V European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2010
J. C. F. Pereira and A. Sequeira (Eds)
Lisbon, Portugal,14-17 June 2010

NUMERICAL SIMULATION OF THE OPENING OF AERODYNAMIC CONTROL SURFACES WITH TWO-DIMENSIONAL UNSTRUCTURED ADAPTIVE MESHES

Giuseppe Quaranta, Dario Isola and Alberto Guardone

Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano,
Via La Masa, 34, 20156 Milano, Italy
e-mail: giuseppe.quaranta@polimi.it, dario.isola@polimi.it, alberto.guardone@polimi.it

Key words: Compressible flows, Spoilers, Dynamic meshes, Grid adaptation, ALE schemes, Finite Volume schemes

Abstract. *A novel two-dimensional numerical scheme for the compressible Euler equations on dynamic meshes is presented. The proposed approach allows to perform computations on moving meshes with adaptation, which is required to preserve the mesh spacing for large boundary displacement. At each time level, the new mesh is obtained by a suitable combination of "elastic" deformation, edge-swapping, grid refinement and de-refinement. The grid modifications—including topology modification due to edge-swapping or the insertion/deletion of new grid nodes—are interpreted at the flow solver level as continuous (in time) deformations of suitably-defined node-centred finite volumes. The solution over the new mesh is obtained without explicitly resorting to interpolation techniques, since the definition of suitable interface velocities allows one to determine the new solution by simple integration of the Arbitrary Lagrangian-Eulerian formulation of the flow equations. Moreover, n-step Backward Differencing Formulæ can be implemented in a straightforward manner, without the need of performing backward interpolation over the n-th previous grids. Numerical simulations of the opening process of the flap in a two-element high-lift configurations are carried out to test the proposed technique. Numerical results includes also the computation of dynamic loads on the airfoil due to the opening of a spoiler, in which the flow-field is characterized by unsteady flow separation at the movable surface trailing edge.*

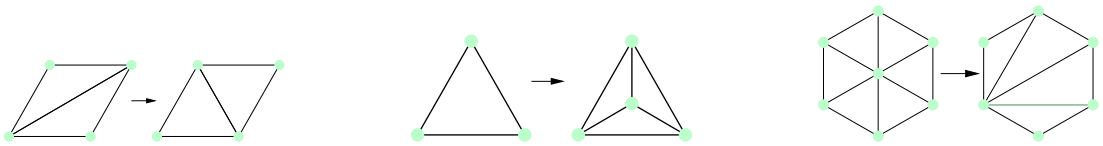
1 INTRODUCTION

The correct investigation of hinge moments during the opening of control surfaces such as aileron, high-lift devices and spoilers is of the utmost importance in the preliminary design of aircraft, not only for the overall aerodynamic behaviour, but also for the general aircraft sizing. To improve the current design of these kind of aerodynamic appendices, high fidelity (Computational Fluid Dynamics) CFD models can be used. Additionally, movable surfaces are used as means for aeroelastic control systems,[1] so time accurate CFD simulation may be required for the correct assessment of the unsteady aerodynamic loads.

The numerical simulation of compressible flows around moving bodies is a very complex task, since the computational domain is continuously changing its shape. So, a new grid complying with the modified geometry must be build at each time level. For small displacements it is usually sufficient to deform the initial grid into the new geometry, without changing the grid topology; very efficient algorithms have been obtained in these cases.[2,3] The computational mesh moves during the simulation so, the standard finite volume Eulerian formulation of the flow equations must be dropped in favour of the Arbitrary Lagrangian–Eulerian (ALE) approach, in which the control volumes are allowed to change in shape and position as time evolves.[4] When large deformation of topology changes are required, grid deformation techniques cannot be used anymore, since they would lead to poor grid quality and possibly to invalid negative-volume elements. Overset grid techniques can be successfully applied in this case.[5] This approach requires to adopt complex interpolation algorithms between the different grids representing the domain and user intervention is often required. An alternative technique is the complete mesh regeneration, or *re-meshing*, which requires to interpolate the solution over the new grid at each time level.[6] Moreover, in order to use multi-step high-order time integration schemes, such as the Backward Difference Formulæ, re-interpolation is needed to compute the backward solutions at each time level over the current grid. Consequently, the computational burden of this approach could be very large.

In order to circumvent the difficulties of both overset grid and re-interpolation techniques, a method based on mesh deformation with local edge swapping has been proposed.[7] This method guarantees high quality grids also in the case of large grid displacements and changes in the topology, without resorting to re-meshing. Modifications of the grid topology due to swapping are interpreted as grid deformations, thus the solution at the current time level is computed by employing standard ALE techniques.

The main hitch of the method proposed by Muffo et al.[7] is in the decrease of grid quality as the movement proceeds. Furthermore, there is no direct control of grid spacing: for example the preservation of a fine region around a moving body. Therefore arbitrary large movements are not allowed and some cases of interest, where a local control of size of the elements is required (e.g. flap or spoiler extraction), could not be easily tackled. In the present work simple mesh adaptation techniques are introduced to obtain a complete



(a) Edge swapping (b) Node insertion (c) Node deletion

Figure 1: Grid adaptation techniques: edge-swapping for mesh quality improvement (a), node insertion for grid refinement (b) node deletion for grid de-refinement (c).

control on grid spacing. The interpretation of grid operations as continuous deformation, first introduced by Muffo et al.,[7] is here extended to the case of node insertion and deletion. Resorting again to the ALE formulation, no interpolation scheme is needed and high order time integration schemes can be implemented easily. This results in a novel strategy for the simulation of unsteady aerodynamics on moving domains with variable connectivity and number of nodes.

2 GRID DISPLACEMENT AND ADAPTATION

In dynamic mesh simulations the internal or domain elements are to be modified to comply with the motion of the boundary elements. The displacement of the internal element nodes can be obtained using different strategies. In non-structured meshes, the most common approaches are based on the so-called elastic analogy. For example, Batina represents each side of the grid as a spring with a nonlinear stiffness proportional to the edge length,[8] whereas Degand and Farhat introduced additional torsional springs at each vertex.[2] The grid deformation algorithm used in this work resorts to a representation of domain element as deformable continuum bodies. The adoption of a local Young modulus proportional to the minimal dimension of each element results in a robust method. In this way small elements are stiff and tend to move rigidly, leaving the burden to absorb the required deformations on the larger elements, usually located far from the regions of interest.[3]

When large boundary movements occur distorted and tangled elements may appear, leading to large numerical errors, and it becomes necessary to adopt a technique to improve the mesh quality. A very simple, but extremely effective, technique to improve the quality of triangular meshes, without inserting new vertices, is the edge swapping technique.[9] Edge swapping has been used in connection with deforming meshes in order to reduce element stretching disconnecting two vertices that move with different velocity, allowing large domain deformations.[7,10]

For very large grid displacements, as it is the case of the opening of aerodynamic surfaces, mesh deformation and edge swapping alone are not sufficient to maintain high quality together with a suitable grid spacing. In these cases a different approach is needed: e.g. complete regeneration, local remeshing or element refinement/derefinement. To reduce the complexity of the mesh adaptation scheme, the use of simple elementary oper-

ations, such as nodes insertion/deletion, has been preferred to total or partial remeshing which usually involves a larger computational burden. In Fig. 1 the implemented techniques are schematically illustrated: edge swapping, mesh refinement, that is performed inserting a node in the centre of mass of the triangle, and mesh derefinement, which requires to reconstruct the connectivity inside the nodes patch surrounding the deleted vertex. Refinement and derefinement procedures are both followed by an edge swapping cycle: indeed, the used node insertion technique alone does not consent to reduce the maximum edge of a triangle, whereas the connectivity reconstruction subsequent to a node deletion does not necessarily represent the one that maximize the local quality amongst the possible configurations.

In the present work, the local dimensions of the grid element are linearly proportional to the distance from boundaries. Defining a suitable size function $h_b(\boldsymbol{x})$ for every boundary, e.g. far field or body surface, the dimension to be imposed to the reticulation will be chosen as $h(\boldsymbol{x}) = \min_{b\in\mathcal{B}} h_b(\boldsymbol{x})$, where $\mathcal{B}$ is the set of boundaries that discretize $\partial\Omega$.

3 EDGE-BASED FINITE VOLUME ALE SOLVER

The Euler equations for compressible inviscid flows in an Arbitrary Lagrangian Eulerian (ALE) framework[11,12] reads

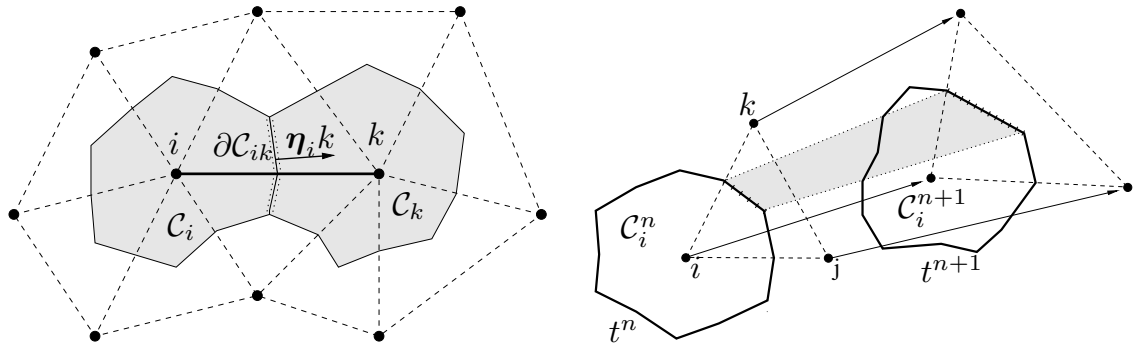
$$\frac{d}{dt}\int_{\mathcal{C}(t)} \mathsf{u} + \oint_{\partial\mathcal{C}(t)} \left[\mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v}\right]\cdot\boldsymbol{n} = \mathsf{0},\ \forall\mathcal{C}(t)\subseteq\Omega(t), \tag{1}$$

where u, $\mathsf{u} = (\rho, \boldsymbol{m}, E^{\mathrm{t}})^{\mathrm{T}} \in \mathbb{R}^+\times\mathbb{R}^3$, being ρ the density of mass, $\boldsymbol{m}$ the linear momentum vector and E^{t} the total energy per unit volume. The solution is sought for in the spatial domain $\Omega\in\mathbb{R}^2$ of boundary $\partial\Omega$ and $\forall\, t\in\mathbb{R}^+$. System (1) is to be made complete by specifying suitable initial and boundary conditions.[13] In Eq. (1), $\mathbf{f} = (\mathsf{f}_x, \mathsf{f}_y)^{\mathrm{T}} \in \mathbb{R}^4\times\mathbb{R}^2$ is the flux function defined as $\mathbf{f}(\mathsf{u}) = \left(\boldsymbol{m},\ \boldsymbol{m}\otimes\boldsymbol{m}/\rho + P(\mathsf{u})\,\mathbf{I}^2,\ \left[E^{\mathrm{t}}+P(\mathsf{u})\right]\rho/\boldsymbol{m}\right)^{\mathrm{T}}$, where P is the fluid pressure, $\mathbf{I}^{\mathrm{d}}$ indicates the identity matrix of dimensions $\mathrm{d}\times\mathrm{d}$, the scalar product in Eq. (1) is computed as $\mathbf{f}(\mathsf{u})\cdot\boldsymbol{n} = \mathsf{f}_x n_x + \mathsf{f}_y n_y$. The vector $\boldsymbol{n} = \boldsymbol{n}(\boldsymbol{s},t) = (n_x, n_y)^{\mathrm{T}}$ is the outward unit normal to the boundary $\partial\mathcal{C}(t)$ of the control volume $\mathcal{C}(t)$ and it is a function of the curvilinear coordinate $\boldsymbol{s}$ along $\partial\mathcal{C}$ and of the time as well. The term $\mathsf{u}\,\boldsymbol{v} = (\rho\boldsymbol{v}, \boldsymbol{m}\otimes\boldsymbol{v}, E^{\mathrm{t}}\boldsymbol{v})^{\mathrm{T}}$, with $\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{s},t)$ the local velocity of $\partial\mathcal{C}(t)$, keeps into account the flux contribution due to the movement of the control volume $\mathcal{C}(t)$.

The finite volume discrete counterpart of the Euler Eq. (1) is obtained by selecting a finite number of non overlapping volumes $\mathcal{C}_i(t)\subset\Omega(t)$, such that $\bigcup_i \mathcal{C}_i(t) \equiv \Omega(t)$. According to the node-centered approach considered here, each finite volume $\mathcal{C}_i$ surrounds a single node i of the triangulation of Ω, as shown in Fig. 2. Over each finite volume, Eq. (1) reads

$$\frac{d[V_i\,\mathsf{u}_i]}{dt} = -\oint_{\partial\mathcal{C}_i}\left[\mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v}\right]\cdot\boldsymbol{n}_i, \quad \forall i\in\mathcal{K}, \tag{2}$$

where $\mathcal{K}$ is the set of all nodes of the triangulation, V_i is the volume (area in two dimen-



(a) (b)

Figure 2: Left: 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. Right: geometrical interpretation of the interface velocity as the area swept by the finite volume interface during the time step (only the portion of the interface i-k associated with element i-j-k is shown.)

sions) of $\mathcal{C}$ and $\boldsymbol{n}_i = \boldsymbol{n}_i(\boldsymbol{s}, t)$ denotes the outward normal with respect to the volume $\mathcal{C}_i$, see Fig. 2. 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 (2) is now rearranged to put into evidence the boundary contribution, namely,

$$\oint_{\partial\mathcal{C}_i} \big[\mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v}\big] \cdot \boldsymbol{n}_i = \sum_{k \in \mathcal{K}_{i,\neq}} \int_{\partial\mathcal{C}_{ik}} \big[\mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v}\big] \cdot \boldsymbol{n}_i + \int_{\partial\mathcal{C}_i \cap \partial\Omega} \big[\mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v}\big] \cdot \boldsymbol{n}_i, \tag{3}$$

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. The set $\partial\mathcal{C}_{ik} = \partial\mathcal{C}_i \cap \partial\mathcal{C}_k$ is the cell interface between the volumes $\mathcal{C}_i$ and $\mathcal{C}_k$ (Fig. 2, left). Each interface $\partial\mathcal{C}_{ik}$ is associated to the corresponding edge i-k connecting nodes i and k of the triangulation of Ω. A suitable (approximate) integrated numerical flux $\Phi \in \mathbb{R}^5$, representing the flux across the cell interface $\partial\mathcal{C}_i \cap \partial\mathcal{C}_k$, is now introduced.[14] Considering a centered approximation of the unknown and of the flux function at the cell interfaces, the domain contributions read

$$\int_{\partial\mathcal{C}_{ik}} \big[\mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v}\big] \cdot \boldsymbol{n}_i \simeq \frac{\mathbf{f}(\mathsf{u}_i) + \mathbf{f}(\mathsf{u}_k)}{2} \cdot \int_{\partial\mathcal{C}_{ik}} \boldsymbol{n}_i - \frac{\mathsf{u}_i + \mathsf{u}_k}{2} \int_{\partial\mathcal{C}_{ik}} \boldsymbol{v} \cdot \boldsymbol{n}_i = -\Phi(\mathsf{u}_i, \mathsf{u}_k, \nu_{ik}, \hat{\boldsymbol{\eta}}_{ik}, \eta_{ik}),$$

where $\boldsymbol{\eta}_{ik}$ is the integrated outward normal, η_{ik} its magnitude and $\hat{\boldsymbol{\eta}}_{ik} = \boldsymbol{\eta}_{ik}/\eta_{ik}$, and where ν_{ik} is the integrated interface velocity, that is,

$$\boldsymbol{\eta}_{ik}(t) = \int_{\partial\mathcal{C}_{ik}} \boldsymbol{n}_i \qquad \text{and} \qquad \nu_{ik}(t) = \int_{\partial\mathcal{C}_{ik}} \boldsymbol{v} \cdot \boldsymbol{n}_i. \tag{4}$$

In Fig. 2, right, the geometrical interpretation of the integrated surface velocity as the area swept by the interface during the time step is depicted. Over the boundary portion $\partial\mathcal{C}_i \cap \Omega$, $\mathsf{u} = \mathsf{u}_i$ and the boundary contribution in (3) simplifies to

$$\int_{\partial\mathcal{C}_i\cap\partial\Omega} \big[\mathbf{f}(\mathsf{u}) - \mathsf{u}\,\boldsymbol{v}\big]\cdot\boldsymbol{n}_i \simeq \mathbf{f}(\mathsf{u}_i)\cdot\int_{\partial\mathcal{C}_{ik}}\boldsymbol{n}_i - \mathsf{u}_i\int_{\partial\mathcal{C}_{ik}}\boldsymbol{v}\cdot\boldsymbol{n}_i = -\Phi^{\partial}(\mathsf{u}_i,\nu_i,\hat{\boldsymbol{\xi}}_i,\xi_i), \tag{5}$$

where $\boldsymbol{\xi}_i$ is the integrated outward normal, ξ_i is its magnitude and $\hat{\boldsymbol{\xi}}_i = \boldsymbol{\xi}_i/\xi_i$, and where ν_i is the integrated interface velocity of the i-th boundary node, with

$$\boldsymbol{\xi}_i(t) = \int_{\partial\mathcal{C}_i\cap\partial\Omega}\boldsymbol{n}_i \qquad \text{and} \qquad \nu_i(t) = \int_{\partial\mathcal{C}_i\cap\partial\Omega}\boldsymbol{v}\cdot\boldsymbol{n}_i\,. \tag{6}$$

A high-resolution integrated numerical flux based on the Total Variation Diminishing (TVD) approach is used.[15] The second order approximation Φ^{II}_{ik} is replaced by the first order Roe flux Φ^{I}_{ik} near flow discontinuities.[16] The switch is controlled by the the limiter of van Leer.[17,18] For a general, namely, not centred approximation of the numerical fluxes, one finally obtains

$$\frac{d}{dt}\left[V_i\,\mathsf{u}_i\right] = \sum_{k\in\mathcal{K}_{i,\neq}}\Phi(\mathsf{u}_i,\mathsf{u}_k,\nu_{ik},\hat{\boldsymbol{\eta}}_{ik},\eta_{ik}) + \Phi^{\partial}(\mathsf{u}_i,\nu_i,\hat{\boldsymbol{\xi}}_i,\xi_i), \qquad \forall i\in\mathcal{K} \tag{7}$$

In the next section, time discretization of the above Euler equations is detailed.

3.1 TIME INTEGRATION FOR DYNAMIC ADAPTIVE GRIDS

The fully discrete form of the Euler equations (1) is now derived. As a preliminary step, the definitions (4)b and Eq. (6)b are first recast in a differential form by introducing

$$V_{i,ik}(t_0,t) = \int_{t_0}^{t}\int_{\partial\mathcal{C}_{ik}(t)}\boldsymbol{v}(\boldsymbol{x}_s,t)\cdot\boldsymbol{n}(\boldsymbol{x}_s,t) \qquad \text{and} \qquad V_{i,\partial}(t_0,t) = \int_{t_0}^{t}\int_{\partial\mathcal{C}_i\cap\partial\Omega}\boldsymbol{v}(\boldsymbol{x}_s,t)\cdot\boldsymbol{n}(\boldsymbol{x}_s,t)\,, \tag{8}$$

where $V_{i,ik}(t_0,t)$ represent the volume swept by the piece of boundary $\partial\mathcal{C}_{ik}$ in the time interval $[t_0,t)$, see Fig. 2. Therefore, one has

$$\frac{dV_{i,ik}}{dt} = \nu_{ik}(t) \qquad \text{and} \qquad \frac{dV_{i,\partial}}{dt} = \nu_i(t)\,.$$

The ordinary differential equations above are now used to complement system (6) to give

$$\begin{cases} \dfrac{d}{dt}[V_i\,\mathsf{u}_i] = \displaystyle\sum_{k\in\mathcal{K}_{i,\neq}}\Phi(\mathsf{u}_i,\mathsf{u}_k,\nu_{ik},\hat{\boldsymbol{\eta}}_{ik},\eta_{ik}) + \Phi^{\partial}(\mathsf{u}_i,\nu_i,\hat{\boldsymbol{\xi}}_i,\xi_i), & \forall i\in\mathcal{K} \\ \dfrac{dV_{i,ik}}{dt} = \nu_{ik}\,, & \forall k\in\mathcal{K}_{i,\neq} \\ \dfrac{dV_{i,\partial}}{dt} = \nu_i\,, \end{cases} \tag{9}$$

The above ODE system can be solved by means of standard integration techniques, as detailed in the following. It is remarkable that the so-called Geometric Conservation Law[19] (GCL) is automatically fulfilled by definitions (4)b and Eq. (6). Indeed, one immediately has

$$\frac{dV_i}{dt} = \oint_{\partial \mathcal{C}_i} \boldsymbol{v} \cdot \boldsymbol{n}_i = \sum_{k \in \mathcal{K}_{i,\neq}} \frac{dV_{i,ik}}{dt} + \frac{dV_{i,\partial}}{dt} = 0.$$

Discretizing the time domain with a constant time step Δt, it is possible to compute the solution u at time t^{n+1} using a Backward Differences Formulæ (BDF) scheme of order $m+1$ to perform an integration of Eq. (9), namely

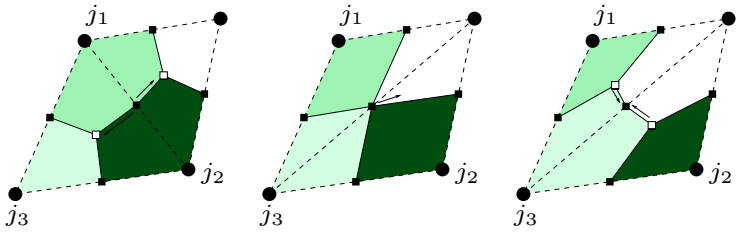
$$\begin{cases} \displaystyle\sum_{p=-1}^{m} a_p V_i^{n-p} \mathsf{u}_i^{n-p} = \left[\sum_{k \in \mathcal{K}_{i,\neq}} \Phi(\mathsf{u}_i, \mathsf{u}_k, \nu_{ik}, \hat{\boldsymbol{\eta}}_{ik}, \eta_{ik})^{n+1} + \Phi^{\partial}(\mathsf{u}_i, \nu_i, \hat{\boldsymbol{\xi}}_i, \xi_i)^{n+1} \right] \Delta t & \\ \displaystyle\sum_{p=-1}^{m-1} \alpha_p \Delta V_{i,ik}^{n-p} = \nu_{ik}^{n+1} \, \Delta t & \forall i \in \mathcal{K} \\ & \forall k \in \mathcal{K}_{i,\neq} \\ \displaystyle\sum_{p=-1}^{m-1} \alpha_p \Delta V_{i,\partial}^{n-p} = \nu_i^{n+1} \, \Delta t & \end{cases} \tag{10}$$

where $\Phi(\,\cdot\,,\,\cdot\,)^{n+1}$ is a short hand for $\Phi(\,\cdot^{\,n+1},\,\cdot^{\,n+1})$, all quantities are assumed to be known at time levels previous of $n+1$ and the grid-dependent quantities V_i^{n+1}, $\hat{\boldsymbol{\eta}}_{ik}^{n+1}$, $\boldsymbol{\eta}^{n+1}$, $\hat{\boldsymbol{\xi}}_i^{n+1}$ and ξ_i^{n+1} are computed from the (known) positions of the grid nodes at time level $n+1$. In Eq. (10) a_p represent the standard $m+1$ order BDF coefficients and $\alpha_p = \sum_{r=-1}^{p} a_r$.

The nonlinear system (10) 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, and by resorting to a dual time-stepping technique,[20] to improve the conditioning number of the Jacobian matrix.

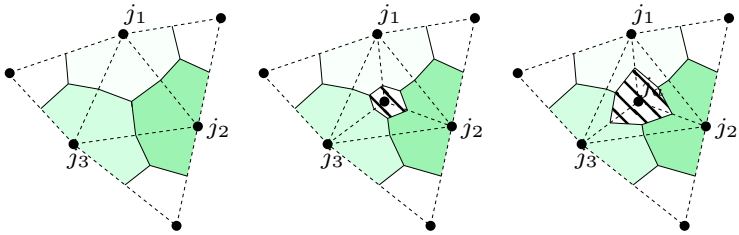
In the case of dynamic adaptive grids of interest here, the modification to the mesh connectivity, i.e. $\mathcal{K}_{i,\neq} = \mathcal{K}_{i,\neq}(t)$, and number of nodes, i.e. $\mathcal{K} = \mathcal{K}(t)$, can be interpreted as continuous deformation of suitably defined finite volume.[7] A brief overview of this interpretation is given in the following:

With reference to Fig. 3, during edge swapping, the interface $\partial\mathcal{C}_{ik}$ associated to the removed edges collapses to a single point in the time interval $[t^n, t^{n+1})$, starting from the pre-swap configuration.[7] During such operation $\partial\mathcal{C}_{ik}$ sweeps a non null volume ΔV_{ik}^{n+1} and this implies a non null interface velocity, see the second equation of system (9). Moreover the interface $\partial\mathcal{C}_{j\ell}$ associated to the inserted edge sweeps a non null volume $\Delta V_{j\ell}^{n+1}$, in order to generate the new grid configuration, starting from a single point. The geometrical interpretation of topological modifications leads to the correct computation of the volumes swept by the interfaces that satisfy Eq. (3.1).



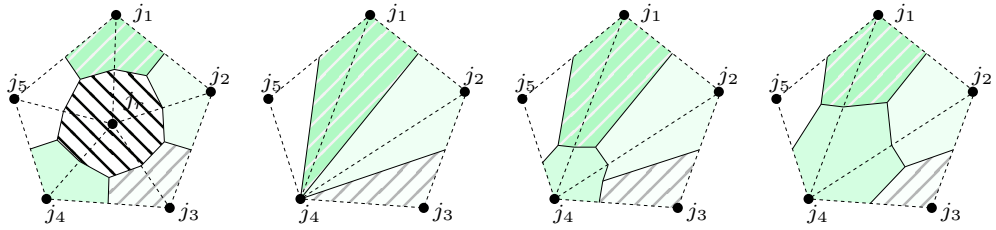
(a) t^n (b) $t^n < t < t^{n+1}$ (c) t^{n+1}

Figure 3: Interpretation of the edge swapping as continuous finite volumes deformation. Refinement. (a) Initial state. (b) Intermediate state (c) Final state.



(a) t^n (b) $t^n < t < t^{n+1}$ (c) t^{n+1}

Figure 4: Interpretation of the node insertion as continuous finite volumes deformation. Refinement. (a) Initial state. (b) Refinement of the central triangle, with insertion of node j_a. Interfaces $\partial\mathcal{C}_{12}$, $\partial\mathcal{C}_{13}$ and $\partial\mathcal{C}_{23}$ sweeps a zero volume. (c) Final state.



(a) t^n (b) $t^n < t < t^{n+1}$ (c) $t^n < t < t^{n+1}$ (d) t^{n+1}

Figure 5: Interpretation of the node deletion as continuous finite volumes deformation. Refinement. (d) Initial state, j_r is the node to be removed. (e) Node j_r collapses on j_4 with deletion of part of $\mathcal{C}_4$. (f) Cell $\mathcal{C}_r$ is annihilated then node j_r is removed and every edge that was connected with j_r is removed. (g) Regeneration of the part of $\mathcal{C}_4$ that is inside the patch. (h) Final configuration .

Fig. 4 illustrates a possible way of interpreting the node insertion operation as a continuous deformation of the finite volumes performed in the time interval $[t^n, t^{n+1})$. At time t^n, the inserted cell $\mathcal{C}_a$ has null-volume, the interfaces $\partial\mathcal{C}_{ak}$ sweep the volume ΔV_{ak}^{n+1} and then stop in the final configuration at time t^{n+1}. As in the case of the edge swapping the correct calculation of swept volumes allows to satisfy the GCL.

In Fig. 5 a possible ALE interpretation of the node deletion procedure is given. The r-th finite volume collapses on node j_4, it is removed and then the finite volumes geometry is regenerated in accordance with the new topology. The swept volumes so calculated allows to compute ν_{rk}^{n+1} associated to the interfaces of the removed cell and the related ALE fluxes: indeed, a null interface entails null Euler fluxes and generally non-null ALE fluxes. Since the numerical flux across $\partial\mathcal{C}_{rk}$ is approximated by the solution u_r^{n+1} of the removed cell and with the solution u_k^{n+1} of any adjacent cell, it is therefore necessary to keep into account the conservation equation associated to a deleted node. Adopting a $m+1$ order BDF scheme, the conservation equation for a deleted node reads

$$a_0[V_r\mathsf{u}_r]^n + a_1[V_r\mathsf{u}_r]^{n-1} + \cdots + a_{m+1}[V_r\mathsf{u}_r]^{n-m-1} = \Delta t \sum_{k \in \mathcal{K}_{r,\neq}^{[n-m,n+1)}} \Phi(\mathsf{u}_r^{n+1}, \mathsf{u}_k^{n+1}, \nu_{rk}^{n+1}, \hat{\boldsymbol{\eta}}_{rk}^{n+1}, 0)\,. \tag{11}$$

Since the left hand side (the time derivative) is known and $\eta_{rk} = 0$, Eq. (11) is an algebraic equation of balance for ALE fluxes and it is therefore not related with the Euler equations. Due to the fact that the r-th node is no longer part of the mesh, the value of u_r^{n+1} seems to be useless but it is indeed necessary to correctly balance the ALE fluxes exchanged with the other cells. Indeed u_r^{n+1} appears as well at the right hand side of equations for the nodes belonging to $\mathcal{K}_{r,\neq}^{[n-m,n+1)}$, whose cells still discretize Ω.

The above consideration on the mesh adaptation procedure allows to take into edge swapping and node insertion/deletion into the time integration procedure as follows. From Eq. (3) it follows that to a removed edge/interface $\partial\mathcal{C}_{ik}$, such as $\eta_{ik}^{n+1} = 0$, is associated a null contribution of standard Euler fluxes and a non-null contribution of ALE fluxes that are explicitly function of ν_{ik}, but not of η_{ik}. Thus to a removed edge is associated a

non null ALE flux, i.e. $\Phi(\mathsf{u}_i, \mathsf{u}_k, \nu_{ik}, \hat{\boldsymbol{\eta}}_{ik}, 0)^{n+1}$, if the related integrated interface velocity differs from zero. Introducing now the set of node $\mathcal{K}^{n+1} = \mathcal{K}(t^{n+1})$ of the triangulation of $\Omega(t^{n+1})$, $N_{\mathcal{K}}^n = \dim(\mathcal{K}^n)$ and $N_{\mathcal{K}}^{n+1} = \dim(\mathcal{K}^{n+1})$, then the total number of the inserted nodes during the time interval $[t^n, t^{n+1})$ is $N_{\text{A}} = N_{\mathcal{K}}^{n+1} - N_{\mathcal{K}}^n$. The presence of N_{A} new finite volumes imposes to write N_{A} new balance equations, thus the first equation of system Eq. (10) is to be solved for all $i \in \mathcal{K}^{n+1}$, but no further modifications of the numerical scheme are required. It is to be noted that, for the a-th cell that has been added at step $n+1$, the totality of metric quantities associated to time steps previous than $n+1$ are null, i.e. $V_a^{n-r} = 0$, $V_a^{n-r} = 0$, $\eta_{ak}^{n-r} = 0$ and $\Delta V_{ak}^{n-r} = 0$ for all $r \geq 0$. Therefore the knowledge of the values of u_a^{n-r} for $r \geq 0$ is unessential. Equation (11) becomes a trivial identity only when $\nu_{rk}^{n+1} = 0$, $\forall k \in \mathcal{K}_{r,\neq}^{[n-m,n+1)}$. Thus $m+2$ time steps after the node removal the equation can be erased from the system. Introducing $\mathcal{K}^{[n-m,n+1)} = \{k \notin \mathcal{K}^{n+1} : \exists V_k(t) \neq 0, t \in [t^{n-m}, t^{n+1})\}$, as the sets of nodes removed not before than t^{n-m}, and $\mathcal{K}^{[n-m,n+1]} = \mathcal{K}^{n+1} \cup \mathcal{K}^{[n-m,n+1)}$ it is possible to recast system (10) for grids with variable number of nodes as

$$\begin{cases} \displaystyle\sum_{p=-1}^{m} a_p V_i^{n-p} \mathsf{u}_i^{n-p} = \Bigg[\sum_{k \in \mathcal{K}_{i,\neq}^{n+1}} \Phi(\mathsf{u}_i, \mathsf{u}_k, \nu_{ik}, \hat{\boldsymbol{\eta}}_{ik}, \eta_{ik})^{n+1} \\ \qquad\qquad + \displaystyle\sum_{k \in \mathcal{K}_{i,\neq}^{[n-m,n+1)}} \Phi(\mathsf{u}_i, \mathsf{u}_k, \nu_{ik}, \hat{\boldsymbol{\eta}}_{ik}, 0)^{n+1} \Bigg] \Delta t \qquad \begin{matrix} \forall i \in \mathcal{K}^{[n-m,n+1]} \\ \forall k \in \mathcal{K}_{i,\neq}^{[n-m,n+1]} \end{matrix} \\ \displaystyle\sum_{p=-1}^{m-1} \alpha_p \Delta V_{i,ik}^{n-p} = \nu_{ik}^{n+1} \, \Delta t \end{cases} \tag{12}$$

System (12) has to be completed with proper boundary terms and IVC, as it has been done in system (10).

4 RESULTS

In the present section, numerical results of unsteady flows over dynamic adaptive grids are reported. These are the case of an oscillating airfoil moving inside a still grid and the opening of a spoiler from a two-dimensional airfoil.

4.1 Oscillating airfoil

In Fig. 6 the initial and final computational grid used in the simulation of the translating and oscillating NACA 0012 airfoil enclosed in fixed-shape external boundary are shown. This test case is the two-dimensional counterpart of the motion of a rotorblade tip which performs a half revolution around the axis.

In Fig. 7 the $C_L - \alpha$ curve of the NACA 0012 airfoil at $M = 0.755$ is shown. The computations were performed using the second- and third-order BDF time integration schemes

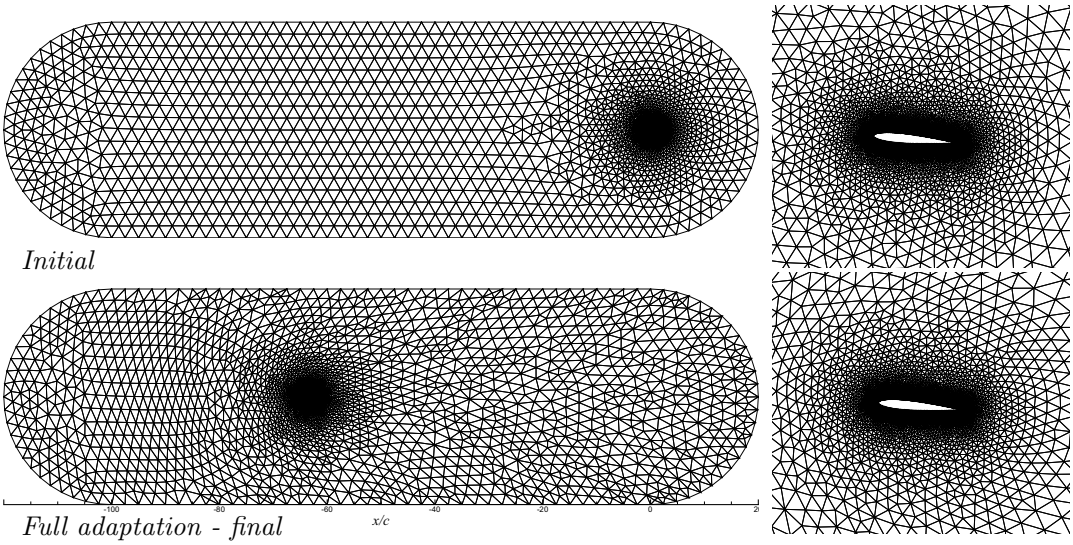


Figure 6: Computational grids for translating NACA 0012 airfoil. Top: initial grid. Bottom: final grid

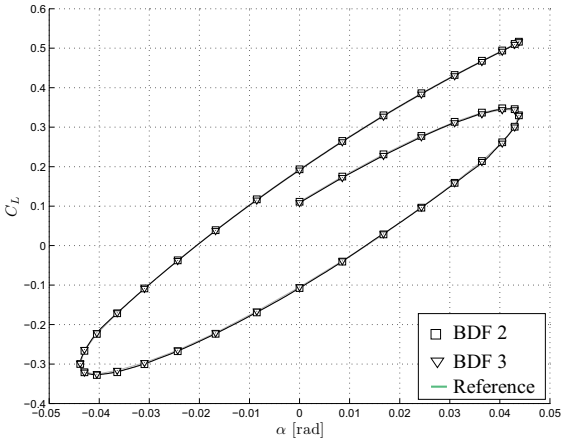


Figure 7: Lift coefficient versus incidence at M_∞=0.755. 32 steps per period.

and no adaptation. The TVD space integration scheme was used. The reference solution has been calculated in a reference moving with the airfoil. With both adopted BDF schemes, the results agree fairly well with the available reference ones thus demonstrating the validity of the proposed approach.

4.2 Opening of a spoiler

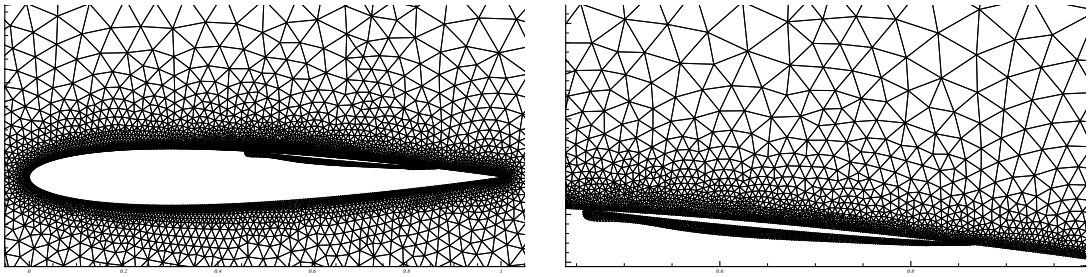
This test case considers the opening phase of a spoiler put on the on the upper surface of an airfoil. Spoilers are control surface used to decrease lift and increase drag in order to control the aircraft roll or as air-brakes. Steady state numerical simulation of three dimensional wings with open spoilers on has been performed using complex chimera techniques.[5,21] However, the time accurate dynamic simulation of the spoiler opening, even in two spatial dimensions it is a very challenging problem from the numerical standpoint. Such kind of simulation can be of interest, since spoilers are considered as possible device for vibration and flutter control[22] or separation control.[23]

For this test case a spoiler on a NACA 0012 airfoil is considered, applying an upward rotation $\Delta\theta = 35$ deg. with the following smooth time law

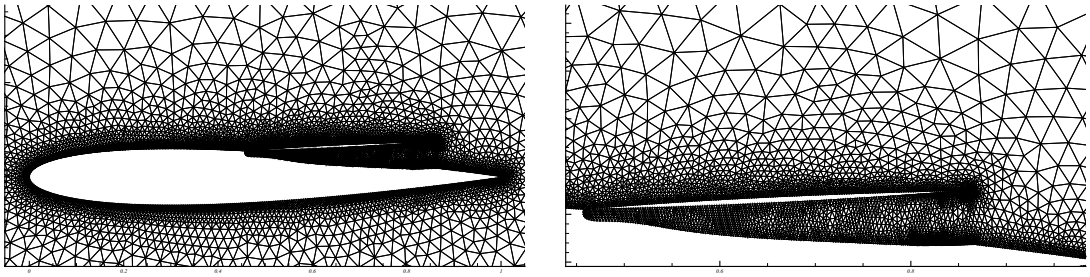
$$\theta = \frac{\Delta\theta}{2}\left[1 - \cos\left(\frac{K}{2t_{ref}}\min(t, T)\right)\right],$$

where $t_{ref} = 0.357$ s, $K = 0.1$, the final time for the opening $T = 2\pi\frac{t_{ref}}{K}$. the asymptotic Mach number is set to 0.3, and for the initial configuration the angle of attack is null. The evolution of the computational grid is shown in Figure 8 from the beginning, where the spoiler is completely closed up to the opening endpoint. The grid is updated using the adaption scheme already presented. It must be noted that the opening of the spoiler causes a significant change in the topology of the mesh. The results in terms of streamlines and pressure coefficient values in the flowfield are shown in Figure 9, where the development of a main vortex structure plus two secondary vortices behind the spoiler is shown. It must be stressed that in the present inviscid simulations the separation point is fixed by the sharp end of the spoiler.

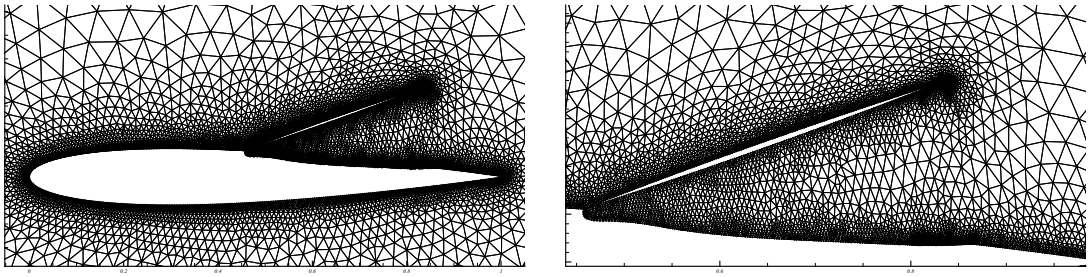
Figure 10 shows the instantaneous distribution of pressure coefficient on the airfoil and on the spoiler. The initial pressure distribution on the the lower and upper side of the airfoil is very similar: the pressure inside the narrow slit is constant, and the air there is essentially still. When the spoiler opening starts, a strong variation of pressure on the lower and upper side of the control surface appears and the surface generates a negative lift. The value of C_p along the lower side of the airfoil main body decreases slightly while increases on the upper side. While the forward part of the main airfoil is not generating any significant amount of force the rear part is lifting. At the end of the opening, the pressure on the lower and the upper side of the main airfoil are significantly different and the total lift is negative, the total force generated on the spoiler is reduced. The rear part of the main airfoil is not generating any significant force while the forward part is



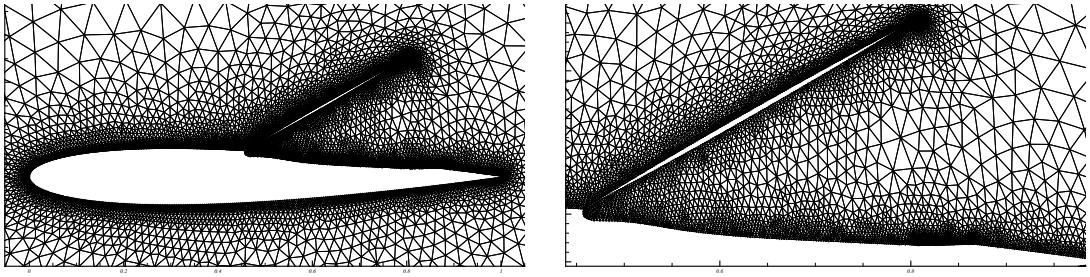
(a) $i = 0$, $t/t_{ref} = 0$



(b) $i = 100$, $t/t_{ref} = 20$

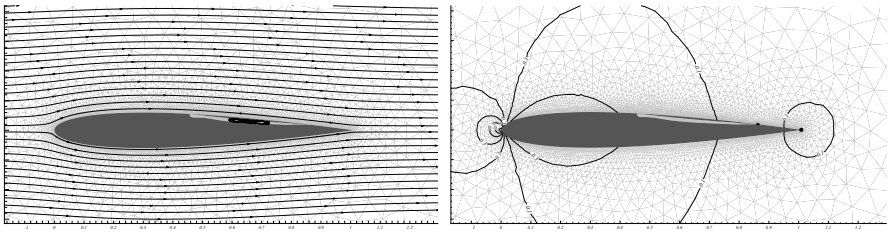


(c) $i = 200$, $t/t_{ref} = 40$

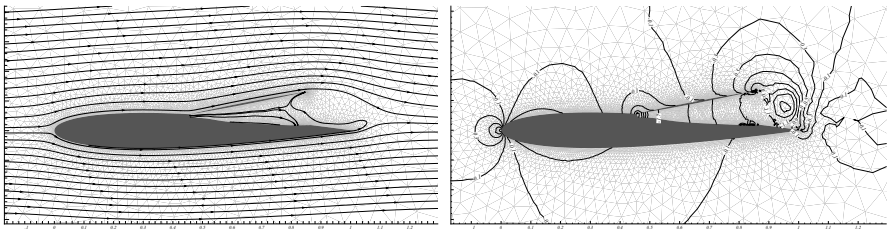


(d) $i = 300$, $t/t_{ref} = 60$

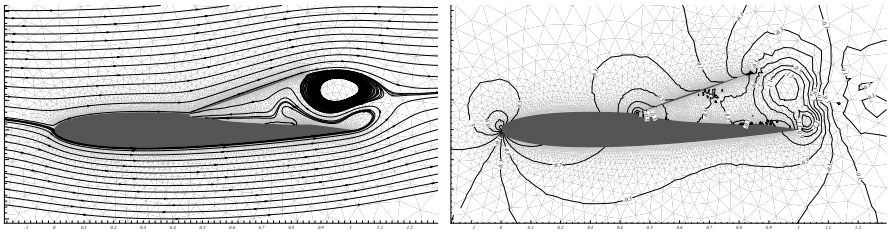
Figure 8: Mesh around the airfoil and close-up at different time steps.$t_{ref} = 0.357$ s.



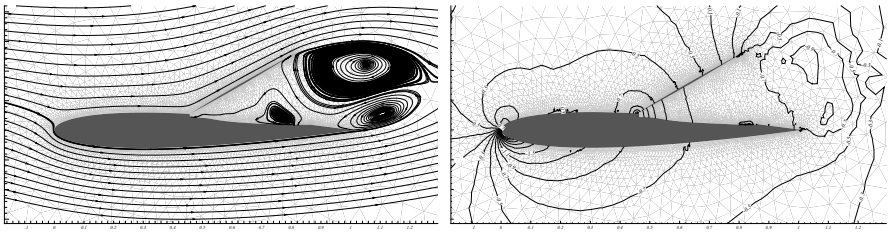
(a) $i = 0$, $t/t_{ref} = 0$



(b) $i = 150$, $t/t_{ref} = 30$



(c) $i = 200$, $t/t_{ref} = 40$



(d) $i = 400$, $t/t_{ref} = 80$

Figure 9: Streamlines and pressure coefficient past the airfoil at different time steps.

significantly down-lifting; the spoiler generates a downward force too. As the spoiler is completely opened, the low pressure peak reaches a lower value while the overall pressure in the area under the control surface increases. The time histories of the global airfoil force coefficients is shown in Figure 11. During the first 3 seconds of motion the lift and drag coefficients are essentially null. After 3 seconds, ($\theta \simeq 1.5$ deg.) a small amount of negative lift is generated, caused by the positive C_p experienced by the upper side. The lift coefficient increases up to 0.2 at $t/t_{ref} = 30$. In this configuration the spoiler is rotated in such way that a negative value of C_p is generated in the region right after the spoiler, i.e. $x/c = 0.8 \div 1$ (see Figure 10(b)) due to the presence of a vortex induced by the flow separation. On the other hand, a rather high value of pressure, $C_p \simeq -0.3$, is experienced on the bottom, allowing the flow to perform an almost full re-compression. As the spoiler is further rotated, the lift coefficient decreases until a -1.6 value is reached. During this phase, the pressure on the upper side of the airfoil is reduced slightly on the tail and increased on the nose, while the lower part of the main airfoil experiences a significant reduction in pressure. The presence of a counter-clockwise vortex close to the airfoil trailing edge affects significantly the flowfield not allowing a complete re-compression on the bottom side. The rotation is completed at $t/t_{ref} = 20\pi$ but the steady state is not reached since the transient is still not damped out.

5 CONCLUSIONS

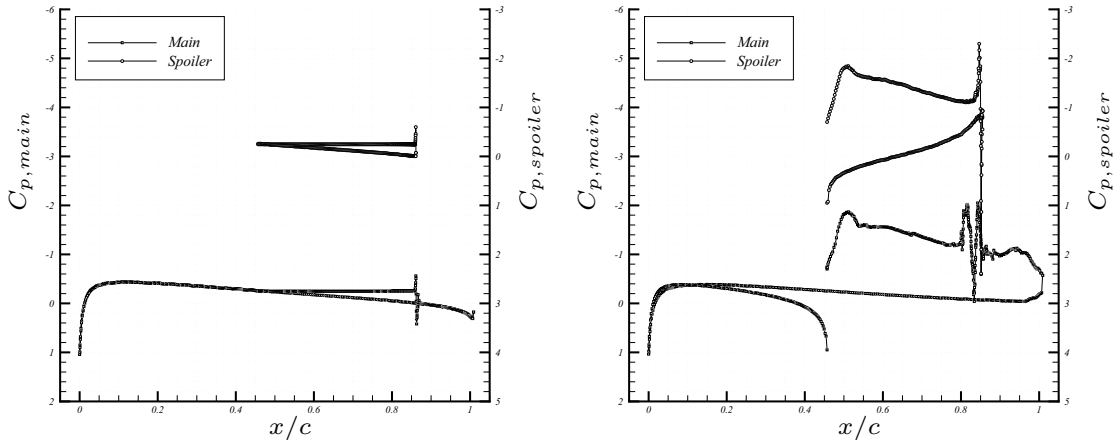
A novel flow solver for unsteady compressible flows with dynamic adaptive meshes was presented. The proposed approach is based on the interpretation of simple mesh adaptation techniques, namely, edge swapping, node insertion and node deletion, in terms of the continuous modification (in time) of suitably defined finite volumes surrounding the nodes of the triangulation. Therefore, the updated value of the unknowns at the grid nodes is automatically recovered by using an Arbitrary Lagrangian Eulerian (ALE) description of the flow equations, without resorting to an explicit interpolation step.

Numerical simulation of an oscillating airfoil translating within a fixed computational mesh were presented to assess the validity of the proposed approach. As an exemplary case, the numerical simulation of the opening process of a spoiler were carried out, to test the capabilities of both the grid alteration algorithm and the flow solver. These preliminary results increase the confidence of the authors on the possibility of applying the present approach to more complex three-dimensional flows.

REFERENCES

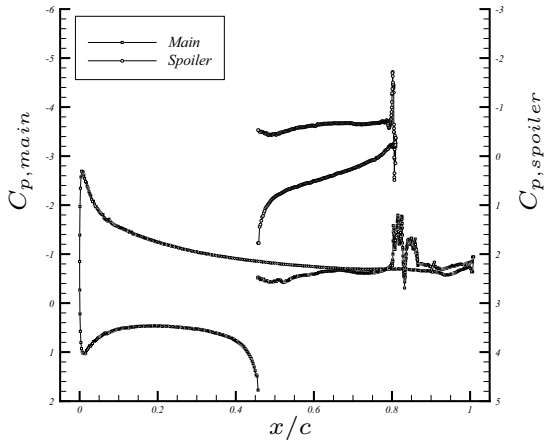
[1] L. Cavagna, S. Ricci, A. Scotti, Active aeroelastic control over a four control surface wind tunnel wing model, in: International Forum on Aeroelasticity and Structural Dynamics IFASD-2007, Stochkolm, Sweden, 2007.

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

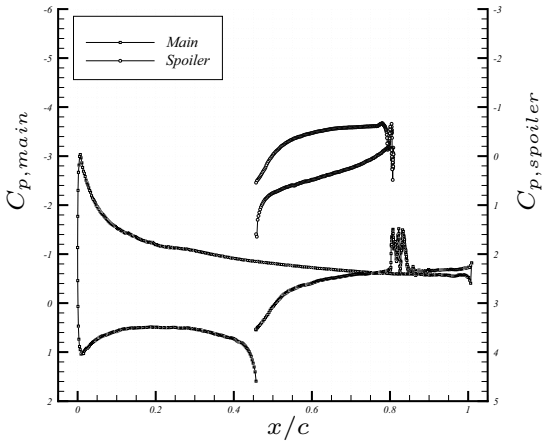


(a) $i = 0$, $t/t_{ref} = 0$

(b) $i = 150$, $t/t_{ref} = 30$



(c) $i = 300$, $t/t_{ref} = 60$



(d) $i = 400$, $t/t_{ref} = 80$

Figure 10: Pressure coefficient around the modified NACA0012 airfoil. The $C_p - x$ curve on the upper and lower side of the spoiler is shifted along the y-axis for clarity.

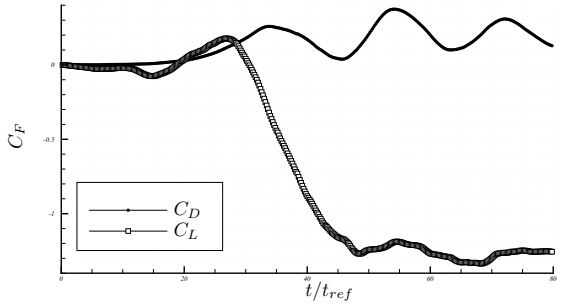


Figure 11: Time history of the forces coefficients.

[3] L. Cavagna, G. Quaranta, P. Mantegazza, Application of Navier-Stokes simulations for aeroelastic stability assessment in transonic regime, Computers & Structures **85** (11-14) (2007) 818–832.

[4] 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.

[5] R. Mertins, E. Elsholz, S. Barakat, B. Colak, 3D viscous flow analysis on wing-body-aileron-spoiler configurations, Aerospace Science and Technology **9** (6) (2005) 476–484.

[6] O. Hassan, K. A. Srensen, K. Morgan, N. P. Weatherill, A method for time accurate turbulent compressible fluid flow simulation with moving boundary components employing local remeshing, International Journal for Numerical Methods in Fluids **53** (8) (2007) 1243–1266.

[7] D. Muffo, G. Quaranta, A. Guardone, Compressilbe fluid-flow ale formulation on changing topology meshes for aeroelastic simulations, in: Proceedings of the 26$^{\text{th}}$ ICAS Congress, Anchorage, Alaska, 2008.

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

[9] L. A. Freitag, C. Ollivier-Gooch, Tetrahedral mesh improvement using swapping and smoothing, International Journal for Numerical Methods in Engineering **40** (21) (1997) 3979–4002.

[10] T. J. Baker, Mesh deformation and modification for time dependent problems, International Journal for Numerical Methods in Fluids **43** (6-7) (2003) 747–768.

[11] J. Donea, An arbitrary Lagrangian-Eulerian finite element method for transient fluid–structure interactions, Comp. Meth. Appl. Mech. Engng. **33** (1982) 689–723.

[12] 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.

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

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

[15] A. Harten, J. M. Hyman, Self adjusting grid methods for one-dimensional hyperbolic conservation laws, J. Comput. Phys. **50** (1983) 253–269.

[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. Monotoniticy and conservation combined in a second order scheme, J. Comput. Phys. **14** (1974) 361–370.

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

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

[21] G. Fillola, M. Le Pape, M. Montagnac, Numerical simulations around wing control surfaces, in: Proceedings of the 24th ICAS International Congress of the aeronautical sciences, Yokohama, Japan, 2004.

[22] J. Rivera Jr, B. Dansberry, R. Bennett, M. Durham, W. Silva, NACA 0012 benchmark model experimental flutter results with unsteady pressure distributions, in: Proceedings of the 33rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference. Dallas, TX, No. AIAA Paper 92-2396, 1992.

[23] M. Schatz, T. Knacke, F. Thiele, R. Meyer, W. Hage, D. Bechert, Separation control by self-activated movable flaps, in: 42nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 2004, pp. 5–8.