Commun. Comput. Phys.
doi: 10.4208/cicp.181010.180511s

Vol. **12**, No. 3, pp. 866-884
September 2012

Numerical Simulation of Compressible Vortical Flows Using a Conservative Unstructured-Grid Adaptive Scheme†

Giuseppe Forestieri, Alberto Guardone*, Dario Isola, Filippo Marulli and Giuseppe Quaranta

Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano, Via La Masa, 34, 20156 Milano, Italy.

Received 18 October 2010; Accepted (in revised version) 18 May 2011

Available online 1 March 2012

Abstract. A two-dimensional numerical scheme for the compressible Euler equations is presented and applied here to the simulation of exemplary compressible vortical flows. The proposed approach allows to perform computations on unstructured moving grids with adaptation, which is required to capture complex features of the flow-field. Grid adaptation is driven by suitable error indicators based on the Mach number and by element-quality constraints as well. At the new time level, the computational grid is obtained by a suitable combination of grid smoothing, edge-swapping, grid refinement and de-refinement. The grid modifications—including topology modification due to edge-swapping or the insertion/deletion of a new grid node—are interpreted at the flow solver level as continuous (in time) deformations of suitably-defined node-centered finite volumes. The solution over the new grid 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. Numerical simulations of the steady oblique-shock problem, of the steady transonic flow and of the start-up unsteady flow around the NACA 0012 airfoil are presented to assess the scheme capabilities to describe these flows accurately.

AMS subject classifications: 52B10, 65D18

Key words: Moving grid method, conservative interpolation.

1 Introduction

The accurate prediction of the trailing vortexes from airplane wings and helicopter blades is of paramount importance for the determination of the aerodynamic characteristics of the aircraft [18]. For example, in air traffic control, the persistence over the airport of

*Corresponding author. *Email address:* alberto.guardone@polimi.it (A. Guardone)

†Presented at the 2nd European Seminar on Coupled Problems June 28–July 2, 2010, Pilsen, Czech Republic.

start-up vortexes originating from lift strongly limits the availability of airstrips. In helicopters aerodynamics, the accurate evaluation of the unsteady dynamics of blade tip vortexes is relevant to the prediction of the aerodynamic loads of the blade and of the so called blade vortex interaction phenomenon, whose occurrence prevents the widespread use of helicopters in urban environment [12]. In nature and in man-made machinery, countless examples of fluid flows can be found for which the accurate evaluation of vortex dynamics is fundamental, including the lift due to wing flapping in insect and bird flight, thrust generated by tail flapping in fishes, unsteady wind loads caused by alternate vortex separation on slender structures such as bridges or towers or the mutual influence of wind turbine in wind farms.

From a numerical point of view, this kind of flows present peculiarities that make them difficult or impossible to simulate accurately. Indeed, different geometrical scales coexist in the flow field which strongly influence each other. For example, the wake dynamics past the separation point in separated or recirculating flows determines in a coupled manner the position of the separation line itself. Moreover, the discrete representation of slip lines, which requires high spatial accuracy, cannot be easily accomplished by the use of high-order spatial discretization or so-called p-refinement. In compressible flows, further difficulties are encountered due to possible occurrence of nonlinear wave-fields including shocks. Indeed, stabilization techniques used to capture shock wave fronts without spurious oscillations, such as for example Total Variation Diminishing schemes [19,21], usually produce inaccurate results if applied to linearly-degenerate waves such as contact discontinuities or slip lines, thus making it necessary to locally adapt the computational grid close to discontinuities to reduce the amount of numerical viscosity.

A two-dimensional adaptive-grid numerical scheme for the compressible Euler equations is applied here to the evaluation of the start-up vortex and vortical wake from two-dimensional airfoils [8,9]. Grid adaptation is driven by suitable error indicators based on the Mach number and by element-quality constraints as well. The error indicator is computed by means of a node-pair finite element approach [17]. At the new time level, the computational grid is obtained by a suitable combination of grid smoothing, edge-swapping, grid refinement and coarsening. These modification to the grid are interpreted at the flow solver level as continuous (in time) deformations of suitably-defined node-centered finite volumes. Therefore, the solution over the new grid is obtained without explicitly resorting to interpolation techniques, since the definition of suitable interface velocities allows to determine the new solution by simple integration of the Arbitrary Lagrangian-Eulerian formulation of the flow equations.

The present paper is structured as follows. First, the grid alteration strategy is presented and the different error indicators are discussed. The edge-based ALE solver is briefly recalled and the solution technique for adaptive moving grid is sketched. Numerical results for the oblique-shock problem and for the NACA 0012 airfoil in steady and unsteady flows are then reported and discussed. The paper ends with some final remarks and considerations.

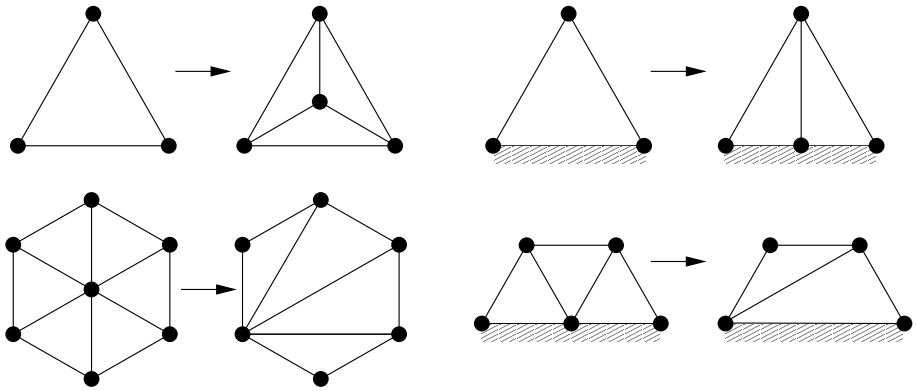


Figure 1: Top: refinement pattern by node insertion in the center of mass of an existing element for a domain (top-left) element and boundary (top-right) element. Bottom: Derefinement pattern by node deletion for a domain (bottom-left) element and boundary (bottom-right) element.

2 Grid alteration strategy

The technique used to adapt the grid to the local feature of the flow field is described in the present section. The goal of the grid alteration procedure is to locally modify the grid spacing so that the numerical error is evenly distributed within the computational domain, according to the principle of error equidistribution. As a consequence, nodes are inserted in the regions where the error is greater than the average error over the whole grid, or deleted where the error is smaller.

To evaluate the local numerical error a discrete error estimator E is required. In most applications, error estimators are either functions of flow gradients or undivided differences [1,6,11,14], or functions of the Hessian matrix $\mathcal{H}$ [2,6,7,13,25,26] of a convenient sensor variable s—such as for example the Mach number considered here—which is representative of the flow features and whose choice depends on the physical problem. In the present study, to cope with the presence of shock waves and smooth-flow regions, the following nodal estimator is used [25]

$$E_i(s)=h_i^2\sqrt{\varepsilon_i^2(s;\boldsymbol{\tau})+\varepsilon_i^2(s;\boldsymbol{\eta})},\qquad \text{with}\quad \varepsilon(s;\boldsymbol{\omega})=\frac{\boldsymbol{\omega}^{\mathrm{T}}\mathcal{H}(s)\boldsymbol{\omega}}{h\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\nabla}s+\epsilon\mu(s)}, \tag{2.1}$$

where $E_i(s)$ is the error estimation of the i-th cell, s is the adaptation sensor variable (local Mach number in this work), h_i is the radius of the circle circumscribing the i-th cell, $\boldsymbol{\omega}$ is a generic versor in $\mathbb{R}^2$, $\boldsymbol{\tau}$ and $\boldsymbol{\eta}$ are the tangent and normal versors to the local velocity vector, ϵ is a constant chosen between 0 and 1 (0.12 here) and $\mu(s)$ is the average value of s over the computational domain. The discrete Hessian matrix $\mathcal{H}(s)$ and the gradient vector $\boldsymbol{\nabla}s$ are computed using a linear finite-element approximation of Lagrangian type within the node-pair representation [17,23].

A grid element is marked for refinement if the error estimator is larger than $E_{\mathrm{R}}=\mu(E(s))+k_{\mathrm{R}}\sigma(E(s))$, with k_{R} a constant (0.1 here) and $\sigma(E(s))$ standard deviation of the

error estimator distribution. Conversely, the grid-coarsening threshold is set at $E_D = k_D \mu(E(s))$, with k_D a constant (0.4 here), to force grid adaptation towards a greater uniformity in error distribution. Elements marked for refinement/derefinement are modified as in Fig. 1. Grid modifications amount to insert a new node in the center of mass of all elements to be refined and to remove all nodes surrounded by elements to be derefined. To improve the grid quality, Laplacian smoothing based on an elastic analogy is performed after the refinement step. After the coarsening phase, edge-swapping and grid smoothing are performed [20]. To limit the number of nodes/elements close to flow discontinuities, a minimum value for the area of the triangles is used.

In unsteady flow fields, at each time level t_n, after computing the solution on the grid from the previous time step t_{n-1}, the grid adaptation procedure is carried out. Then, the solution is recalculated at the same time step t_n using the new adapted grid. This procedure is repeated until the error estimator is uniformly distributed at t_n. Such an iterative procedure clearly implies a large computational effort and it will be called in the following Ad_1.

A strategy to reduce the computational effort, with particular reference to unsteady problems, is also implemented here. The idea is to compute the solution at t_n using the grid at t_{n-1}, to perform an iterative adaptation procedure, and to recompute the solution on the final grid only at the end of the grid adaptation routine, without intermediate computations. To this purpose, an empirical error redistribution technique is used, in which each new/modified element inherits the area-weighted value of the error estimator of its parental element for grid refinement or neighboring elements for grid derefinement. A complete convergence of the iterative procedure at time t_n is achieved when the number of refined and derefined element is below a pre-defined threshold. After the final solution at t_n is computed on the adapted grid, the computation advances to the next time level t_{n+1}. Hereinafter, this latter strategy will be called Ad_2.

3 Edge-based ALE solver for adaptive moving grids

The Euler equations in an Arbitrary Lagrangian Eulerian (ALE) framework [3,4] for compressible two-dimensional flows read

$$\frac{d}{dt}\int_{\mathcal{C}(t)} \mathsf{u} + \oint_{\partial\mathcal{C}(t)} \left[\mathbf{f}(\mathsf{u}) - \mathsf{u}\boldsymbol{v}\right]\cdot\boldsymbol{n} = 0, \qquad \forall \mathcal{C}(t) \subseteq \Omega(t), \tag{3.1}$$

where $\mathsf{u} = (\rho, \boldsymbol{m}, E^{\mathrm{t}})^{\mathrm{T}} \in \mathbb{R}^+ \times \mathbb{R}^3$ is the vector of the conservative variables density, momentum vector and total energy per unit volume. The solution is sought for in the spatial domain $\Omega(t) \in \mathbb{R}^2$ of boundary $\partial\Omega(t)$ and $\forall t \in \mathbb{R}^+$. System (3.1) is made complete by specifying suitable initial and boundary conditions [5]. The flux function $\mathbf{f} = (\mathsf{f}_x, \mathsf{f}_y)^{\mathrm{T}} \in \mathbb{R}^4 \times \mathbb{R}^2$ is defined as $\mathbf{f}(\mathsf{u}) = (\boldsymbol{m},\ \boldsymbol{m}\otimes\boldsymbol{m}/\rho + P(\mathsf{u})\ \mathsf{I}^2,\ [E^{\mathrm{t}} + P(\mathsf{u})]\boldsymbol{m}/\rho)^{\mathrm{T}}$ where I^2 is the 2×2 identity matrix. The normal vector $\boldsymbol{n} = \boldsymbol{n}(\boldsymbol{s}, t) = (n_x, n_y)^{\mathrm{T}}$ is a function of the curvilinear coordinate $\boldsymbol{s}$ along $\partial\mathcal{C}$ and of the time as well. In (3.1), the term $\mathsf{u}\boldsymbol{v} = (\rho\boldsymbol{v}, \boldsymbol{m}\otimes\boldsymbol{v}, E^{\mathrm{t}}\boldsymbol{v})^{\mathrm{T}}$, where

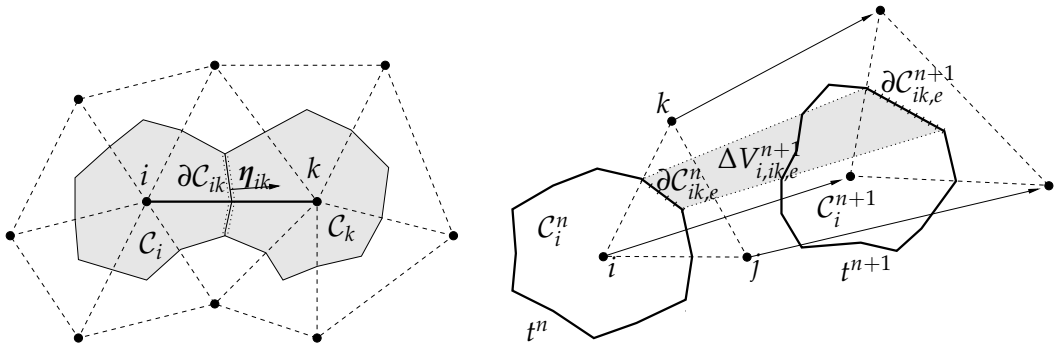


Figure 2: Left: edge associated with the finite volume interface $\partial\mathcal{C}_{ik}=\partial\mathcal{C}_i\cap\partial\mathcal{C}_k$ and metric vector $\boldsymbol{\eta}_{ik}$ in two spatial dimensions. The two shaded regions are the finite volumes $\mathcal{C}_i$ and $\mathcal{C}_k$; dashed lines indicate the underlying triangulation. Right: area swept by portion of the interface $\partial\mathcal{C}_{ik,e}$ pertaining to element e, made of nodes i,j and k, during the time interval $[t^n,t^{n+1}]$.

$\boldsymbol{v}=\boldsymbol{v}(\boldsymbol{s},t)$ is the local velocity of the interface $\partial\mathcal{C}(t)$, accounts for the flux contribution due to the movement of the control volume $\mathcal{C}(t)$. The finite volume discrete counterpart of the Euler equation (3.1) is obtained by selecting a finite number of non overlapping volumes $\mathcal{C}_i(t)\subset\Omega(t)$, such that $\bigcup_i\mathcal{C}_i(t)\equiv\Omega(t)$. In 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 equation (3.1) reads

$$\frac{d[V_i\mathsf{u}_i]}{dt}=-\oint_{\partial\mathcal{C}_i}\left[\mathbf{f}(\mathsf{u})-\mathsf{u}\boldsymbol{v}\right]\cdot\boldsymbol{n},\quad\forall i\in\mathcal{K},\tag{3.2}$$

where V_i is the volume of the i-th cell and $\mathcal{K}$ is the set of all nodes of the triangulation. 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 (3.2) is split into domain and boundary contributions as follows

$$\oint_{\partial\mathcal{C}_i}\left[\mathbf{f}(\mathsf{u})-\mathsf{u}\boldsymbol{v}\right]\cdot\boldsymbol{n}=\sum_{k\in\mathcal{K}_{i,\neq}}\int_{\partial\mathcal{C}_{ik}}\left[\mathbf{f}(\mathsf{u})-\mathsf{u}\boldsymbol{v}\right]\cdot\boldsymbol{n}+\int_{\partial\mathcal{C}_i\cap\partial\Omega}\left[\mathbf{f}(\mathsf{u})-\mathsf{u}\boldsymbol{v}\right]\cdot\boldsymbol{n},\tag{3.3}$$

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, (i.e. the node to node connectivity) and $\partial\mathcal{C}_{ik}=\partial\mathcal{C}_i\cap\partial\mathcal{C}_k$ is the cell interface between the volumes $\mathcal{C}_i$ and $\mathcal{C}_k$ (see Fig. 2). 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 $\Phi\in\mathbb{R}^4$, representing the exchange across the cell interface $\partial\mathcal{C}_i\cap\partial\mathcal{C}_k$, is introduced [10]. Considering a centered approximation of the unknown and of the flux function at the cell interfaces, the domain contributions read

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

https://doi.org/10.4208/cicp.181010.180511s

The integrated normal vector $\boldsymbol{\eta}_{ik}$ and the integrated normal interface velocity ν_{ik} are defined as

$$\boldsymbol{\eta}_{ik}(t)=\int_{\partial\mathcal{C}_{ik}}\boldsymbol{n} \qquad \text{and} \qquad \nu_{ik}(t)=\int_{\partial\mathcal{C}_{ik}}\boldsymbol{v}\cdot\boldsymbol{n}=\frac{dV_{i,ik}}{dt}, \tag{3.4}$$

where $V_{i,ik}$ is the volume portion pertaining to cell i swept by the interface i-k, see Fig. 2(b). In (3.4), $\boldsymbol{\eta}_{ik}$ is expressed in terms of its magnitude η_{ik} times the unit vector $\hat{\boldsymbol{\eta}}_{ik}=\boldsymbol{\eta}_{ik}/\eta_{ik}$. By assuming a constant interface flux also for boundary contributions, since $\mathsf{u}=\mathsf{u}_i$ over the boundary portion $\partial\mathcal{C}_i\cap\Omega$, the boundary integrals in (3.3) simplifies to

$$\int_{\partial\mathcal{C}_i\cap\partial\Omega}\left[\mathbf{f}(\mathsf{u})-\mathsf{u}\boldsymbol{v}\right]\cdot\boldsymbol{n}\simeq\mathbf{f}(\mathsf{u}_i)\cdot\boldsymbol{\xi}_i-\mathsf{u}_i\nu_i=-\Phi^{\partial}(\mathsf{u}_i,\nu_i,\hat{\boldsymbol{\xi}}_i,\xi_i), \tag{3.5}$$

where

$$\boldsymbol{\xi}_i(t)=\int_{\partial\mathcal{C}_i\cap\partial\Omega}\boldsymbol{n} \qquad \text{and} \qquad \nu_i(t)=\int_{\partial\mathcal{C}_i\cap\partial\Omega}\boldsymbol{v}\cdot\boldsymbol{n}=\frac{dV_{i,\partial}}{dt}, \tag{3.6}$$

with $V_{i,\partial}$ volume swept by the portion of $\mathcal{C}_i$ that belongs to the boundary $\partial\Omega$ of the computational domain. The vector $\boldsymbol{\xi}_i$ is rewritten in terms of its magnitude ξ_i and unit vector $\hat{\boldsymbol{\xi}}_i=\boldsymbol{\xi}_i/\xi_i$.

In the actual computations, the centered approximation of the flux function at the cell interfaces is replaced by a Total Variation Diminishing (TVD) numerical flux [10, 21]. To this purpose, the second order centered approximation is replaced by the first order Roe flux near flow discontinuities [16]. The switch is controlled by the limiter proposed by van Leer [21]. 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. As done in [24], the extension nodes belong to the two edges best aligned with i-k.

By complementing the discrete form of the Euler equation for a general, namely, not centered approximation of the numerical fluxes, with the definitions (3.4) and (3.6), one finally obtains the following system of $N_{\text{dof}}+N_{ik}+N_{i,\partial}$ ODEs

$$\begin{cases}\dfrac{d}{dt}[V_i\mathsf{u}_i]=\displaystyle\sum_{k\in\mathcal{K}_{i,\neq}}\Phi(\mathsf{u}_i,\mathsf{u}_k,\nu_{ik},\hat{\boldsymbol{\eta}}_{ik},\eta_{ik})+\Phi^{\partial}(\mathsf{u}_i,\nu_i,\hat{\boldsymbol{\xi}}_i,\xi_i), & \\ & \forall i\in\mathcal{K},\\ \dfrac{dV_{i,ik}}{dt}=\nu_{ik}, & \forall k\in\mathcal{K}_{i,\neq},\\ \dfrac{dV_{i,\partial}}{dt}=\nu_i, & \end{cases} \tag{3.7}$$

of $N_{\text{dof}}=N_{\mathcal{K}}\times 4$ conservation laws equations, with $N_{\mathcal{K}}=\dim(\mathcal{K})$ the total number of grid points, and $N_{ik}+N_{i,\partial}$ consistency conditions, with N_{ik} and $N_{i,\partial}$ number of grid edges and of boundary nodes, respectively. The ODE system above is solved using a Backward Differences Formulæ (BDF) scheme of order either one or two, as reported in the numerical results section. The nonlinear system (3.7) for the fluid variables u at time level $n+1$ is

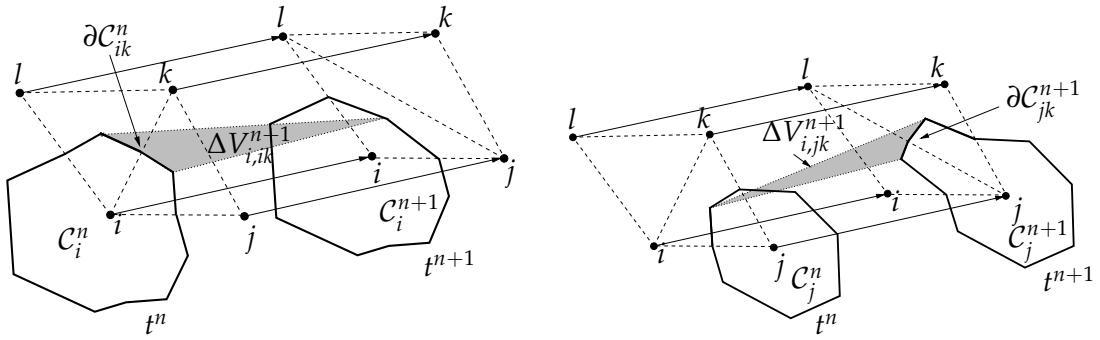


Figure 3: Interpretation of the edge swapping as continuous finite volume deformation. Left: evaluation of the normal interface velocity (area of the shaded region) for edge i-k that is deleted due to edge-swapping from edge i-k at time t^n into edge j-k at time t^{n+1}. Right: evaluation of the normal interface velocity for edge j-k that is created due to edge-swapping.

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 [22], to improve the conditioning number of the Jacobian matrix.

As discussed in [8, 9], all grid adaptation techniques can be interpreted as the effect of a continuous modification in time of suitably defined finite volume. An example is given in Fig. 3, in which the geometrical interpretation of edge swapping in the ALE framework is sketched. In Fig. 3, at time t^{n+1}, $\eta_{ik}=0$, so the Eulerian numerical flux is zero, but $v_{ik}=\Delta V^{n+1}_{i,ik}\neq 0$ and the ALE contribution is non-zero. All grid modification due to node insertion, node deletion or element swapping are automatically taken into account by the ALE framework; the reader is referred to [8, 9] for a detailed description of the ALE interpretation of grid adaptation.

4 Numerical results

Numerical experiments are carried out to test the accuracy of the proposed solution procedure. The computational efficiency of the two adaptation strategies named Ad_1 and Ad_2 discussed in Section 2 is first evaluated against the standard oblique-shock test case in Section 4.1. In Section 4.2, steady flow simulations around the NACA 0012 airfoil are performed to assess the solver capabilities at capturing relevant flow features of transonic flows, including shock waves, shear lines and regions of smooth flows. Start-up vortex formation and evolution around the NACA 0012 is investigated and reported in Section 4.3.

4.1 Oblique shock problem

The standard oblique-shock test problem is performed to compare the efficiency of the two proposed adaptation strategies. In the problem under scrutiny, the steady state is a

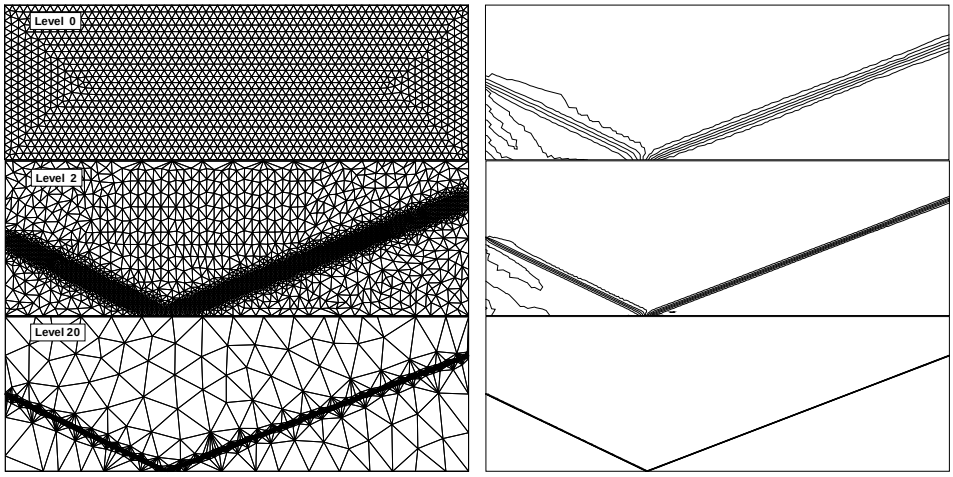


Figure 4: Grid and Mach number in the flow field for the oblique-shock problem for Ad_1. From top to bottom: original grid (1 532 nodes, 2 900 elements), intermediate grid at the 2-th adaptation level (2 735 nodes, 5 340 elements), final grid at the 20-th adaptation level (10 557 nodes, 20 994 elements).

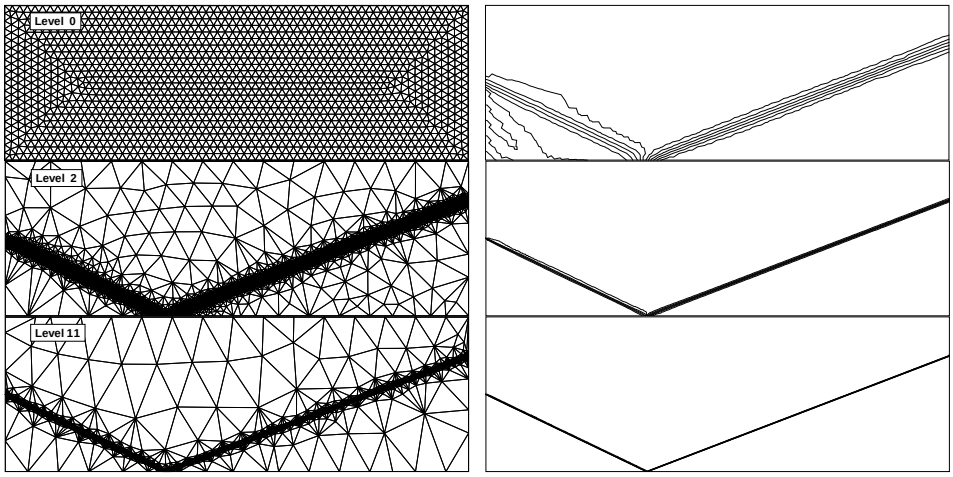
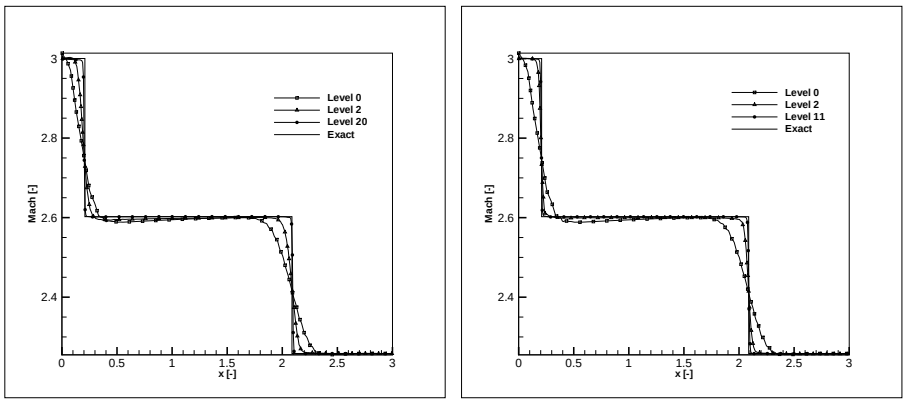


Figure 5: Grid and Mach number in the flow field for the oblique-shock problem for Ad_2. From top to bottom: original grid (1 532 nodes, 2 900 elements), intermediate grid at the 2-th adaptation level (3 787 nodes, 7 460 elements), final grid at the 11-th adaptation level (11 500 nodes, 22 916 elements).

parallel supersonic flow with an oblique shock ($-25.60°$) that is reflected at $y=0.5$ along the lower solid boundary.

Starting from a grid made of 1 532 nodes and 2 900 elements, convergence on both grid and solution is achieved using the two proposed strategies (Ad_1 and Ad_2). A minimum element area of 10^{-6} is imposed; the variable used to compute the error estimator is the local Mach number. For both strategies, the relative difference of the area-averaged mean error μ_w of two successive adaptation cycles is below 2%, namely $\left(\mu_w^{i+1}-\mu_w^i\right)/\mu_w^1<0.02$.



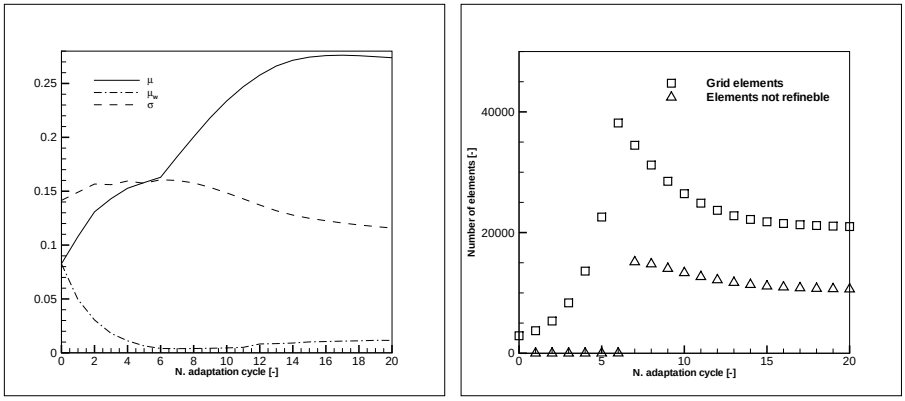
(a) Ad_1 (b) Ad_2

Figure 6: Comparison between analytical and numerical solution at $y=0.4$ for the oblique-shock problem for different adaptation steps and strategies.

The computational times are 170m 01s using Ad_1 and 53m 28s using Ad_2 on a single core of an Intel Xeon QuadCore 3.166GHz. The number of adaptation cycles required to reach convergence is 20 and 11 for Ad_1 and Ad_2, respectively. Therefore, the cost of each adaptation step for the Ad_1 technique is twice as that of the Ad_2, possibly because a better initial guess is computed for the latter thus reducing the time required to reach steady state. Figs. 4 and 5 show the evolution of the grid and of the solution in terms of Mach number at different adaptation levels (initial, intermediate and final) for Ad_1 and Ad_2, respectively. Figs. 6(a) and 6(b) show the numerical solutions at $y=0.4$ against the exact one. The Ad_2 technique shows better results after two second adaptation steps already, moreover it requires roughly half of the computational time. Figs. 7 and 8 show the error and number of points at each time steps for both Ad_1 and Ad_2, respectively. In both cases the algebraical error μ increases due to the redistribution of elements from low error regions (constant solution regions) to high error regions (shock waves). On the contrary the mean error scaled on areas μ_w decreases till the threshold of minimum area is reached and then remains quite constant.

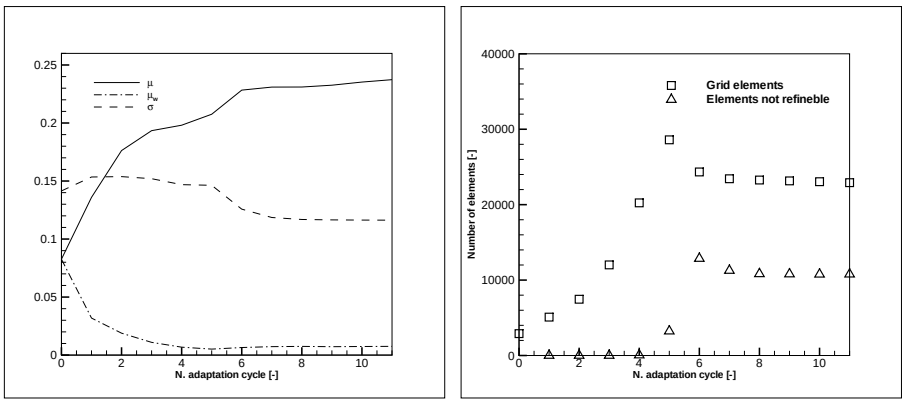
4.2 Steady transonic flow around the NACA0012 airfoil

The case of the compressible inviscid flow developing past a NACA 0012 airfoil at 5° incidence and Mach = 0.7 is discussed. This transonic flow is characterized by the presence of a shock wave located at about 45% of the chord and by a slip (entropy) line detaching from the trailing edge. Fig. 9 shows the Mach number obtained after 15 cycles of Ad_1 steps, i.e. refinement, smoothing, derefinement, edge-swapping and smoothing cycle. The variable adopted to evaluate the error is the local Mach number. After each adaptation step, the steady state is computed to update the error estimator. The proposed adaptation procedure is capable of capturing both the shock wave and the shear line.



(a) (b)

Figure 7: Adaptation technique Ad_1. Left: algebraic mean $\mu(E(s))$, area-averaged mean $\mu_w(E(s))$ and the standard deviation $\sigma(E(s))$ as a function of the adaptation step for the steady-state oblique-shock problem. Right: number of grid elements and of not refineable elements as a function of the adaptation step.



(a) (b)

Figure 8: Adaptation technique Ad_2. Left: algebraic mean $\mu(E(s))$, area-averaged mean $\mu_w(E(s))$ and standard deviation $\sigma(E(s))$ as a function of the adaptation step for the steady-state oblique-shock problem. Right: number of grid elements and of not refineable elements as a function of the adaptation step.

Fig. 10 shows the pressure coefficient over the airfoil after 0, 2 and 15 adaptation steps. Fig. 11(a) shows the algebraic mean $\mu(E(s)) = \sum_{i\in\mathcal{K}} E_i(s)/N_{\mathcal{K}}$, the area-averaged mean $\mu_w(E(s)) = \sum_{i\in\mathcal{K}}(V_i E_i(s))/\sum_{i\in\mathcal{K}} V_i$ and the standard deviation $\sigma(E(s))$ as a function of the adaptation step. Fig. 11(b) shows the number of grid elements at each adaptation step and the number of elements that cannot be refined, namely, whose areas are equal or below the limiting value of 10^{-7}. During the first iterations, both μ and σ decrease, due to the insertion of additional nodes close to the airfoil. When the fifth adaptation cycle is reached these quantities increase because new nodes are inserted close to flow discontinuities (i.e. large error regions) and simultaneously elements are removed from

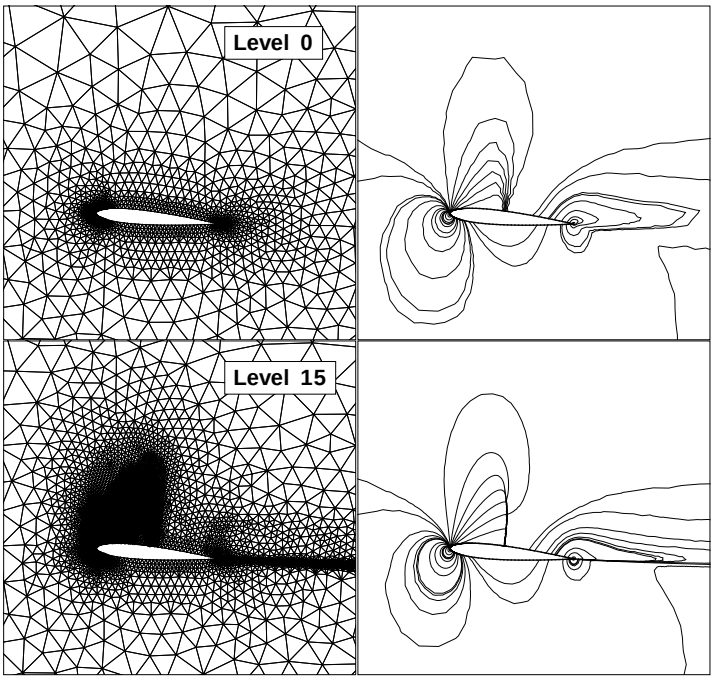


Figure 9: Grid and Mach contour for NACA 0012 at Mach = 0.7. First row: original grid (3 179 nodes, 6 030 elements) and isolines (25 levels from 0.08 to 1.5). Second row: 15-th adaptation level (26 499 nodes, 52 568 elements) and isolines (25 levels from 0.08 to 1.5).

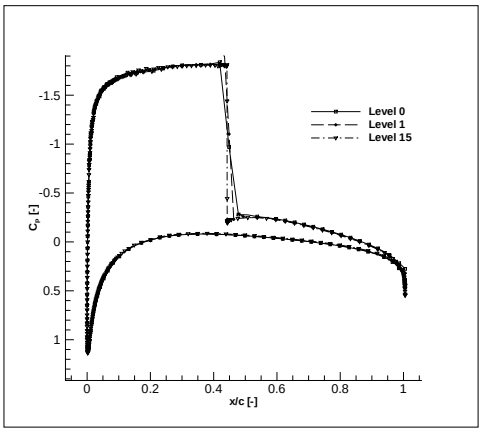
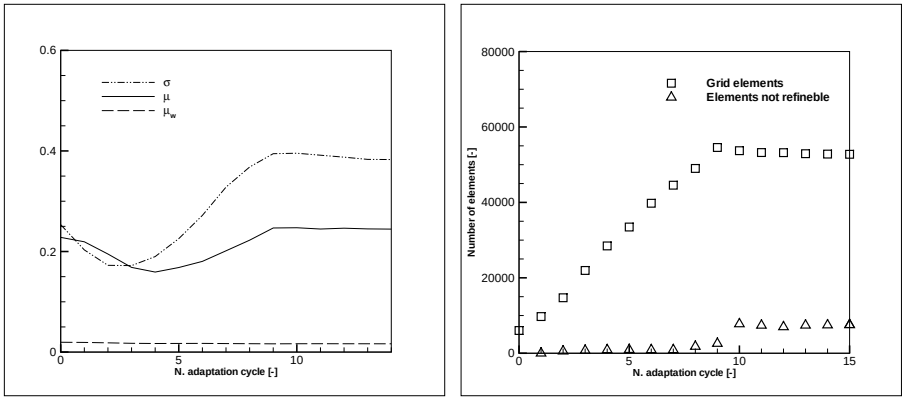


Figure 10: NACA 0012 at Mach 0.7. Pressure coefficient over the airfoil for the initial grid and for adapted grids (first and last adaptation steps).

low error regions. As new grids are produced, the number of elements that cannot be refined increases. From the tenth to last adaptation step, no new nodes are added and only mesh coarsening occurs.



(a) (b)

Figure 11: Left: algebraic mean $\mu(E(s))$, area-averaged mean $\mu_w(E(s))$ and standard deviation $\sigma(E(s))$ as a function of the adaptation step for the steady-state NACA 0012 simulations. Right: number of grid elements and of not refineable elements as a function of the adaptation step.

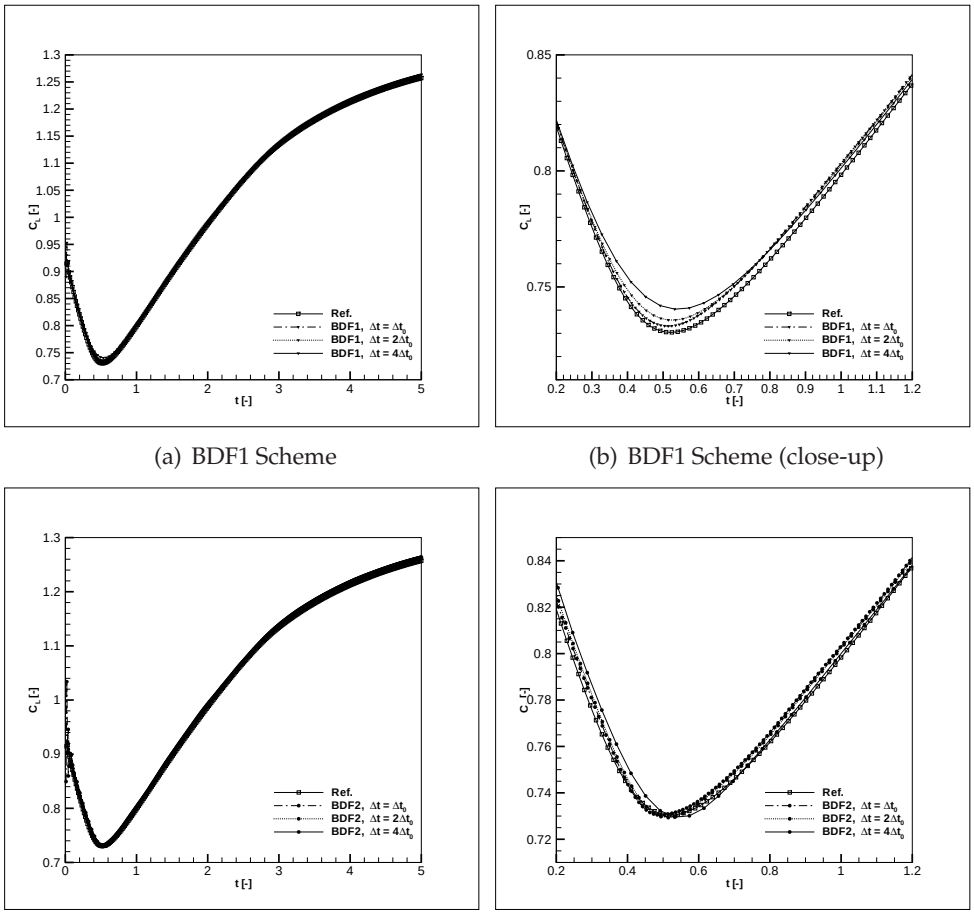
4.3 Start-up vortex from the NACA 0012 airfoil

The solution technique is now applied to the computation of the two-dimensional unsteady flow past the NACA 0012 airfoil impulsively set into motion. Similarly to the previous case, the presence of a sharp trailing edge guarantees that the Kutta condition is fulfilled and the Euler equations can be used to correctly represent the space-time evolution of the start-up vortex in the computational domain.

As a consequence of lift generation over the airfoil, a trailing-edge vortex is produced. This vortex is named start-up vortex and its dynamics strongly influence the time history of the force coefficient over the airfoil. Since a clockwise circulation $\Gamma_a(t)$ around the airfoil is generated, the starting vortex is associated to a counterclockwise circulation $\Gamma_v(t)$. As the distance between the airfoil and the start-up vortex increases due to their relative motion, the influence of the latter on the aerodynamic coefficients vanishes. Eventually, the steady-state value of the lift coefficient $C_L(t)$ is attained and $\Gamma_v(\infty)=-\Gamma_a(\infty)$.

The influence of the start-up vortex dynamics on the airfoil is very difficult to investigate numerically. Since both the flow field around the airfoil and the start-up vortex need to be captured accurately at the same time, since the circulation around the airfoil and the intensity of the start-up vortex balances instantaneously, due to the Kelvin's circulation theorem. It is therefore mandatory to adapt the computational grid in an unsteady fashion to follow the vortex dynamics and, at the same time, accurately compute the flow field close to the airfoil.

In the numerical tests, the airfoil reference is used. The free stream velocity is therefore a step function that starts at zero and reaches its asymptotic value V_∞ at the time $t>0$. In the computations, the time variable t is made dimensionless by the free-stream velocity V_∞ divided by the airfoil chord c. The considered dimensionless time interval is $t\in[0,5]$. Tests were performed for a free-stream Mach number $M_\infty=0.7$ and for $\alpha=10^\circ$ incidence.

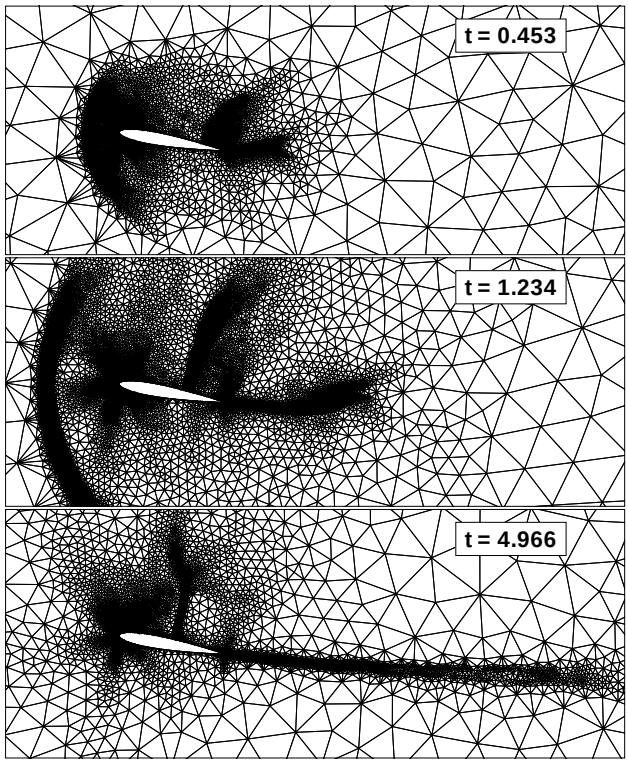


(a) BDF1 Scheme (b) BDF1 Scheme (close-up)

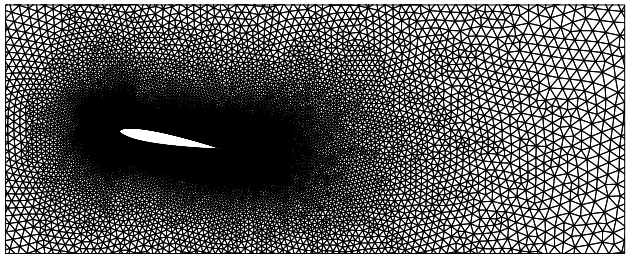
(c) BDF2 Scheme (d) BDF2 Scheme (close-up)

Figure 12: Impulsively-started NACA 0012. Mach $=0.7$, $\alpha=10^\circ$: lift coefficient time history for the reference grid against solution over the initial coarse grid for different values of the time step using the one-step BDF1 (top) and the two-step BDF2 (bottom) schemes (c). In (b) and (d) an enlarged view of the time interval across the minimum C_L is shown.

Simulations on both adapted and fixed grid were carried out. The fixed grid used to compute the reference solution is made of 29 367 nodes and 58 094 elements. The adaptation process starts from an initial grid made of 4 989 nodes and 9 627 elements. The adapted-grid computations were carried out for two different thresholds for the minimum area of the elements, namely, 3×10^{-5} and 3×10^{-4}. The variable adopted to evaluate the error is the local Mach number. Time integration over the reference grid were performed using the BDF1 scheme with $\Delta t=0.0164$. To assess time convergence, both the one-step BDF1 and two-step BDF2 time integration schemes were used over the initial grid for $\Delta t=\Delta t_0=0.0103$, $\Delta t=2\Delta t_0$, $\Delta t=4\Delta t_0$. Results for the lift coefficient C_L are reported in Fig. 12 and demonstrate time convergence to the reference solution for the smallest



(a) Adapted grid

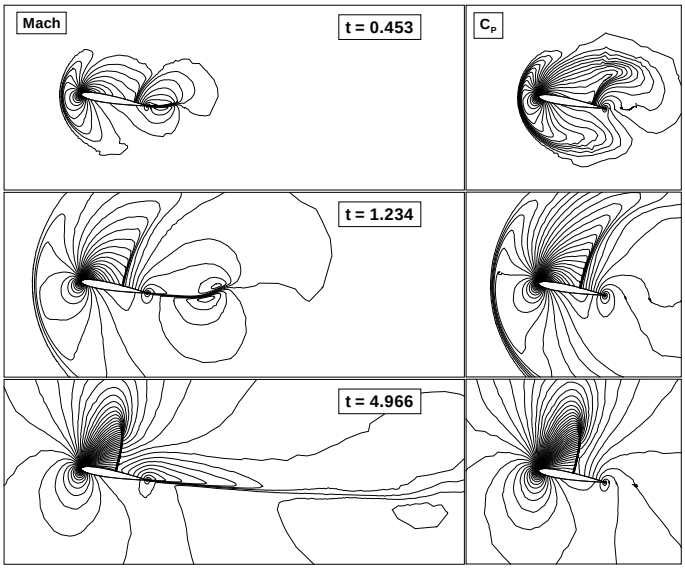


(b) Reference grid

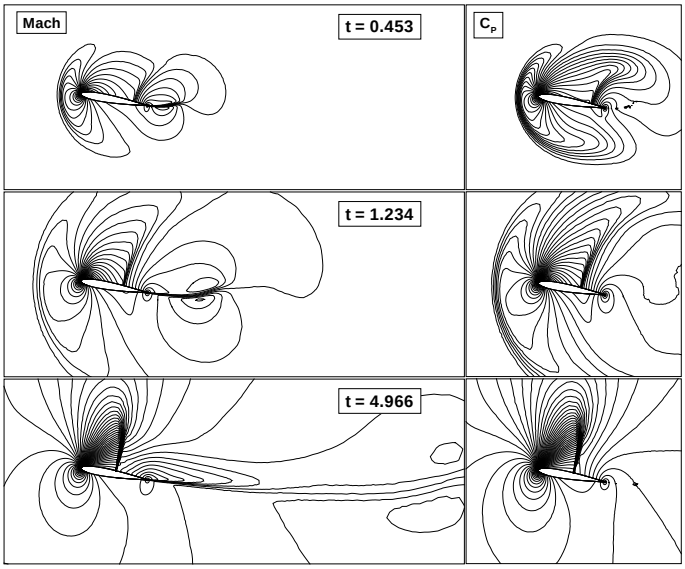
Figure 13: Impulsively-started NACA 0012 for Mach = 0.7, $\alpha = 10^{\circ}$: close-up of the computational grids at three different time levels (a) and reference grid (b).

time step for both the BDF1 and BDF2 schemes. For BDF2, the solution is almost indistinguishable from the reference also for $\Delta t = 2\Delta t_0$. Therefore, the simulations were run using the Ad_2 adaptation procedure with $\Delta t = \Delta t_0 = 0.0103$ and the BDF1 time integration scheme.

Fig. 13 shows the adapted grids for the simulation with the minimum element area equal to 3×10^{-5} at three different dimensionless time levels. These correspond to the minimum of the lift coefficient ($t = 0.453$, 9 213 nodes and 18 196 elements), a condition



(a) Adapted grid



(b) Reference grid

Figure 14: Impulsively-started NACA 0012 for Mach $=$ 0.7, $\alpha = 10^\circ$: contour line of the Mach number and of the pressure coefficient C_P at different time levels. Adapted grids (a) and reference grid (b) solutions.

where the influence of the start-up vortex begins to be negligible ($t = 1.234$, grid: 8 360 nodes and 16 494 elements) and the final configuration ($t = 4.966$, grid: 16 986 nodes and 33 733 elements), respectively. Fig. 14 shows the Mach number and the pressure coefficient C_P in the flow field at the three time levels. The $C_P(t)$ profile over the airfoil is

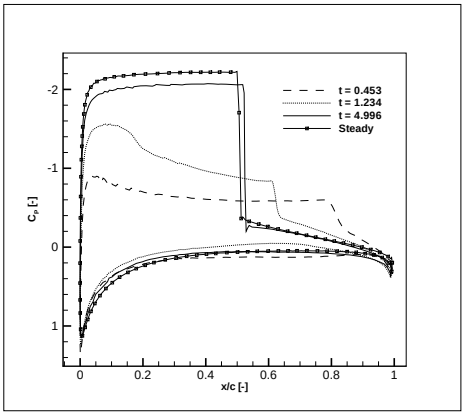


Figure 15: Impulsively-started NACA 0012. Mach = 0.7, $\alpha=10^\circ$: pressure coefficient over the airfoil at different time level for adapted grids.

depicted in Fig. 15 for the three considered states. The airfoil lift coefficient is plotted in Fig. 16 as a function of the dimensionless time.

A shock wave is formed on the top of the airfoils and terminate a supersonic region. The shock wave evolution is nicely captured by the adapted grid scheme, whereas in the reference dense grid computations the shock is smeared far from the airfoil. The start-up vortex is generated at $t=0$; its evolution is well captured by the adaptive scheme, which is also capable of refining the grid close to the slip line from the trailing edge. With reference to Fig. 13, in adapted grid simulations a large number of nodes is added to capture the pressure wave in front of the airfoil which is clearly visible at time $t=1.234$); this accounts for the higher computational time. The propagation of this disturbance does not affect significantly the overall value of the lift coefficient, which is used here for comparisons. The above can be appreciated in Fig. 17, where the total number of grid nodes at each

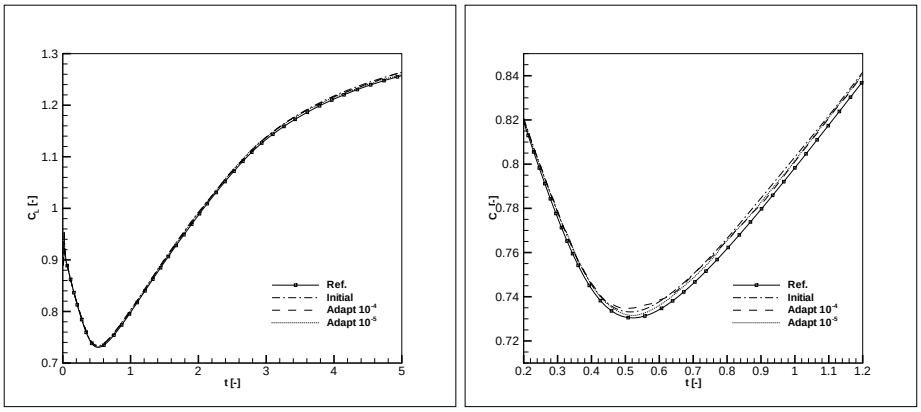
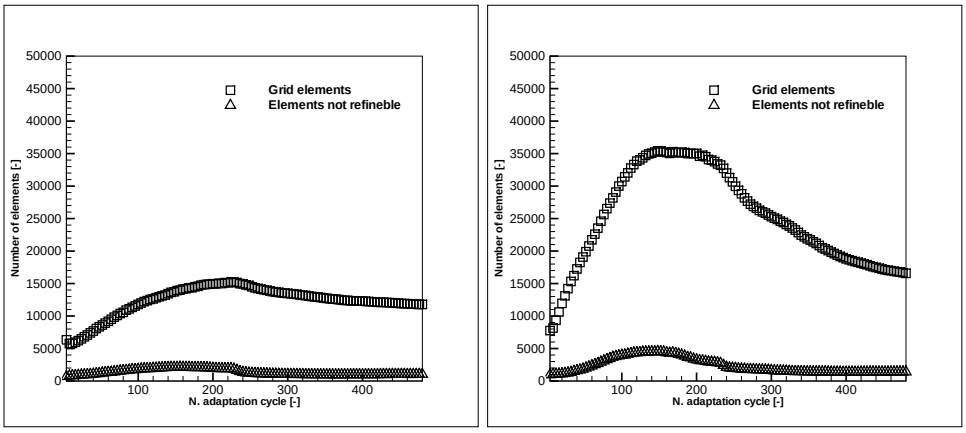


Figure 16: Impulsively-started NACA 0012. Mach = 0.7, $\alpha=10^\circ$: lift coefficient time history for the reference grid against adapted grids with minimum reference length of 3×10^{-5}.

https://doi.org/10.4208/cicp.181010.180511s



(a) $h_{\min}=10^{-4}$ (b) $h_{\min}=10^{-5}$

Figure 17: Mach $=0.7$, $\alpha=10^{\circ}$: number of grid elements and of not refineable elements as a function of the adaptation step for element size limit of 10^{-4} (left) and 10^{-5} (right).

adaptation step is shown. In the finer grid case, a large number of nodes is inserted during adaptation cycles between 100 and 200 to capture the pressure wave dynamics. The CPU time required to run each simulation is 9 hours, 19 minutes and 51 s for the reference simulations, which corresponds to an average of 110s for time step (305 time steps total), 3 hours, 21 minutes and 29 s (25 s $\times$ 486) for the adapted-grid simulation with minimum element area of 3×10^{-4} and 14 hours, 18 minutes and 44 s (106 s $\times$ 486) for a minimum area of 3×10^{-5} on a single core of an Intel Xeon QuadCore 3.166GHz. The adoption of the adaptation procedure is found not to influence the (average) computational effort per time step.

5 Conclusions

A finite-volume ALE solver for two-dimensional compressible flows over unstructured grid was used to compute vortical flows around airfoils. The numerical scheme implements a novel grid alteration procedure, which includes edge-swapping, node insertion and node deletion, to modify the local grid spacing to minimize the numerical error. The solution at the new time level is obtained without resorting to interpolation between the diverse grids, thanks to an original interpretation of the grid alteration process within the ALE framework. This methodology has also been used in other works in which simulations with large displacements of bodies were involved in [8, 9, 15].

Numerical simulations of compressible transonic and supersonic flow fields were presented to assess the suitability of the proposed approach to the numerical simulations of unsteady vortical flows of compressible fluids, with specific reference to the unsteady case of a start-up vortex from an impulsively started airfoil. The numerical results confirm the correctness of the proposed methodology and are comparable to fixed-grid re-

sults over finer grids. The present adaptation procedure can be formally extended to three-dimensional flows with no modification. Current research activities are devoted to the evaluation of the interface velocities resulting from edge-swapping, node insertion and deletion in three-dimensional grids made of tetrahedral elements.

References

[1] T. J. Baker. Mesh adaptation strategies for problems in fluid dynamics. Finite Elements Analysis and Design, 25:243–273, 1997.
[2] M. J. Castro-Diaz, F. Hect, B. Mohammad, and O. Pironneau. Anisotropic unstructured mesh adaptation for flow simulations. Int. J. Numer. Meth. Fluids., 25:475–491, 1997.
[3] J. Donea. An arbitrary Lagrangian-Eulerian finite element method for transient fluid–structure interactions. Comp. Meth. Appl. Mech. Engng., 33:689–723, 1982.
[4] J. Donea, A. Huerta, J.-Ph. Ponthot, and A. Rodríguez-Ferran. Arbitrary Lagrangian-Eulerian methods. In R. Stein, E. de Borst, and T.J.R. Hughes, editors, The Encyclopedia of Computational Mechanics, volume 1, chapter 14, pages 413–437. Wiley, 2004.
[5] E. Godlewski and P. A. Raviart. Numerical Approximation of Hyperbolic Systems of Conservation Laws. Springer-Verlag, New York, 1994.
[6] G. P. Warren, W. K. Anderson, J. T. Thomas, and S. L. Krist. Grid convergence for adaptive methods. In AIAA 10^{th} Computational Fluid Dynamics Conference, 1991. AIAA Paper 91-1592.
[7] H. G. Habashi, J. Dompierre, Y. Bourgault, D. Ait-Ali-Yahia, M. Fortin, and M. G. Vallet. Anisotropic mesh adaptation: towards user-indipendent, mesh indipendent and solver-indipendent CFD. Part I: General principles. Int. J. Num. Meth. Fluids, 32(6):725–744, 2000.
[8] D. Isola, A. Guardone, and G. Quaranta. Arbitrary Lagrangian Eulerian formulation for grids with variable topology. In E. Oñate M. Papadrakakis B. Schrefler, editor, Proceedings of the III Int. Conf. on Computational Methods for Coupled Problems in Science and Engineering ECCOMAS COUPLED PROBLEMS 2009. CIMNE, Barcelona, 2009.
[9] D. Isola, A. Guardone, and G. Quaranta. An ALE scheme without interpolation for moving domain with adaptive grids. In 40^{th} Fluid Dynamics Conference and Exhibit, 2010.
[10] R. J. LeVeque. Finite Volume Methods for Conservation Laws and Hyperbolic Systems. Cambridge University Press, 2002.
[11] R. Lohner. Mesh adaptation in fluid mechanics. Engineering Fracture Mechanics, 50:819–847, 1995.
[12] A. Gessow and G. C. Meyer. Aerodynamics of the helicopter. Frederick Ungar Publishing, 1952.
[13] J. Peraire, M. Vadhati, K. Morgan, and O. C. Zienkiewicz. Adaptive remeshing for compressible flow computations. J.Comput. Phys., 72:449–466, 1987.
[14] S. Pirzadeh. Unstructured viscous grid genereration by the advancing layers method. AIAA J., 32(8):1735–1737, 1994.
[15] G. Quaranta, D. Isola, and A. Guardone. Numerical simulation of the opening of aerodynamic control surfaces with two-dimensional unstructured adaptive meshes. In 5th European Conference on Computational Fluid Dynamics - ECCOMAS CFD 2010, 2010.
[16] P. L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. J. Comput. Phys., 43:357–372, 1981.

[17] V. Selmin. The node-centred finite volume approach: bridge between finite differences and finite elements. Comp. Meth. Appl. Mech. Engng., 102:107–138, 1993.
[18] P. R. Spalart. Airplane trailing vortices. Ann. Rev. Fluid Mech., 30:107–138, 1998.
[19] P. K. Sweby. High resolution schemes using flux limiters for hyperbolic conservation laws. SIAM J. Numer. Anal., 21:985–1011, 1984.
[20] J. F. Thompson. Numerical Grid Generation: Foundations and Applications. Elsevier Science Publishing, 1985.
[21] B. van Leer. Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second order scheme. J. Comput. Phys., 14:361–370, 1974.
[22] V. Venkatakrishnan and D. J. Mavriplis. Implicit method for the computation of unsteady flows on unstructured grids. J. Comput. Phys., 127:380–397, 1996.
[23] M. Fossati, A. Guardone, and L. Vigevano. A node-pair finite element/finite volume mesh adaptation technique for compressible flows. In 40th Fluid Dynamics Conference and Exhibit, 2010.
[24] N. P. Weatherill, O. Hassan, M. Marchant, and D. Marcum. Adaptive inviscid solutions for aerospace geometries on efficiently generated unstructured tetrahedral meshes. AIAA Paper 93-3390, 1993.
[25] B. E. Webster, M. S. Shepard, Z. Rhusak, and J. E. Flaherty. Automated adaptive time discontinuous finite element method for unsteady compressible airfoil. AIAA J., 32:748–757, 1994.
[26] G. Xia, D. Li, and C. L. Merkle. Anisotropic grid adaptation on unstructured meshes. In 39th Aerospace Sciences Meeting and Exhibit, 2001. AIAA Paper 2001-0443.