1}_{ik} = \frac{1}{4} \frac{\boldsymbol{\eta}^{n+1}_{ik}}{|\boldsymbol{\eta}^{n+1}_{ik}|^2} \sum_{e \in \mathcal{E}_i \cap \mathcal{E}_k} (\boldsymbol{v}^n_{ik} + \boldsymbol{v}^n_e) \cdot (\boldsymbol{\eta}^n_{ik,e} + \boldsymbol{\eta}^{n+1}_{ik,e}). \tag{42}$$

A similar procedure is followed to determine the average boundary velocity $\boldsymbol{\nu}_i$, to give

$$\Delta V^n_{i,\partial,e} = \int_{t^n}^{t^{n+1}} \int_{\partial \mathcal{C}_{i,e}(t) \cap \Omega(t)} \boldsymbol{v} \cdot \boldsymbol{n}_i = \int_{t^n}^{t^{n+1}} \frac{\boldsymbol{v}_i(t) + \boldsymbol{v}_e(t)}{2} \cdot \boldsymbol{\xi}_{i,e}(t)\,, \tag{43}$$

and hence

$$\boldsymbol{\nu}^{n+1}_i = \frac{1}{4} \frac{\boldsymbol{\xi}^{n+1}_i}{|\boldsymbol{\xi}^{n+1}_i|^2} \sum_{e \in \mathcal{E}_i \cap \partial\Omega} (\boldsymbol{v}^n_i + \boldsymbol{v}^n_e) \cdot (\boldsymbol{\xi}^n_{i,e} + \boldsymbol{\xi}^{n+1}_{i,e})\,, \tag{44}$$

Note that expressions (42) and (44) requires to know only the nodes' coordinates at time level n and $n+1$.

A different result is obtained by assuming that the grid velocity is a linear function of time in the interval $t^n < t < t^{n+1}$. In this case, grid coordinates are approximated over $t^n < t < t^{n+1}$ by the parabola

$$\boldsymbol{x}(t) = \frac{\boldsymbol{v}^{n+1} - \boldsymbol{v}^n}{2} \frac{(t - t^n)^2}{\Delta t^n} + \boldsymbol{v}^n (t - t^n) + \boldsymbol{x}^n$$

where the grid velocities at time t^n and t^{n+1} read

$$\boldsymbol{v}^n = -\frac{\boldsymbol{x}^{n+1} - 4\boldsymbol{x}^{n+\frac{1}{2}} + 3\boldsymbol{x}^n}{\Delta t^n} \qquad \text{and} \qquad \boldsymbol{v}^{n+1} = \frac{3\boldsymbol{x}^{n+1} - 4\boldsymbol{x}^{n+\frac{1}{2}} + \boldsymbol{x}^n}{\Delta t^n}$$

respectively. The elemental contribution to the metric vector $\boldsymbol{\eta}_{ik}$ is therefore a parabolic function as well, namely,

$$\begin{aligned}\boldsymbol{\eta}_{ik,e}(t) = \frac{1}{2}\Big[(\boldsymbol{v}^{n+1}_{ik} - \boldsymbol{v}^{n+1}_e) - (\boldsymbol{v}^n_{ik} - \boldsymbol{v}^n_e)\Big] \times \hat{\boldsymbol{z}}\, \frac{(t - t^n)^2}{\Delta t^n} \\ + (\boldsymbol{v}^n_{ik} - \boldsymbol{v}^n_e) \times \hat{\boldsymbol{z}}\, (t - t^n) + (\boldsymbol{x}^n_{ik} - \boldsymbol{x}^n_e),\end{aligned}$$

whose evaluation requires to known the nodes' coordinates at time level t^n,

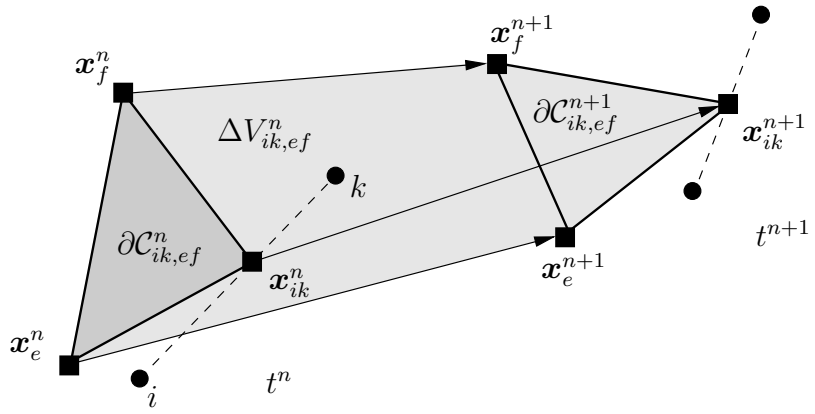


Fig. 8. Volume $\Delta V_{ik,ef}^n$ swept from time t^n to t^{n+1} by the interface portion $\partial\mathcal{C}_{ik,ef}$, namely, by the triangle $(\boldsymbol{x}_{ik}, \boldsymbol{x}_e, \boldsymbol{x}_f)$, pertaining to edge i-k, to the e-th element and to the f-th face of e, under the assumption that the velocity of points $\boldsymbol{x}_{ik}, \boldsymbol{x}_e$ and $\boldsymbol{x}_f$ is constant over Δt^n.

$t^{n+1/2}$ and t^{n+1}, which immediately gives

$$\Delta V_{i,ik,e} = \int_{t^n}^{t^{n+1}} \int_{\partial\mathcal{C}_{ik,e}} \boldsymbol{v}\cdot\boldsymbol{n} = \Big[(\boldsymbol{v}_{ik}^{n+1} + \boldsymbol{v}_e^{n+1})\cdot\boldsymbol{\eta}_{ik,e}^{n+1} + 2(\boldsymbol{v}_{ik}^{n+\frac{1}{2}} + \boldsymbol{v}_e^{n+\frac{1}{2}})\cdot\boldsymbol{\eta}_{ik,e}^{n+\frac{1}{2}} + (\boldsymbol{v}_{ik}^n + \boldsymbol{v}_e^n)\cdot\boldsymbol{\eta}_{ik,e}^n\Big]\frac{\Delta t^n}{12} \tag{45}$$

where, from definition (14),

$$\boldsymbol{\eta}_{ik,e}^n = (\boldsymbol{x}_{ik}^n - \boldsymbol{x}_e^n)\times\hat{\boldsymbol{z}}, \qquad \boldsymbol{\eta}_{ik,e}^{n+1/2} = (\boldsymbol{x}_{ik}^{n+1/2} - \boldsymbol{x}_e^{n+1/2})\times\hat{\boldsymbol{z}}$$
$$\text{and} \quad \boldsymbol{\eta}_{ik,e}^{n+1} = (\boldsymbol{x}_{ik}^{n+1} - \boldsymbol{x}_e^{n+1})\times\hat{\boldsymbol{z}}$$

and where $\boldsymbol{v}^{n+1/2} = (\boldsymbol{x}^{n+1} - \boldsymbol{x}^n)/\Delta t^n$. The average interfaces velocities $\boldsymbol{\nu}_{ik}^{n+1}$ and $\boldsymbol{\nu}_i^{n+1}$ are therefore computed as follows

$$\boldsymbol{\nu}_{ik}^{n+1} = \frac{1}{12}\frac{\boldsymbol{\eta}_{ik}^{n+1}}{|\boldsymbol{\eta}_{ik}^{n+1}|^2}\sum_{e\in\mathcal{E}_i\cap\mathcal{E}_k}\Big[(\boldsymbol{v}_{ik}^{n+1} + \boldsymbol{v}_e^{n+1})\cdot\boldsymbol{\eta}_{ik,e}^{n+1} + 2(\boldsymbol{v}_{ik}^{n+\frac{1}{2}} + \boldsymbol{v}_e^{n+\frac{1}{2}})\cdot\boldsymbol{\eta}_{ik,e}^{n+\frac{1}{2}} + (\boldsymbol{v}_{ik}^n + \boldsymbol{v}_e^n)\cdot\boldsymbol{\eta}_{ik,e}^n\Big], \tag{46}$$

$$\boldsymbol{\nu}_i^{n+1} = \frac{1}{12}\frac{\boldsymbol{\xi}_i^{n+1}}{|\boldsymbol{\xi}_i^{n+1}|^2}\sum_{e\in\mathcal{E}_i\cap\partial\Omega}\Big[(\boldsymbol{v}_i^{n+1} + \boldsymbol{v}_e^{n+1})\cdot\boldsymbol{\xi}_{i,e}^{n+1} + 2(\boldsymbol{v}_i^{n+\frac{1}{2}} + \boldsymbol{v}_e^{n+\frac{1}{2}})\cdot\boldsymbol{\xi}_{i,e}^{n+\frac{1}{2}} + (\boldsymbol{v}_i^n + \boldsymbol{v}_e^n)\cdot\boldsymbol{\xi}_{i,e}^n\Big], \tag{47}$$

where the derivation of the last expression, similar to that presented above, has been omitted for conciseness. The influence on the overall accuracy of using either the linear or the parabolic approximations for the node coordinates presented above is investigated in section 6.1.

4.1.2 Three-spatial dimensions

Considering now the three-dimensional case, the difference $V_{ik}^{n+1} - V_{ik}^{n}$ is expressed in terms of element/face contributions as follows

$$\Delta V_{i,ik}^{n} = V_{i,ik}^{n+1} - V_{i,ik}^{n} = \sum_{e \in \mathcal{E}_i \cap \mathcal{E}_k} \sum_{f \in \mathcal{F}_{ik,e}} \Delta V_{i,ik,ef}^{n},$$

where, with reference to Fig. 8, $\Delta V_{i,ik,ef}^{n}$ is the area swept by the portion $\partial \mathcal{C}_{ik,ef}$, namely, by the triangle with vertices $(\boldsymbol{x}_{ik}, \boldsymbol{x}_e, \boldsymbol{x}_f)$, of the finite volume interface $\partial \mathcal{C}_i$ pertaining to the edge i-k, to the e-th element and to the f-th face of e, see section 2.2.2. As in the two-dimensional case, each contribution $\Delta V_{i,ik,ef}^{n}$ is now written as the integral of the local velocity $\boldsymbol{v}$ over the triangle $\partial \mathcal{C}_{ik,ef}$ from time level t^n to t^{n+1} as follows

$$\int_{t^n}^{t^{n+1}} \int_{\partial \mathcal{C}_{ik,ef}} \boldsymbol{v} \cdot \boldsymbol{n} = \frac{\boldsymbol{v}_{ik}^{n} + \boldsymbol{v}_{e}^{n} + \boldsymbol{v}_{f}^{n}}{3} \cdot \int_{t^n}^{t^{n+1}} \boldsymbol{\eta}_{ik,ef}(t), \tag{48}$$

where the grid velocity has been assumed to be constant during the time step. An explicit expression for $\boldsymbol{\eta}_{ik,ef}(t)$ is now computed from the following expressions for the coordinates of the vertices

$$\begin{aligned} \boldsymbol{x}_{ik}(t) &= \boldsymbol{x}_{ik}^{n} + \boldsymbol{v}_{ik}^{n}(t - t^n), \\ \boldsymbol{x}_{e}(t) &= \boldsymbol{x}_{e}^{n} + \boldsymbol{v}_{e}^{n}(t - t^n), \\ \boldsymbol{x}_{f}(t) &= \boldsymbol{x}_{f}^{n} + \boldsymbol{v}_{f}^{n}(t - t^n), \end{aligned}$$

valid for $t^n < t < t^{n+1}$ and under the assumption of constant velocities over Δt^n, and from definition (18) as follows

$$\begin{aligned} \boldsymbol{\eta}_{ik,ef}(t) &= \frac{1}{2}[\boldsymbol{x}_f(t) - \boldsymbol{x}_e(t)] \times [\boldsymbol{x}_{ik}(t) - \boldsymbol{x}_e(t)] \\ &= \frac{1}{2}[\boldsymbol{x}_f^n - \boldsymbol{x}_e^n + (\boldsymbol{v}_f^n - \boldsymbol{v}_e^n)(t - t^n)] \times [\boldsymbol{x}_{ik}^n - \boldsymbol{x}_e^n + (\boldsymbol{v}_{ik}^n - \boldsymbol{v}_e^n)(t - t^n)] \\ &= \frac{1}{2}\Big\{ (\boldsymbol{x}_f^n - \boldsymbol{x}_e^n) \times (\boldsymbol{x}_{ik}^n - \boldsymbol{x}_e^n) \\ &\quad + \Big[(\boldsymbol{x}_f^n - \boldsymbol{x}_e^n) \times (\boldsymbol{x}_{ik}^{n+1} - \boldsymbol{x}_e^{n+1}) + (\boldsymbol{x}_f^{n+1} - \boldsymbol{x}_e^{n+1}) \times (\boldsymbol{x}_{ik}^n - \boldsymbol{x}_e^n) \\ &\qquad - 2(\boldsymbol{x}_f^n - \boldsymbol{x}_e^n) \times (\boldsymbol{x}_{ik}^n - \boldsymbol{x}_e^n) \Big] \frac{t - t^n}{\Delta t^n} \\ &\quad + \Big[(\boldsymbol{x}_f^{n+1} - \boldsymbol{x}_e^{n+1}) \times (\boldsymbol{x}_{ik}^{n+1} - \boldsymbol{x}_e^{n+1}) \\ &\qquad + (\boldsymbol{x}_f^n - \boldsymbol{x}_e^n) \times (\boldsymbol{x}_{ik}^{n+1} - \boldsymbol{x}_e^{n+1}) + (\boldsymbol{x}_f^{n+1} - \boldsymbol{x}_e^{n+1}) \times (\boldsymbol{x}_{ik}^n - \boldsymbol{x}_e^n) \\ &\qquad\quad + (\boldsymbol{x}_f^n - \boldsymbol{x}_e^n) \times (\boldsymbol{x}_{ik}^n - \boldsymbol{x}_e^n) \Big] \left(\frac{t - t^n}{\Delta t^n} \right)^2 \Big\} \end{aligned}$$

Differently from the two-dimensional case, where $\boldsymbol{\eta}_{ik,e}(t)$ is a linear function of time, in the three-spatial dimension $\boldsymbol{\eta}_{ik,ef}$ is found to be a parabolic function

of t, or

$$\boldsymbol{\eta}_{ik,ef}(t) = \frac{1}{2}\Big(\boldsymbol{\eta}_{ik,ef}^{n+1} - 2\boldsymbol{\eta}_{ik,ef}^{n+\frac{1}{2}} + \boldsymbol{\eta}_{ik,ef}^{n}\Big)\left(\frac{t-t^n}{\Delta t^n}\right)^2 - \Big(\boldsymbol{\eta}_{ik,ef}^{n+1} - 4\boldsymbol{\eta}_{ik,ef}^{n+\frac{1}{2}} + 3\boldsymbol{\eta}_{ik,ef}^{n}\Big)\frac{t-t^n}{\Delta t^n} + \boldsymbol{\eta}_{ik,ef}^{n}$$

where $t^{n+\frac{1}{2}} = t^n + \frac{\Delta t^n}{2}$ and $\boldsymbol{\eta}_{ik,ef}^{n+1/2} = \boldsymbol{\eta}_{ik,ef}(t^{n+1/2})$. Note that the evaluation of $\boldsymbol{\eta}_{ik,ef}^{n+1/2}$, namely,

$$\boldsymbol{\eta}_{ik,ef}^{n+\frac{1}{2}} = \frac{1}{4}(\boldsymbol{x}_f^{n+1} - \boldsymbol{x}_e^{n+1}) \times (\boldsymbol{x}_{ik}^{n+1} - \boldsymbol{x}_e^{n+1}) + \frac{3}{4}(\boldsymbol{x}_f^{n+1} - \boldsymbol{x}_e^{n+1}) \times (\boldsymbol{x}_{ik}^{n} - \boldsymbol{x}_e^{n}) + \frac{3}{4}(\boldsymbol{x}_f^{n} - \boldsymbol{x}_e^{n}) \times (\boldsymbol{x}_{ik}^{n+1} - \boldsymbol{x}_e^{n+1}) - \frac{1}{4}(\boldsymbol{x}_f^{n} - \boldsymbol{x}_e^{n}) \times (\boldsymbol{x}_{ik}^{n} - \boldsymbol{x}_e^{n})$$

does not require to compute any intermediate value for the grid nodes' coordinates, namely, its expression involves only nodes' coordinates at time n and $n+1$. Substituting the above expression into (48) and integrating in time one obtains

$$\Delta V_{i,ik,ef}^n = (\boldsymbol{v}_{ik}^n + \boldsymbol{v}_e^n + \boldsymbol{v}_f^n) \cdot \Big(\boldsymbol{\eta}_{ik,ef}^{n+1} + 4\boldsymbol{\eta}_{ik,ef}^{n+\frac{1}{2}} + \boldsymbol{\eta}_{ik,ef}^{n}\Big)\frac{\Delta t^n}{18}. \tag{49}$$

It is now possible to provide an expression for the average interface velocity satisfying the IVC condition (29) as

$$\boldsymbol{\nu}_{ik}^{n+1} = \frac{1}{18}\frac{\boldsymbol{\eta}_{ik}^{n+1}}{|\boldsymbol{\eta}_{ik}^{n+1}|^2} \sum_{e\in\mathcal{E}_i\cap\mathcal{E}_k} \sum_{f\in\mathcal{F}_{ik,e}} (\boldsymbol{v}_{ik}^n + \boldsymbol{v}_e^n + \boldsymbol{v}_f^n) \cdot \Big(\boldsymbol{\eta}_{ik,ef}^{n+1} + 4\boldsymbol{\eta}_{ik,ef}^{n+\frac{1}{2}} + \boldsymbol{\eta}_{ik,ef}^{n}\Big). \tag{50}$$

A similar procedure leads to the following definition of the average boundary velocity, in which

$$\Delta V_{i,\partial,ef}^n = (\boldsymbol{v}_i^n + \boldsymbol{v}_e^n + \boldsymbol{v}_f^n) \cdot \Big(\boldsymbol{\xi}_{i,ef}^{n+1} + 4\boldsymbol{\xi}_{ik,ef}^{n+\frac{1}{2}} + \boldsymbol{\xi}_{i,ef}^{n}\Big)\frac{\Delta t^n}{18}, \tag{51}$$

and hence

$$\boldsymbol{\nu}_{i}^{n+1} = \frac{1}{18}\frac{\boldsymbol{\xi}_{i}^{n+1}}{|\boldsymbol{\xi}_{i}^{n+1}|^2} \sum_{e\in\mathcal{E}_i^{\partial}} \sum_{f\in\mathcal{F}_{i,e}} (\boldsymbol{v}_{ik}^n + \boldsymbol{v}_e^n + \boldsymbol{v}_f^n) \cdot \Big(\boldsymbol{\xi}_{i,ef}^{n+1} + 4\boldsymbol{\xi}_{i,ef}^{n+\frac{1}{2}} + \boldsymbol{\xi}_{i,ef}^{n}\Big). \tag{52}$$

4.2 BDF and multistep schemes with variable time-step

In the present section, the expressions of the average interface veloctities $\boldsymbol{\nu}_{ik}$ and $\boldsymbol{\nu}_i$ satisfying the IVC condition for a Backward Differences Formulae (BDF) scheme and for a Multistage (MS) scheme are now derived. For a

nonlinear ODE $dy/dx = f(x, y)$, the second-order BDF scheme with variable time step reads

$$a_{-1}y^{n+1} + a_0 y^n + a_1 y^{n-1} = f(x^{n+1}, y^{n+1})\,\Delta t^n, \tag{53}$$

where the coefficients a_i are

$$a_{-1} = \frac{1+2\beta^n}{1+\beta^n},\ a_0 = -(1+\beta^n),\ a_1 = \frac{(\beta^n)^2}{1+\beta^n}, \quad \text{with } \beta^n = \frac{\Delta t^n}{\Delta t^{n-1}}.$$

For convenience, the BDF scheme is now recast in terms of time differences $\Delta y^{n+1} = y^{n+1} - y^n$ as follows

$$\alpha_{-1}\Delta y^n + \alpha_0 \Delta y^{n-1} = f(x^{n+1}, y^{n+1})\,\Delta t^n, \tag{54}$$

$$\text{with} \qquad \alpha_{-1} = \frac{1+2\beta^n}{1+\beta^n} \qquad \text{and} \qquad \alpha_0 = -\frac{(\beta^n)^2}{1+\beta^n}.$$

The fully discrete form of (38) therefore reads

$$\begin{cases} \alpha_{-1}\Delta[V_i^n\,\mathsf{u}_i^n] - \alpha_0\Delta[V_i^{n-1}\,\mathsf{u}_i^{n-1}] = \Big[\displaystyle\sum_{k\in\mathcal{K}_{i,\neq}} \Psi(\mathsf{u}_i^{n+1}, \mathsf{u}_k^{n+1}, \boldsymbol{\nu}_{ik}^{n+1}, \boldsymbol{\eta}_{ik}^{n+1}) \\ \qquad\qquad + \Psi^{\partial}(\mathsf{u}_i^{n+1}, \boldsymbol{\nu}_i^{n+1}, \boldsymbol{\xi}_i^{n+1})\Big]\,\Delta t^n \\ \alpha_{-1}\Delta V_{i,ik}^n - \alpha_0 \Delta V_{i,ik}^{n-1} = \boldsymbol{\nu}_{ik}^{n+1}\cdot\boldsymbol{\eta}_{ik}^{n+1}\,\Delta t^n \\ \alpha_{-1}\Delta V_{i,\partial}^{n1} - \alpha_0 \Delta V_{i,\partial}^{n-1} = \boldsymbol{\nu}_{i}^{n+1}\cdot\boldsymbol{\xi}_{i}^{n+1}\,\Delta t^n \end{cases} \tag{55}$$

Therefore, one has

$$\begin{aligned} \boldsymbol{\nu}_{ik}^{n+1} &= \frac{1}{\Delta t^n}\big(\alpha_{-1}\Delta V_{i,ik}^n - \alpha_0 \Delta V_{i,ik}^{n-1}\big)\frac{\boldsymbol{\eta}_{ik}^{n+1}}{|\boldsymbol{\eta}_{ik}^{n+1}|^2} \\ \boldsymbol{\nu}_{i}^{n+1} &= \frac{1}{\Delta t^n}\big(\alpha_{-1}\Delta V_{i,\partial}^n - \alpha_0 \Delta V_{i,\partial}^{n-1}\big)\frac{\boldsymbol{\xi}_{i}^{n+1}}{|\boldsymbol{\xi}_{i}^{n+1}|^2} \end{aligned} \tag{56}$$

where the values of $\Delta V_{i,ik}^n$ and $\Delta V_{i,ik}^{n-1}$ are computed from relations (41) and (49) in two and three spatial dimensions, respectively, and where $\Delta V_{i,\partial}^n$ and $\Delta V_{i,\partial}^{n-1}$ are computed from relations (43) and (51). The above definition of the interface velocity allows for the IVC condition to be identically satisfied and, differently from other approaches [26], does not require to modify the coefficients α of the BDF scheme, thus preserving the time accuracy.

In fact, the definitions above can be extended to allow for the use of a generic Multistep (MS) scheme a follows. To this purpose, a MS scheme for the model

ODE $dx(t)/dt = f(t)$ for the scalar unknown $x = x(t)$, namely

$$\sum_{l=-1}^{p} a_l x^{n-l} = \Delta t^n \sum_{h=q}^{r} b_h f(x^{n-h}),$$

with $p > 0$, $q \geq -1$ and $p \geq q$, is now considered. Note that $a_{-1} \neq 0$ and that $b_q \neq 0$. If $q = -1$, the scheme is implicit. For a MS scheme to be consistent, the coefficients a_l and b_h must satisfy the following conditions

$$\sum_{l=-1}^{p} a_l = 0, \qquad \text{and} \qquad \sum_{l=-1}^{p} l a_l + \sum_{h=-q}^{r} b_h = 0,$$

which allow for the MS scheme to be recast in the following form

$$\sum_{l=-1}^{p-1} \alpha_l \Delta x^{n-l} = \Delta t^n \sum_{h=-q}^{r} b_h f(x^{n-h}),$$

where $\alpha^l = \sum_{j=-1}^{l} a^j$. The average interface velocities at time level $n - q$ satisfying the IVC condition for a MS scheme are therefore computed as

$$\begin{aligned} \boldsymbol{\nu}_{ik}^{n-q} &= \frac{1}{\Delta t^n} \left(\sum_{l=-1}^{p-1} \alpha_l \Delta V_{i,ik}^{n-l} - \Delta t^n \sum_{h=1-q}^{r} b_h \boldsymbol{\nu}_{ik}^{n-h} \cdot \boldsymbol{\eta}_{ik}^{n-h} \right) \frac{\boldsymbol{\eta}_{ik}^{n+1}}{|\boldsymbol{\eta}_{ik}^{n+1}|^2} \\ \boldsymbol{\nu}_{i}^{n-q} &= \frac{1}{\Delta t^n} \left(\sum_{l=-1}^{p-1} \alpha_l \Delta V_{i,\partial}^{n-l} - \Delta t^n \sum_{h=1-q}^{r} b_h \boldsymbol{\nu}_{i}^{n-h} \cdot \boldsymbol{\xi}_{ik}^{n-h} \right) \frac{\boldsymbol{\xi}_{i}^{n+1}}{|\boldsymbol{\xi}_{i}^{n+1}|^2} \end{aligned} \tag{57}$$

4.3 The IVC condition and the Geometric Conservation Law (GCL)

Starting from the original statement of Thomas and Lombard [11], many authors recognized the importance of the so-called Geometric Conservation Law (GCL) in solving dynamic mesh problem in general and fluid-structure interactions in particular. Strictly speaking, the GCL is a constraint on the discrete form of the flow equations which states that the numerical scheme is to be capable of reproducing a uniform flow over a moving mesh without introducing any numerical disturbances. The GCL has been found to be critical especially in aeroelastic computations and aeroelastic solver which do not satisfy the GCL have been shown to compute flutter conditions incorrectly [27].

The interplay of the GCL and time-accuracy has been investigated by many authors [27–29,13]; in particular, in [30], the GCL is proved to be a sufficient condition for achieving first-order time accuracy. In [29] the GCL is demonstrated to be not necessary for high-order time accuracy and high-order time accurate numerical schemes that do not fulfill the GCL have been devised. In

[13] the relation between time-accuracy and the GCL is further investigated and the GCL is enforced in time-accurate numerical scheme to preserve the non-linear stability of the time integration scheme [31]. More recently, in [32], the rôle of the GCL in devising time-accurate numerical schemes on dynamic meshes is studied in the framework of edge-based solvers for compressible flows.

In the present edge-based finite volume approach, the GCL condition for the BE scheme, namely, cf. (39)

$$V_i^{n+1}\,\mathsf{u}_i^{n+1} - V_i^n\,\mathsf{u}_i^n \\ = \Big[\sum_{k\in\mathcal{K}_{i,\neq}} \Psi(\mathsf{u}_i^{n+1},\mathsf{u}_k^{n+1},\boldsymbol{\nu}_{ik}^{n+1},\boldsymbol{\eta}_{ik}^{n+1}) + \Psi^{\partial}(\mathsf{u}_i^{n+1},\boldsymbol{\nu}_i^{n+1},\boldsymbol{\xi}_i^{n+1})\Big]\,\Delta t^n$$

is obtained by imposing that the numerical scheme computes a uniform flow, namely, $\mathsf{u}_i = \text{const.}\ \forall i \in \mathcal{K}$, exactly. Hence

$$V_i^{n+1} - V_i^n = \Big[\sum_{k\in\mathcal{K}_{i,\neq}} \boldsymbol{\nu}_{ik}^{n+1}\cdot\boldsymbol{\eta}_{ik}^{n+1} + \boldsymbol{\nu}_i^{n+1}\cdot\boldsymbol{\xi}_i^{n+1}\Big]\,\Delta t^n,$$

which is easily verified to be an identity provided that the interface velocities satisfy the IVC conditions (42), (44) and (50), (52) in two and three spatial dimensions, respectively, see section 4.1. The GCL is fulfilled by all the IVC-compliant time integration schemes considered in the present work. In the fact, the discrete system implementing the IVC condition ensures that the discrete counterpart of the conservation equation

$$\frac{dV_i}{dt} = \oint_{\partial\mathcal{C}_i} \boldsymbol{v}\cdot\boldsymbol{n}_i,$$

is computed in a way that is consistent with the time integration scheme used in the flow solver. As a result, the IVC condition introduced here is a sufficient condition for the numerical scheme to satisfy the GCL. Remarkably enough, as noted in section 3, the IVC condition generalizes the GCL to possibly non uniform flows and simplifies to the GCL itself in the case of uniform flows.

The above can be further clarified by resorting to the following one-dimensional example. A finite volume $\mathcal{C}_i$ (a segment, in the present one-dimensional case), bounded by two interfaces (points) located at x_a and x_b, respectively, with $x_a < x_b$, is considered. Both coordinates x_a and x_b are independent functions of time. Note that the outward normal n_a and n_b at x_a and x_b, respectively, are scalar quantities in one spatial dimension and that $n_a = -1$ and $n_b = 1$.

Therefore, the mass conservation law for the considered finite volume reads

$$\frac{d}{dt}\int_{x_a(t)}^{x_b(t)}\rho(x,t)\,dx = \rho(x_a(t),t)\Big[w(x_a(t),t)-v(x_a(t),t)\Big] - \rho(x_b(t),t)\Big[w(x_b(t),t)-v(x_b(t),t)\Big], \quad (58)$$

where w is the local fluid velocity. The interface velocities $v_a(t) = v(x_a(t),t)$ and $v_b(t) = v(x_b(t),t)$ are now computed according to the prescriptions of the GCL and the IVC as follows. More precisely, we are interested in evaluating the consequences of substituting the actual interface velocities $v_a(t)$ and $v_b(t)$ with the velocities $\tilde{v}_a(t)$ and $\tilde{v}_b(t)$ computed by imposing either the GCL or the IVC.

The interface velocities $\tilde{v}_a^{\mathrm{GCL}}(t)$ and $\tilde{v}_b^{\mathrm{GCL}}(t)$ fulfilling the GCL conditions can be easily computed by evaluating the mass conservation law for a flow with constant density and constant fluid velocity, namely, in an uniform flow. In this case, one has

$$\frac{dV_i}{dt} \equiv \frac{d}{dt}\int_{x_a(t)}^{x_b(t)} dx = \tilde{v}_b^{\mathrm{GCL}}(t) - \tilde{v}_a^{\mathrm{GCL}}(t),$$

therefore, the GCL reduces to the following condition

$$\tilde{v}_b^{\mathrm{GCL}}(t) - \tilde{v}_a^{\mathrm{GCL}}(t) = v_b(t) - v_a(t), \quad (59)$$

which identifies a one-parameter family of interface velocities. The interface velocities $\tilde{v}_a^{\mathrm{IVC}}(t)$ and $\tilde{v}_b^{\mathrm{IVC}}(t)$ satisfying the IVC are instead computed from definition (29) as follows

$$\tilde{v}_a^{\mathrm{IVC}}(t) = v_a(t) \qquad \text{and} \qquad \tilde{v}_b^{\mathrm{IVC}}(t) = v_b(t), \quad (60)$$

and are univocally determined by imposing the IVC only. It is to be noted that the interface velocities (60) computed via the IVC satisfy the GCL condition (59). Conversely, within the one-parameter family of solutions identified by the GCL condition (59), only one pair of interface velocities fulfills the IVC.

As expected, both choices (59) and (60) are feasible, that is, do not introduce any approximation, if substituted into the mass conservation law (58) in the case of a uniform steady flow, namely, in the case $\rho(x,t) \equiv \rho_0$ and $w \equiv w_0$, where both ρ_0 and w_0 are constant. In order to elucidate the differences between the choices (59) and (60), the mass conservation law (58) is now specialized to a steady but nonuniform flow, for which $\rho(x,t) = \rho(x) = \rho_0 + x(\rho_1 - \rho_0)/L$ and $w(x,t) = \rho_0 w_0/\rho(x)$, where ρ_1 is a constant and L is the length of the domain. Substituting these definitions into (58) immediately

gives

$$\rho_0\Big[v_b(t)-v_a(t)\Big]+\frac{\rho_1-\rho_0}{L}\Big[x_b(t)v_b(t)-x_a(t)v_a(t)\Big]$$
$$=\rho_0\Big[\tilde{v}_b(t)-\tilde{v}_a(t)\Big]+\frac{\rho_1-\rho_0}{L}\Big[x_b(t)\tilde{v}_b(t)-x_a(t)\tilde{v}_a(t)\Big],$$

where $\tilde{v}_a(t)$ and $\tilde{v}_b(t)$ indicate the modified interface velocities satifying either the GCL or the IVC. The equation above is identically satisfied if $\tilde{v}_a(t)$ and $\tilde{v}_b(t)$ are computed according to definitions (60), that is, according to the IVC. This is not the case for interfaces velocities fullfilling the GCL, cf. relation (59). To quantify the error introduced by a GCL-compliant scheme that does not fulfill the IVC, the very simple situation in which the control volume translates at constant velocity is now studied. The density and fluid velocity profile are the same considered above. The mass conservation law (58) for $v_a = v_b = v_0$ reads

$$\frac{d}{dt}\int_{x_a(t)}^{x_b(t)}\rho(x,t)\,dx = l(\rho_1-\rho_0)v_0,$$

where $l=(x_b(t)-x_a(t))/L=$ constant. The interface velocities $v_a(t)$ and $v_b(t)$ in (58) are now replaced by the velocities $\tilde{v}_a(t)$ and $\tilde{v}_b(t)$ computed using the GCL. These are chosen as

$$\tilde{v}=\alpha v_a(t)=\alpha v_0 \qquad \text{and} \qquad \tilde{v}_b(t)=v_b(t)+(\alpha-1)v_a(t)=\alpha v_0,$$

where $\alpha\in\mathbb{R}$, which are immediately verified to satisfy the GCL (59). Note that for $\alpha=1$ the interface velocities defined above fulfill also the IVC (60). Substituting $\tilde{v}_a(t)$ and $\tilde{v}_b(t)$ into (58) gives

$$\frac{d}{dt}\int_{x_a(t)}^{x_b(t)}\rho(x,t)\,dx = l(\rho_1-\rho_0)v_0+(\alpha-1)l(\rho_1-\rho_0)v_0,$$

where the second term on the right hand side represents the error introduced by substituting the exact interface velocities with their GCL-compliant counterpart. This error cancels only if $\alpha=1$, namely, if the IVC (60) is also satisfied.

Admittely, choice (60) for the interface velocities is indeed the most obvious one and numerous numerical scheme on dynamic meshes enforce already the IVC at each interface, although the GCL condition (59) only is explicitly imposed. Moreover, as noticed above, the latter is not sufficient to uniquely determine the interface velocities. The present study is a first attempt to provide a formal definition of the IVC and to clarify how currently used methods, that implicitly impose the IVC, are suitable also for nonuniform flows.

5 Mesh movement

In the present section, the mesh movement strategy is briefly described. The boundary of the new (deformed) mesh is to be conformal to the new boundaries of the domain and, at the same time, the overall quality of the mesh elements in the inner domain must be preserved to reduce numerical errors.

The mesh movement is performed in two steps. First, the displacement of each boundary node of the fluid mesh is obatined by either solving the coupled structural problems or from a given movement law; then, the position of the inner nodes is modified accordingly. The first step is usually not trivial because the mesh used for the structure and the one used for representing the boundary of the flow field are generally different, so an interface scheme must be defined to exchange information between the two grids. In the present work, a conservative interfacing procedure based on the Moving Least Square (MLS) method is used [33].

The displacement of internal nodes can be obtained using different strategies. These can be gathered in two classes: interpolation methods [34,35], mainly used for structured meshes, and those based on some form of elastic analogy, which are more suitable for unstructured meshes. Batina [36] introduced the elastic analogy by representing each side of the grid as a spring with a nonlinear stiffness proportional to the edge length. To avoid the occurrence of invalid elements with negative volumes, Degand and Farhat [37] introduced additional torsional springs at each vertex. Given its complexity, the mesh movement step may require a nonnegligible computational effort to the point that it may become one of the most time-consuming tasks in the computation [37].

Therefore, the envisaged deformation scheme must fulfill the following requirements: a) Robustness: the scheme must handle significant boundary displacements and be capable of producing valid grids (all elements with positive volume) with an acceptable quality especially in the areas where low numerical errors are sought for, i.e. near the wall boundaries. b) Computational efficiency. c) Easy of use; the user intervention on the algorithm is to be minimal.

The grid deformation algorithm presented here extends to idea of the elastic analogy by representing each element as a deformable body and moves from the discussion presented in [38]. Differently from the spring analogy, such a choice avoids element breakthrough also in the case of large deformations. To reduce the computational burden, a simple linear constitutive law is used, namely, in three spatial dimensions one has $\boldsymbol{\sigma} = \{\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{xy}, \sigma_{yz}, \sigma_{zx}\}^T$, and $\boldsymbol{\varepsilon} = \{\varepsilon_{xx}, \varepsilon_{yy}, \varepsilon_{zz}, \varepsilon_{xy}, \varepsilon_{yz}, \varepsilon_{zx}\}^T$, as

$$\boldsymbol{\sigma} = \boldsymbol{D}\boldsymbol{\varepsilon},$$

where the $\boldsymbol{D}$ matrix is equal to

$$\boldsymbol{D} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-2\nu & 0 & 0 \\ 0 & 0 & 0 & 0 & 1-2\nu & 0 \\ 0 & 0 & 0 & 0 & 0 & 1-2\nu \end{pmatrix},$$

where E is the Young elastic modulus and ν is the Poisson coefficient. When two dimensional grids are investigated, the analogy with the plane strain elastic model is used; the stress and strain vectors become $\boldsymbol{\sigma} = \{\sigma_{xx}, \sigma_{yy}, \sigma_{xy}\}^T$, and $\boldsymbol{\varepsilon} = \{\varepsilon_{xx}, \varepsilon_{yy}, \varepsilon_{xy}\}^T$, and the $\boldsymbol{D}$ matrix equal to

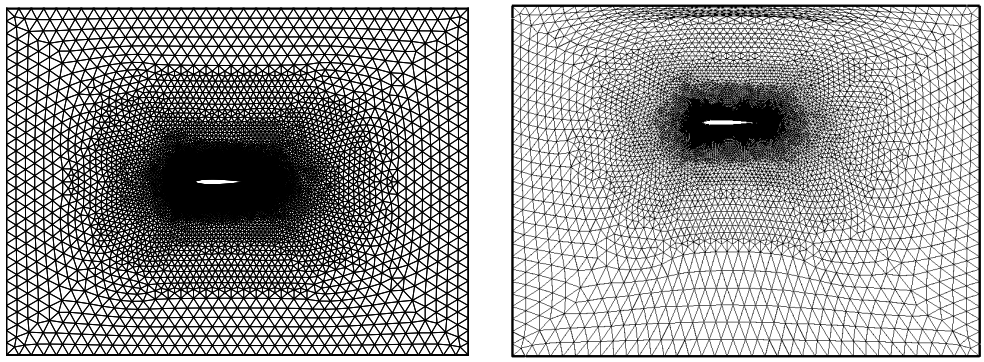
$$\boldsymbol{D} = \frac{E}{(1+\nu)(1-2\nu)} \begin{pmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & 1-2\nu \end{pmatrix}.$$

The correct grid deformation is achieved adopting a local Young modulus proportional to the minimal dimension of each element following a simple law

$$E_e = \frac{1}{\min\limits_{i,k \in \mathcal{K}_e} \|\boldsymbol{x}_i - \boldsymbol{x}_k\|^\beta}, \tag{61}$$

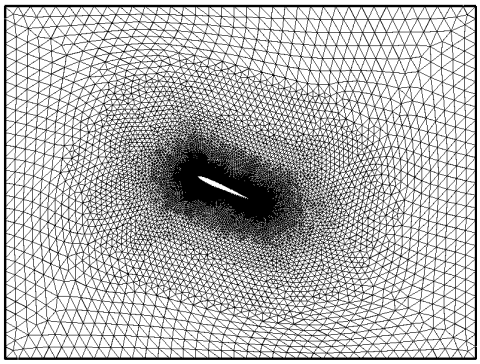
where $\mathcal{K}_e$ is the set of all nodes belonging to the e-th element. In this way the small elements close to wall boundaries are more stiff, so they tend to move rigidly with the walls, leaving the burden to absorb the global deformations on the larger elements, usually located far from the boundaries. The coefficient β can be used to control the mesh deformation behavior, increasing the stiffness ratio between small and large elements. A Poisson coefficient $\nu \in [0; 0.35]$ is chosen in order to avoid bad numerical conditioning of the problem. The mesh deformation problem is then solved by measn of a standard finite element approach. The wall boundary displacements are imposed simply as Dirichlet boundary conditions for the elastic mesh problem. Further improvements can be obtained through the adoption of anysotropic continua.

The effectiveness of the proposed strategy is shown in Fig. 9(b), where a two-dimensional unstructured mesh around the NACA 0012 airfoil is deformed to adapt it to a one chord plunge. The variable stiffness produces an almost rigid displacement for the small triangles near the airfoil walls, with a visible

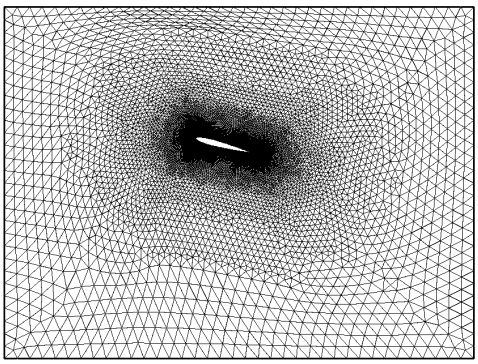


(a) Non-deformed grid

(b) One chord airfoil plunge



(c) 20 deg. airfoil pitch



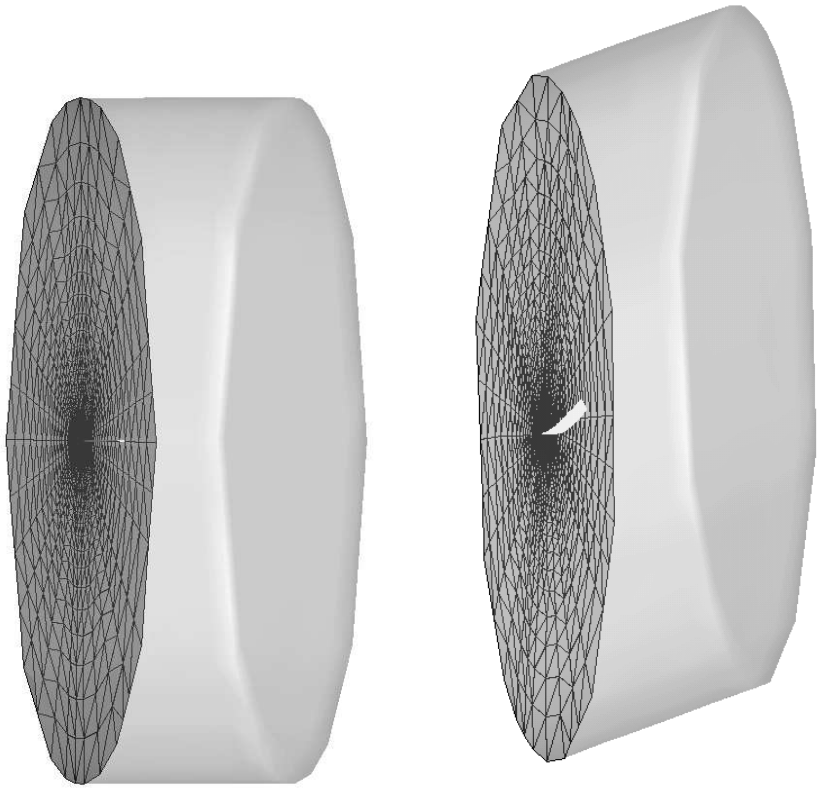
(d) Pitch and plunge linear combination

Fig. 9. Grid deformation for a NACA 0012 airfoil.

distortion only near the grid external boundary, where larger numerical errors may be acceptable. A similar behavior is found for the airfoil pitch, Fig. 9(c).

Furthermore, the linearity of the equations describing the mesh movement problems allows for a further reduction of the computational time, since the global mesh deformation can be represented as a superposition of basic deformed grids computed in advance. An example is shown in Fig. 9(d), where the two movement of pitch and plunge are linearly combined. Large saving are obtained when three dimensional cases are solved, using as basic elements for the superposition the deformed meshes associated with structural normal modes [1]. Of course, the superposition approach should be applied only when small structural displacements are considered, which is usually the case when aircraft aeroelastic stability is under investigation.

The quality of the deformed mesh can be further improved by allowing the displacement of the far field boundaries. In this way it is possible to achieve the required mesh deformation without large element distortions. Figure 10 shows the three-dimensional grid around the AGARD 445.6 wing, which is considered as a test case in the following section. The left side shows the non-deformed grid with the shaded far field boundary. The right side shows the deformed wing surface grid and the corresponding deformation of the external



(a) (b)

Fig. 10. Three-dimensional mesh aroound the AGARD 445.6 wing. a) Undeformed far-field boundary. b) The far-field boundary is allowed to move to imporve the mesh quality.

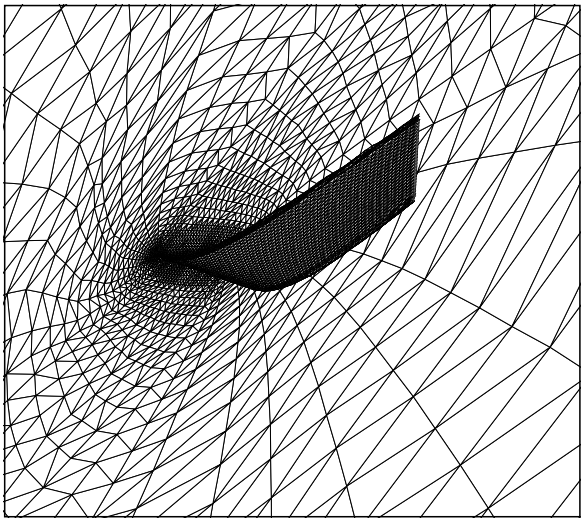


Fig. 11. Detail of the computational grid on the surface of the AGARD 445.6 wing (cf. Fig. 10) for a deformable far-field boundary.

boundaries. In this case large deformations can be achieved easily, as shown in Fig. 11.

6 Numerical results

Numerical simulations are now performed to investigate the influence of the IVC on the overall scheme stability and time accuracy.

The BDF2 and BDF3 time integration schemes have been implemented and compared for two-dimensional computations of a NACA0012 pitching airfoil. For the three-dimensional case a full aeroelastic stability analysis is performed for the Agard 445.6 wing in the subsonic, transonic and supersonic regimes.

6.1 Pitching airfoil

The first investigation on the numerical properties of the proposed scheme are performed in the case of a NACA 0012 pitching airfoil. The computational domain is represented by an unstructured mesh of triangles with 4686 nodes; the far-filed boundary is a circle with radius equal to 20 chords. The angle of attack of the simulated airfoil is prescribed as a sinusoidal function of time with a mean value of 0.016° and amplitude of 2.51°, with a reduced frequency $k = \omega c/2v_\infty = 0.1589$, where c is the airfoil chord, ω the frequency and v_∞ the flow velocity. A reference solution is computed by performing a time integration with a very small time step, using 512 steps per period; the relative errors are computed as the L_2 norm of the difference between the reference solution and the approximated one. In Figs. 12–13 the values of lift coefficient C_ℓ for BDF2 and BDF3 schemes are compared with the reference solutions at two different asymptotic Mach numbers. For large time steps, namely, $\Delta\tau = tv_\infty/c = 0.128$, the BDF3 scheme is found to be more accurate than BDF2. A more refined time discretization, $\Delta\tau = 0.03125$, shows no significant differences between the two schemes. Fig. 14 plots the L_2 norm of the relative error as a function of the time step $\Delta\tau$ on a logarithmic scale for several Mach numbers. The BDF2 exhibit a curve slope equal to 1.8, that is close to the expected value, while for the BDF3 scheme the average slope is equal to 2.7, up to a relative error of about 10^{-5}. Convergence rapidly degrades for even smaller time step; however, from fixed grid computations on unsteady problems, the flattening of the convergence curve at lower time steps is believed to be related to the (first-order) implementation of the boundary condition in the Eulerian code and not to the ALE formulation itself. A similar error analysis has been performed using a parabolic interpolation in time of the node positions, namely, using expressions (45) for the interface and boundary velocities. The adoption of the parabolic interpolation is not found to produce significant improvements with respect to the linear one, as it can be appreciated in Fig. 14.

Figs. 15 compares a uniform time step solution with the corresponding variable

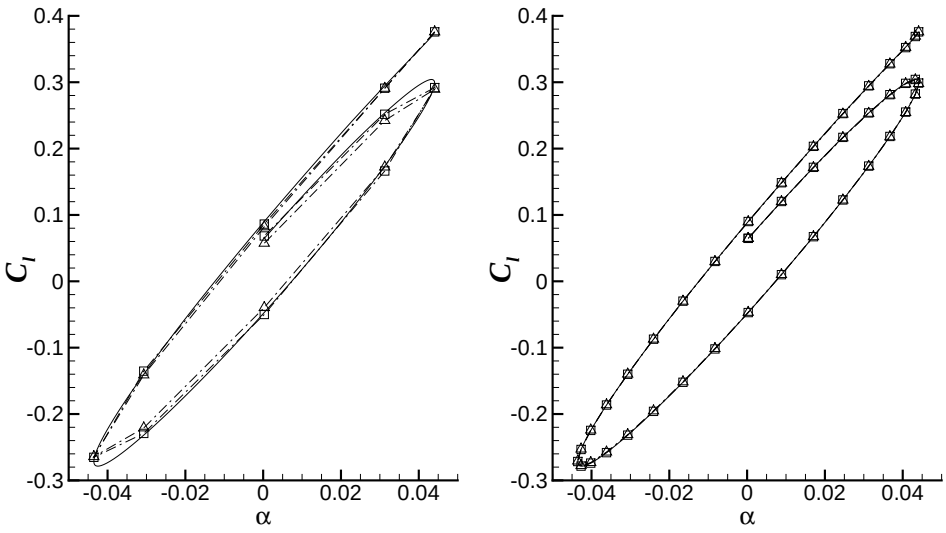


Fig. 12. Lift coefficient as a function of the angle of attack for Mach = 0.55. BDF2 and BDF3 solutions obtained using 8 steps (a) and 32 steps (b) per period.

variable time step solution obtained from the BDF3 scheme and using the same number of time steps, In the variable step case the time step amplitude is chosen as a function of the slope variation of the input signal, namely the motion law imposed to the airfoil, assuming a negligible delay between the input and the aerodynamics response. This assumption can be considered valid if the reduced frequency k is small enough, as it is the case here with $k = 0.016$. For higher reduced frequencies a more effective strategy for the choice of the time step needs to be implemented. The error analysis performed on the variable time step scheme gives a relative L_2 error of 3.96×10^{-5}, while the fixed step scheme resulted in an error of 4.67×10^{-5}.

6.2 Flutter analysis of the Agard 445.6 wing

The occurence of strong nonlinearities in the flow field, such as for example shock waves or flow separation, requires for the adoption of CFD methods for aeroelastic stability assessment in the transonic regime. However, it can be observed that while the steady flow fields is highly nonlinear, the same does not hold for the unsteady loads which influences the aeroelastic stability. The latter can be often considered with an high level of accuracy as linear [39,40]. In any case, nonlinearities in the flow equations require to study the stability of each flight configuration independently. To speed up the analysis, especially when a large number of configurations needs to be tested, a linearized model of the unsteady aerodynamic forces is extracted from CFD solutions by evaluating the aerodynamic response to relevant modal deformations of the structure, without the need of performing a fully coupled nonlinear analysis for each test

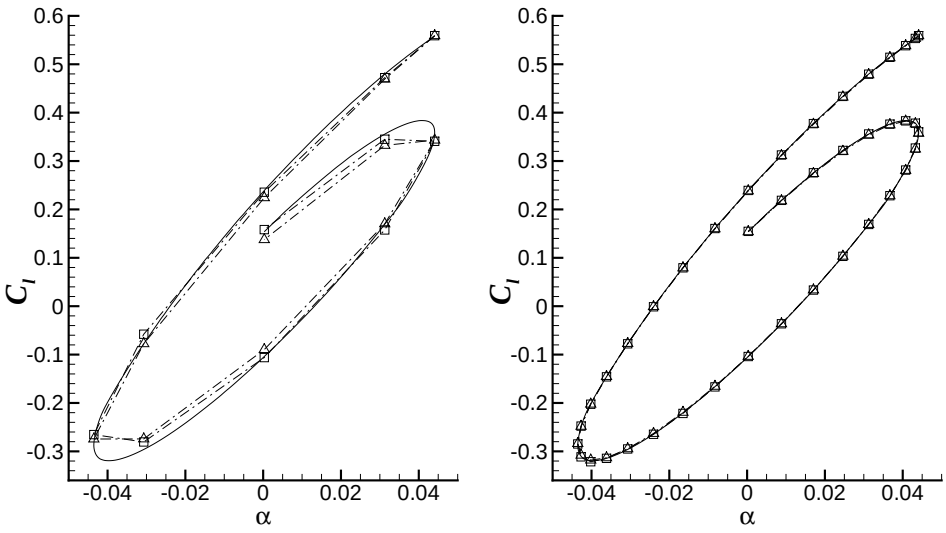


Fig. 13. Lift coefficient as a function of the angle of attack for Mach = 0.755. BDF2 and BDF3 solutions obtained using 8 steps (a) and 32 steps (b) per period.

condition. The result of the linearisation is a Reduced Order Model (ROM) for aerodynamic unsteady forces. It is therefore possible to use CFD–ALE time marching solutions as a sort of "numerical experiments" for the extraction of the dynamics of the flow field. To this purpose it is necessary to run a set of specified simulation with imposed wall boundary movements, choosing a simple excitation method which requires a reasonable computational cost but permits a good identification of the principal dynamics of aerodynamic forces $\boldsymbol{F}_a$. The details of the procedure adopted here for the identification of state space linear ROM can be found in Cavagna, Quaranta and Mantegazza [41].

A linear modal representation of the structure is used, as it is usually done in classic aeroelastic analysis [1]. In this case, the classical flutter problem can be stated as follows: find the dynamic pressure q which gives rise to unstable free movements for a given (linear) elastic structure represented in modal form as

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{K}\boldsymbol{q} = \frac{1}{2}\rho u^2 \boldsymbol{F}_a(\boldsymbol{q}, Mach), \tag{62}$$

where $\boldsymbol{M}$, $\boldsymbol{C}$, $\boldsymbol{K}$ are respectively the modal mass, damping and stiffness matrices, and $\boldsymbol{F}_a$ are the generalized aerodynamic forces associated with each mode. Note that by coupling the identified dynamic system with Equation (62), the flutter problem becomes a standard eigenvalue problem [42].

As an application to the CA scheme outlined above, the Agard 445.6 deformable wing test case is now considered. Experimental results for the wind tunnel tests can be found in the report [43]. The tested wing presents a clear drop of the flutter velocity in transonic flow conditions, so it has been taken

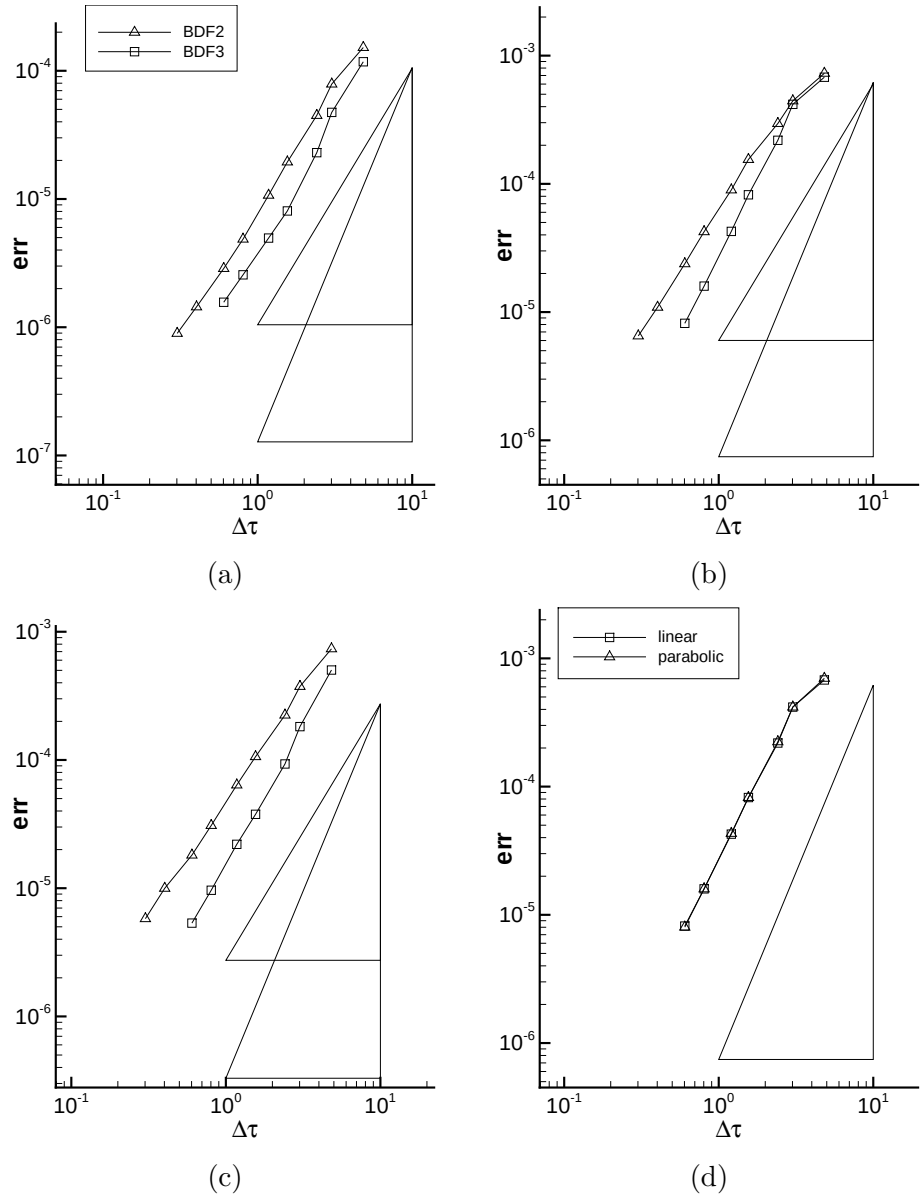


Fig. 14. Comparison of BDF2 and BDF3 convergence curves for different Mach numbers, namely, 0.55 (a), 0.755 (b) and 1.114 (c). d) Comparison between the linear and parabolic mesh node positions for the BDF3 scheme and for Mach number 0.755.

by many authors as the reference case to assess the quality of the transonic flutter prediction [44–46]. For numerical flutter computations only the first two modes are taken into account, where the first is a bending mode while the second is a torsional one (cf. Fig. 16), as they are deemed sufficient to predict the onset of flutter instability. A three-dimensional tetrahedral fluid mesh, containing 22 014 nodes is used as the aerodynamic grid. The input signal given to the aerodynamic system, represented by the modal deformation

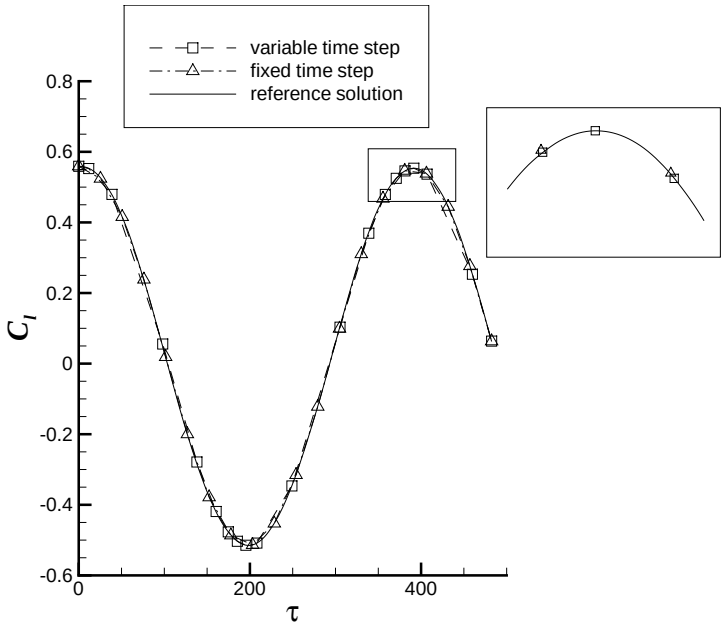


Fig. 15. Comparison between fixed and variable time-step BDF3 solutions.

amplitude imposed to the boundary of fluid grid, is of the form:

$$q_i = \begin{cases} A_{\infty,i}(1 - \cos(k\tau))/2, & \text{for} \quad \tau < \dfrac{\pi}{k_{\max}} \\ A_{\infty,i}, & \text{for} \quad \tau \geq \dfrac{\pi}{k_{\max}} \end{cases}$$

where $k_{\max}$ is the highest reduced frequency of interest and $A_{\infty,i}$ is the amplitude of the i-th mode ($A_{\infty,i} \simeq 0.01$ here). Fig. 16 shows the maximum deformations reached during the simulation and corresponding Mach contours for the first and second modes. The results of the simulation with imposed wall boundary movements along the first and the second mode are summarized in Figs. 17 and 18, where the real and imaginary part of the frequency response matrix of the generalized aerodynamic forces are shown. The plot illustrates the large differences in the coefficient trends at different mach number. These results are then used to compute the flutter onset point.

Flutter analysis are performed for several Mach numbers, namely in the subsonic, transonic and supersonic regimes. The result of the numerical analysis are summarized in Fig. 19 and Table 6.2. The plot shows the flutter index, defined as

$$I_{\mathrm{f}} = \frac{v_{\mathrm{f}}}{c\,\omega_{\mathrm{a}}\sqrt{\mu}},$$

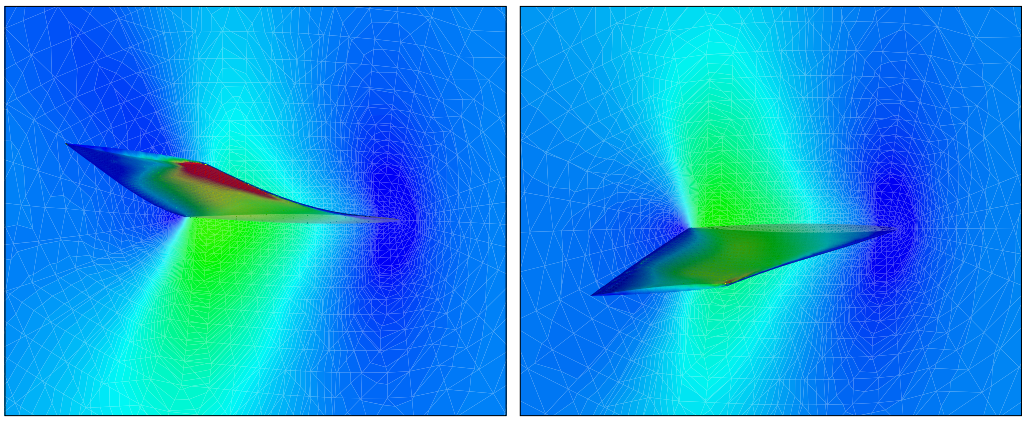
where v_{f} is the flutter velocity, c is half of the wing chord at the root, ω_{a} is the frequency of the torsional mode and μ is the mass ratio, i.e. the ratio between the structural mass and the mass of the equivalent volume of fluid at reference density. The agreement with experimental results for subsonic and

Mach	ρ [kg/m^3]	v_f [m/s]		I_f	
		BDF3	Exp.	BDF3	Exp.
0.678	0.208	226.46	231.37	0.408	0.417
0.901	0.099	290.78	296.76	0.363	0.370
0.960	0.063	306.24	309.08	0.306	0.308
1.072	0.056	430.64	344.81	0.399	0.320
1.140	0.078	535.56	364.42	0.592	0.403

Table 1
Numerical and experimental flutter results.

transonic flows is good, with a clear evidence of the flutter dip phenomenon at Mach 0.96. For supersonic cases the error on the flutter index is higher. However, the matching for supersonic cases is a challenging problem, and similar results have been obtained by other researches [45–47], To test the effect of moving the far-field boundaries, two transient simulation have been performed with a sinusoidal variation of the wing nodes position along the first mode of vibration; in the first one the far field nodes are kept fixed, while in the second one they are allowed to move freely following the grid deformation. Fig. 20 shows the lift coefficient C_ℓ vs. non dimensional time τ for the two cases showing that the movement of the far-field boundary, while limiting the element distortions, does not affect the solution.

The relative L_2 norm of the difference between the two solutions at the end of the simulation, when the undeformed conditions is reached again, reads $||\Delta \mathsf{u}||_{L_2} \,/\, ||\mathsf{u}||_{L_2} = (1.06, 1.09, 1.40, 1.02, 0.96) \times 10^{-5}$.



(a) (b)

Fig. 16. First (a) and second (b) modal deformations and relative Mach contours with Mach = 0.96.

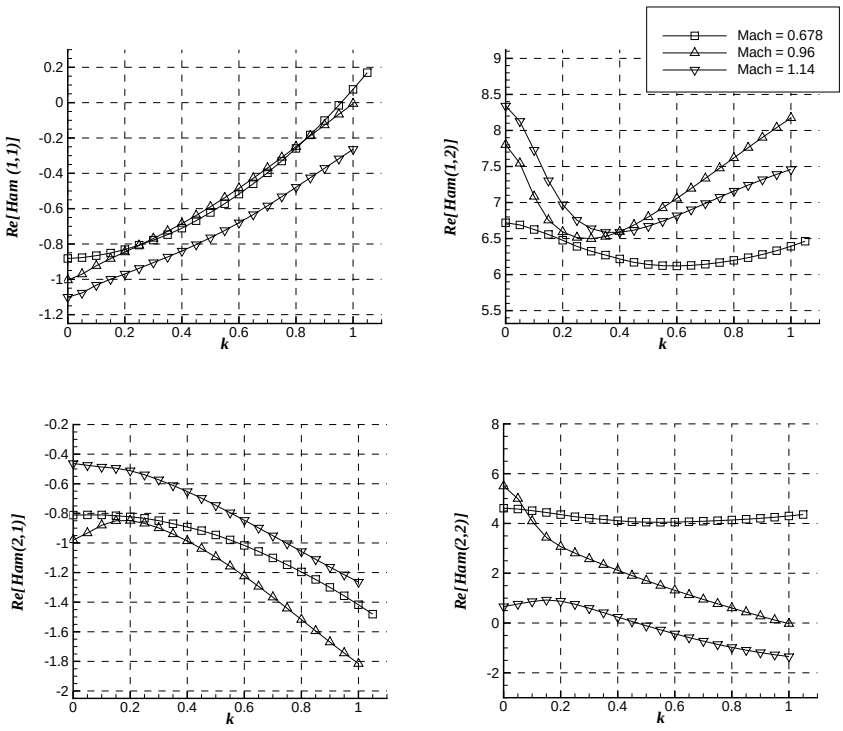


Fig. 17. Real part of the frequency response matrix of the generalized aerodynamic forces for several Mach numbers for the AGARD 445.6 test case.

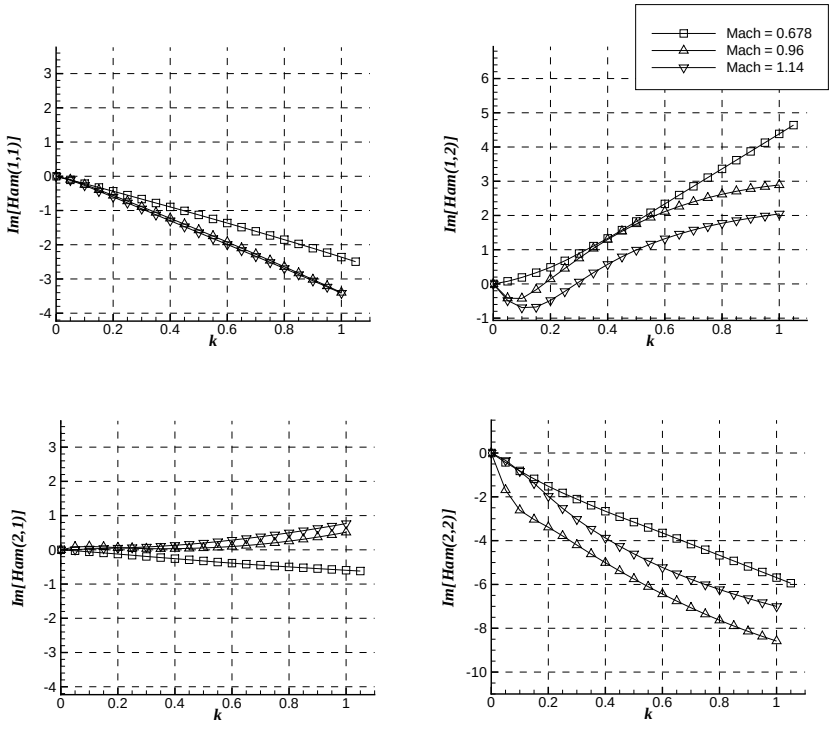


Fig. 18. Imaginary part of the frequency response matrix of gthe eneralized aerodynamic forces for several Mach numbers for the AGARD 445.6 test case.

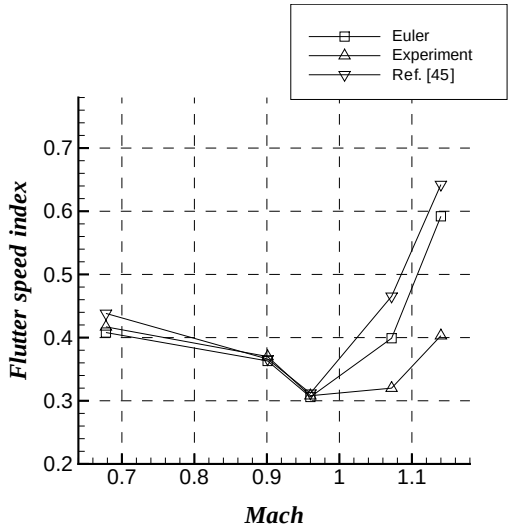


Fig. 19. Flutter speed index.

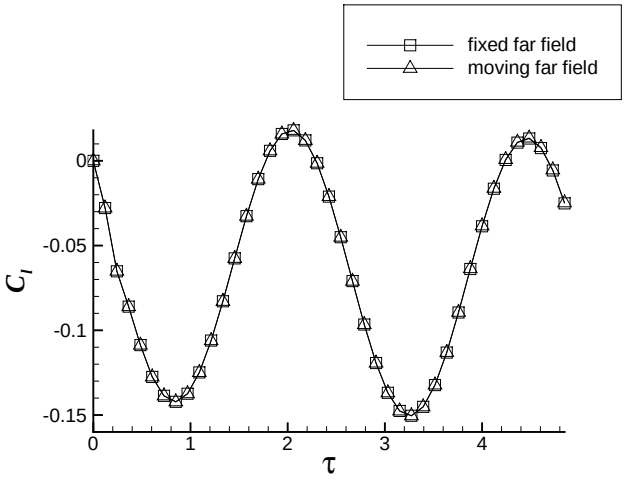


Fig. 20. Comparison between fixed and deformable far field.

7 Conclusions

A general representation of the advective fluxes for the ALE formulation has been given, which provides a simple way to extend virtually any spatial discretization scheme from fixed to moving grids. Moreover, the transformation matrix defined by (22) allows for the correct imposition of boundary conditions on moving solid walls and on the far field, thus resulting in more flexibility for the grid deformation strategy.

A novel compatibility condition relating the velocity at the interface between adjacent control volumes to the time derivatives of the volume themselves is introduced. The Interface Velocity Consistency condition amounts to write an additional conservation law for these quantities, which is to be solved together

with the flow equations, and it represents an extension of the well-known Geometric Conservation Law to the case of nonuniform flow fields.

Numerical results for a two-dimensional test case confirm the accuracy of the present approach for different time integration schemes including variable time step size. The simulations are only slightly affected by the chosen interpolation law for the grid velocity. As a consequence, a linear interpolation of nodes positions is sufficient to achieve the designed time accuracy, allowing a more efficient algorithm for advancing in time. Three-dimensional flutter analysis for a standard test case are in good agreement with available experimental results.

The extension of the conservation laws system (21) to the case of viscous flows will be straightforward as the viscous fluxes do not depend on the deformation velocity of the mesh but depend only on mesh configuration at a given time. As a result, all the relevant features underlined in the present report can be easily extended to viscous flow models.

References

[1] R. L. Bisplinghoff, H. Ashley, Principles of Aeroelasticity, Wiley & sons, New York, 1962.

[2] R. M. Bennet, J. W. Edwards, An overview of recent developments in computational aeroelasticity, in: Proceedings of the 29th AIAA Fluid Dynamics Conference, Albuquerque, NM, 1998.

[3] D. M. Schuster, D. D. Liu, L. J. Huttsell, Computational aeroelasticity: Success, progress, challenge, Journal of Aircraft 40 (5) (2003) 843–856.

[4] R. Yurkovich, Status of unsteady aerodynamic prediction for flutter of high-performance aircraft, Journal of Aircraft 40 (5) (2003) 832–842.

[5] J. Donea, Arbitrary lagrangian–eulerian finite element methods, in: T. Belytschko, T. J. Hughes (Eds.), Computational Methods for Transient Analysis, Elsevier Science Publisher, Amsterdam, The Netherlands, 1983, Ch. 10, pp. 474–516.

[6] F. Nobile, Numerical approximation of fluid–structure interaction problems with application to hemodynamics, Ph.D. thesis, École Polytechnique Fédérale de Lausanne, Switzerland (September 2001).

[7] P. Geuzaine, C. Grandmont, C. Farhat, Design and analysis of ALE schemes with provable second-order time-accuracy for inviscid and viscous flow simulations, Journal of Computational Physics 191 (2003) 206–227.

[8] C. Farhat, P. Geuzaine, Design and analysis of robust ale time-integrators for the solution of unsteady flow problems on moving grids, Computer Methods in Applied Mechanics and Engineering 193 (2004) 4073–4095.

[9] D. Mavriplis, Z. Yang, Construction of the discrete geometric conservation law for high-order time-accurate simulations on dynamic meshes, Journal of Computational Physics 213 (3) (2006) 557–573, 2006.

[10] T. Fanion, M. A. Fernández, P. Le Tallec, Deriving adequate formulations for fluid-structure interaction problems: from ALE to transpiration, Rev. Européenne Élém. Finis 9 (6–7) (2000) 681–708.

[11] P. D. Thomas, C. K. Lombard, Geometric conservation law and its application to flow computations on moving grids, AIAA J. 17 (1979) 1030–1037.

[12] S. Piperno, C. Farhat, Partitioned procedure for the transient solution of coupled aeroelastic problems. Part II: Energy transfer analysis and three dimensional applications, Computer Methods in Applied Mechanics and Engineering 190 (2001) 3147–3170.

[13] D. J. Mavriplis, Z. Yang, Construction of the discrete geometric conservation law for high-order time-accurate simulations on dynamic meshes, J. Comput. Phys. 213 (2006) 557–573.

[14] E. Godlewski, P. A. Raviart, Numerical approximation of hyperbolic systems of conservation laws, Springer-Verlag, New York, 1994.

[15] R. J. LeVeque, Finite volume methods for conservation laws and hyperbolic systems, Cambridge University Press, 2002.

[16] P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. 43 (1981) 357–372.

[17] B. van Leer, Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second order scheme, J. Comput. Phys. 14 (1974) 361–370.

[18] N. Weatherill, O. Hassan, M. Marchant, D. Marcum, Adaptive inviscid solutions for aerospace geometries on efficiently generated unstructured tetrahedral meshes, aIAA Paper 93-3390 (1993).

[19] V. Selmin, The node-centred finite volume approach: bridge between finite differences and finite elements, Comp. Meths. Appl. Mech. Eng. 102 (1993) 107–138.

[20] A. Guardone, L. Quartapelle, Spatially factorized Galerkin and Taylor-Galerkin schemes for multidimensional conservation laws, Scientific Report DIA-SR 00-18, Politecnico di Milano, Italy (2000).

[21] V. Selmin, L. Formaggia, Unified construction of finite element and finite volume discretizations for compressible flows, Int. J. Numer. Meth. Eng. 39 (1996) 1–32.

[22] J. Donea, An arbitrary Lagrangian-Eulerian finite element method for transient fluid–structure interactions, Comput. Methods Appl. Mech. Engrg. 33 (1982) 689–723.

[23] J. Donea, A. Huerta, J.-P. Ponthot, A. Rodríguez-Ferran, Arbitrary Lagrangian-Eulerian methods, in: R. Stein, E. de Borst, T. Hughes (Eds.), The Encyclopedia of Computational Mechanics, Vol. 1, Wiley, 2004, Ch. 14, pp. 413–437.

[24] V. Venkatakrishnan, D. J. Mavriplis, Implicit method for the computation of unsteady flows on unstructured grids, J. Comput. Phys. 127 (1996) 380–397.

[25] B. Nkonga, H. Guillard, Godunov type method on non-structured meshes for three-dimensional moving boundary problems, Comp. Meth. Appl. Mech. Engng. 113 (1994) 183–204.

[26] V. Selmin, Coupled fluid/structure system, in: W. Haase (Ed.), Progress in Computational Flow-Structure Interaction, Vol. 81 of Notes on numerical fluid mechanics and multidisciplinary design, Springer-Verlag, Berlin, 2002.

[27] B. Koobus, C. Farhat, Second-order time accurate and geometrically conservative implicit schemes for flow computations on unstructured dynamic meshes, Comput. Methods Appl. Mech. Engrg. 170 (1999) 103–129.

[28] M. R. Visbal, D. V. Gaitonde, The use of high-order finite-differences schemes on curvilinear deforming meshes, J. Comput. Phys. 181 (2002) 155–185.

[29] P. Geuzaine, C. Grandmont, C. Farhat, Design and analysis of ALE schemes with provable second-order time-accuracy for inviscid and viscous flow simulations, J. Comput. Phys. 191 (2003) 206–227.

[30] H. Guillard, C. Farhat, On the significance of the geometric conservation law for flow computations on moving meshes, Comput. Methods Appl. Mech. Engrg. 190 (2000) 1467–1482.

[31] C. Farhat, P. Geuzaine, C. Crandmont, The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids, J. Comput. Phys. 174 (2001) 669–694.

[32] O. Hassan, K. A. Sørensen, K. Morgan, N. P. Weatherill, A method for time accurate turbulent compressible fluid flow simulation with moving boundary components employing local remeshing, Int. J. Num. Meth. Fluids. In press.

[33] G. Quaranta, P. Masarati, P. Mantegazza, A conservative mesh-free approach for fluid-structure interface problems, in: M. Papadrakakis, E. Oñate, B. Schrefler (Eds.), International Conference on Computational Methods for Coupled Problems in Science and Engineering, CIMNE, Santorini, Greece, 2005.

[34] C. Byun, G. Guruswamy, A parallel, multi-block, moving grid method for aeroelastic applications on full aircraft, Tech. Rep. 98-4782, AIAA (September 1998).

[35] J. Reuther, A. Jameson, J. Farmer, L. Martinelli, D. Saunders, Aerodynamics shape optimization of complex aircraft configurations via an adjoint formulation, AIAA Paper 96-0094, AIAA (January 1996).

[36] J. Batina, Unsteady euler airfoil solution using unstructured dynamic meshes, AIAA Journal 28 (1990) 1381–1388.

[37] C. Degand, C. Farhat, A three-dimensional torsional spring analogy method for unstructured dynamic meshes, Computers and Structures 80 (2002) 305–316.

[38] T. Belytschko, W. K. Liu, B. Moran, Nonlinear Finite Elements for Continua and Structures, John Wiley & Sons, Chichester, UK, 2000.

[39] D. E. Raveh, Computational-fluid-dynamic-based aeroelastic analysis and structural design optimization—a researcher's perspective, Computer Methods in Applied Mechanics and Engineering 194 (2005) 3453–3471.

[40] L. Cavagna, G. Quaranta, G. L. Ghiringhelli, P. Mantegazza, Efficient application of CFD aeroelastic methods using commercial software, in: International Forum on Aeroelasticity and Structural Dynamics IFASD-2005, Munich, Germany, 2005.

[41] L. Cavagna, G. Quaranta, P. Mantegazza, Application of Navier-Stokes simulations for aeroelastic assessment in transonic regime, to be published.

[42] G. Pasinetti, P. Mantegazza, Single finite states modeling of aerodynamic forces related to structural motions and gusts, AIAA Journal 37 (5) (1999) 604–612.

[43] E. C. Yates, AGARD standard aeroelastic configurations for dynamic response. I wing 445.6, R 765, AGARD (1985).

[44] E. M. Lee-Raush, J. T. Batina, Wing flutter boundary prediction using unsteady Euler aerodynamic method, AIAA Paper 93-1422 (April 1993).

[45] F. Liu, J. Cai, Y. Zhu, H. Tsai, A. Wong, Calculation of wing flutter by a coupled fluid-structure method, Journal of Aircraft 38 (2) (2001) 334–342.

[46] R. Gordnier, R. Melville, Transonic flutter simulations using an implicit aeroelastic solver, Journal of Aircraft 37 (5) (2000) 872–879.

[47] Z. Yang, D. Mavriplis, Higher-order Time Integration Schemes for Aeroelastic Applications on Unstructured Meshes, in: 44^{th} AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2006.