Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 197 (2008) 1219–1232
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

Dynamic boundary conditions in computational fluid dynamics

Mario A. Storti *, Norberto M. Nigro, Rodrigo R. Paz, Lisandro D. Dalcín

Centro Internacional de Métodos Computacionales en Ingeniería (CIMEC), INTEC(CONICET-UNL), Güemes 3450, S3000GLN Santa Fe, Argentina

Received 27 April 2007; received in revised form 11 October 2007; accepted 26 October 2007
Available online 12 November 2007

Abstract

The number and type of boundary conditions to be used in the numerical modeling of fluid mechanics problems is normally chosen according to a simplified analysis of the characteristics, and also from the experience of the modeler. The problem is harder at inflow/outflow boundaries which are, in most cases, artificial boundaries, so that a bad decision about the boundary conditions to be imposed may affect the precision and stability of the whole computation. For inviscid flows, the analysis of the sense of propagation in the normal direction to the boundaries gives the number of conditions to be imposed and, in addition, the conditions that are "*absorbing*" for the waves impinging normally to the boundary. In practice, it amounts to counting the number of positive and negative eigenvalues of the advective flux Jacobian projected onto the normal. The problem is even harder when the number of incoming characteristics varies during the computation, and the correct treatment of these cases poses both mathematical and practical problems. One example considered here is a compressible flow where the flow regime at a certain part of an inlet/outlet boundary can change from subsonic to supersonic and the flow can revert. In this work the technique for dynamically imposing the correct number of boundary conditions along the computation, using Lagrange multipliers and penalization, is discussed and several numerical examples are presented.

Keywords: Finite elements; Computational fluid dynamics; Absorbing boundary conditions

1. Introduction

Deciding how many and which boundary conditions to impose at each part of an artificial boundary is often a difficult problem. This decision is made from the number of incoming characteristics n_+ and the quantities known for each problem. If the number of conditions imposed on the boundary is in excess they are absorbed through spurious shocks at the boundary. On the other hand, if less conditions are imposed, then the problem is mathematically ill posed. Even if the number of imposed boundary conditions is correct, this does not guarantee that the boundary conditions are non-reflective.

When dealing with models in infinite domains an artificial boundary distant as far as possible from the region of interest must be introduced. The simplest choice is to impose a boundary condition, assuming that the flow far from the region of interest is undisturbed. However, the boundary condition can be freely chosen so as to give the best solution for a given position of the boundary. Boundary conditions that tend to give the solution as if the domain were infinite are called generally "*absorbing*" (ABC) or "*non reflective*" (NRBC). ABC's tend to give a better solution for a given position of the artificial boundary or, in other words, they allow to put the artificial boundary closer to the region of interest for a given admissible error. Of course, the advantage of putting the artificial boundary closer to the region of interest is the reduction in computational cost. However, in some cases, like for instance the solution of the Helmholtz equation on exterior domains, using absorbing boundary conditions is required since using a non absorbing boundary condition (like Dirichlet or Neumann) may lead to a

* Corresponding author. Tel.: +54 342 4511594; fax: +54 342 4511169.
E-mail addresses: mstorti@intec.unl.edu.ar (M.A. Storti), nnigro@intec.unl.edu.ar (N.M. Nigro), rodrigop@intec.unl.edu.ar (R.R. Paz), dalcinl@intec.unl.edu.ar (L.D. Dalcín).
URL: http://www.cimec.org.ar (M.A. Storti).

0045-7825/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2007.10.014

lack of convergence of the problem, because these conditions are completely reflective and therefore, wave energy is trapped in the domain, producing false resonance modes.

There are basically two approaches for the design of ABC's, *global* and *local*. Global boundary conditions are usually more accurate but expensive. In the limit, a global ABC may reproduce the effect of the whole external problem onto the boundary, i.e., even maintaining a fixed position of the artificial boundary the ABC may give a convergent solution while refining the interior mesh. In general these ABC's are *non-local*, i.e., its discrete operator is a dense matrix. Global boundary conditions exist and are popular for the simpler linear operators, like potential flow problems and frequency domain analysis of wave problems, like the Helmholtz equations for acoustics or the Maxwell equations [1–7].

The discrete operator for local absorbing boundary conditions is usually sparse but has a lower order accuracy and, in general, it is needed to move the artificial boundary condition to infinity while refining meshes in order to make the whole algorithm convergent. These kind of ABC's are popular for more complex non-linear fluid dynamic problems, like compressible or incompressible, Navier–Stokes equations or the inviscid Euler equations. An excellent review has been written by Tsynkov [8].

In order to have an ABC not any n_+ conditions must be imposed at the boundary but exactly those n_+ corresponding to the incoming characteristics. This can be determined through an eigenvalue decomposition problem of the advective flux Jacobian at the boundary.

In many cases, the number of incoming characteristics may change during the computation. For instance, in compressible flow it is common that the flow goes from subsonic to supersonic in certain parts of the outlet boundary. In 3D, this means passing from one imposed boundary condition to none.

In more complex problems, several combinations of regimes can be attained: subsonic inlet, supersonic inlet, subsonic outlet, supersonic outlet. A typical case where this can happen is the free fall of a blunt symmetrical object like an ellipse, for instance. If the body starts from rest, it will initially accelerate and, depending on the size and relation between the densities of the body and the surrounding atmosphere, it may reach the supersonic regime. As the body falls, even at subsonic speeds, its angle of attack tends to increase until eventually it stalls, and then falls towards its rear part, and repeating the process in a characteristic movement that recalls the falling of tree leaves. During the falling, the speed of the object varies periodically, accelerating when the angle of attach is small and the body experiences little drag, and decelerating when the angle of attack is large. For a supersonic fall, the regime may change from supersonic to subsonic and back during the fall. In addition, if the problem is solved in a reference frame attached to the body, the unperturbed flow may come from every direction relative to the body's axis. In this way, the regime and direction of the flow at a given point of the boundary may change through the whole possible combinations.

Another example is the modeling of the ignition of a rocket exhaust nozzle. In this case, the condition at the outlet boundary changes from rest to supersonic flow as the shock produced at the throat reaches the exterior boundary.

For transport of scalars, this behavior may happen if the transport velocity varies in time and the flow gets reverted at the boundary. One such situation is when modeling the transport of a scalar like smoke or contaminant concentration in a building with several openings under an external wind. Assume that the concentration of solid particles or contaminant is so low that its influence on the fluid is negligible so that the movement of the fluid inside the building can be solved first and then the transport equation for the scalar, taking the velocity of the fluid as the transport velocity. As the flow in the interior fluctuates, the normal component of velocity at a given opening may reverse its direction.

The change of the number of imposed boundary conditions at a given point of the boundary is hard to implement from the computational point of view since it involves the change of the structure of the Jacobian matrix. The solution proposed here is to impose these conditions through Lagrange multipliers or penalization techniques. The main objective of this paper is to discuss numerical aspects related to the use of this techniques, to discuss specific issues relative to the physical problems described above, and to show some numerical examples.

2. General advective–diffusive systems of equations

Consider an advective diffusive system of equations in conservative form

$$\frac{\partial \mathscr{H}(\mathbf{U})}{\partial t} + \frac{\partial \mathscr{F}_{c,j}(\mathbf{U})}{\partial x_j} = \frac{\partial \mathscr{F}_{d,j}(\mathbf{U}, \nabla \mathbf{U})}{\partial x_j} + \mathbf{G}. \tag{1}$$

Here $\mathbf{U} \in \mathbb{R}^n$ is the state vector, t is time, $\mathscr{F}_{c,j}$ and $\mathscr{F}_{d,j}$ are the advective and diffusive fluxes respectively, $\mathbf{G}$ is a source term including, for instance, gravity acceleration or external heat sources, and x_j are the spatial coordinates.

The notation is standard, except perhaps for the "*generic enthalpy function*" $\mathscr{H}(\mathbf{U})$. The inclusion of the enthalpy function allows the inclusion of conservative equations in terms of non-conservative variables. Some well-known advective diffusive systems of equations may be cast in this general setting as follows.

2.1. Linear advection diffusion

The heat advection–diffusion equation in terms of temperature can be put in this form through the definitions

$$
\begin{aligned}
&\mathbf{U} = T,\\
&\mathscr{H}(\mathbf{U}) = \rho C_p T,\\
&\mathscr{F}_{c,j}(\mathbf{U}) = \rho C_p T u_j,\\
&\mathscr{F}_{d,j}(\mathbf{U},\nabla\mathbf{U}) = -q_j = -k\frac{\partial T}{\partial x_j},
\end{aligned}
\tag{2}
$$

where $\mathbf{U}$ is the state vector, C_p the specific heat, u_j a component of a given velocity field $\mathbf{u}$, T the temperature (the unknown field), $\mathbf{q}$ the heat flux vector and k the thermal conductivity of the medium.

2.2. Gas dynamics equations

Gas dynamic equations of a compressible flow can be put in conservative form with the following definitions:

$$
\begin{aligned}
&\mathbf{U}_p = [\rho,\mathbf{u},p]^T,\\
&\mathbf{U} = \mathbf{U}_c = [\rho,\rho\mathbf{u},\rho e]^T,\\
&\mathscr{H}(\mathbf{U}_p) = \mathbf{U},\\
&\mathscr{F}_{c,j}n_j = \begin{bmatrix} \rho(\mathbf{u}\cdot\hat{\mathbf{n}}) \\ \rho\mathbf{u}(\mathbf{u}\cdot\hat{\mathbf{n}}) + p\hat{\mathbf{n}} \\ (\rho e + p)(\mathbf{u}\cdot\hat{\mathbf{n}}) \end{bmatrix},\\
&\mathscr{F}_{d,j}(\mathbf{U},\nabla\mathbf{U})n_j = \begin{bmatrix} 0 \\ \mathbf{T}\cdot\hat{\mathbf{n}} \\ T_{ik}u_k n_i - q_i n_i \end{bmatrix}.
\end{aligned}
\tag{3}
$$

Note that even if the equations are put in terms of conservative variables, the diffusive and convective fluxes are expressed in term of the primitive variables $\mathbf{U}_p = [\rho,\mathbf{u},p]^T$ and where ρ is the density, p the pressure, e the specific total energy, $\mathbf{T}$ the Newtonian viscous stress tensor and $\hat{\mathbf{n}}$ the normal vector (outward) to a given surface. However, the fluxes can be thought as implicitly depending on the conservative variables, since the relation $\mathbf{U}_c(\mathbf{U})$ is one to one. Now, the conservation equations can be also thought in terms of any other set of variables, for instance the primitive variables, if the "*enthalpy function*" $\mathscr{H}(\mathbf{U}_p) = \mathbf{U}_c(\mathbf{U}_p)$ is introduced.

3. Variational formulation

In this section, the variational formulation of the compressible Navier–Stokes equations using SUPG (Streamline Upwind Petrov–Galerkin [9,10]) finite element method and the shock capturing operator [11] is presented. Consider a finite element discretization of the Ω into subdomains Ω^e, $e = 1,2,\ldots,n_{\text{elem}}$. Based on this discretization, the finite element function spaces for the trial solutions and for the weighting functions, $\mathscr{V}^h$ and $\mathscr{S}^h$ respectively, can be defined. These function spaces are selected as subsets of $[\mathbf{H}^{1h}(\Omega)]^{n_{\text{dof}}}$ when taking Dirichlet boundary conditions, where $\mathbf{H}^{1h}(\Omega)$ is the finite dimensional Sobolev functional space over Ω, and $n_{\text{dof}} = n_{\text{sd}} + 2$ is the number of dof's in the continuum problem (n_{sd} is the number of spatial dimensions).

The stabilizing finite element formulation of the quasi-linear form of (3) is written as follows: find $\mathbf{U}^h \in \mathscr{S}^h$ such that $\forall\mathbf{W}^h \in \mathscr{V}^h$

$$
\begin{aligned}
&\int_\Omega \mathbf{W}^h\cdot\left(\frac{\partial\mathscr{H}(\mathbf{U}^h)}{\partial t} + \frac{\partial\mathscr{F}^h_c}{\partial x_i}\right)\mathrm{d}\Omega = \int_\Omega \mathbf{W}^h\cdot\left(\frac{\partial\mathscr{F}^h_d}{\partial x_i} + \mathbf{G}\right)\mathrm{d}\Omega,\\
&\int_\Omega \mathbf{W}^h\cdot\left(\frac{\partial\mathscr{H}(\mathbf{U}^h)}{\partial t} + \mathbf{A}^h_i\frac{\partial\mathbf{U}^h}{\partial x_i} - \mathbf{G}\right)\mathrm{d}\Omega\\
&\quad - \int_\Omega \frac{\partial\mathbf{W}^h}{\partial x_i}\cdot\mathbf{K}^h_{ij}\frac{\partial\mathbf{U}^h}{\partial x_j}\mathrm{d}\Omega - \int_{\Gamma_h}\mathbf{W}^h\cdot\mathbf{H}^h\,\mathrm{d}\Gamma\\
&\quad + \sum_{e=1}^{n_{\text{elem}}}\int_{\Omega^e}\tau(\mathbf{A}^h_k)^T\frac{\partial\mathbf{W}^h}{\partial x_k}\\
&\quad\cdot\left\{\frac{\partial\mathbf{U}^h}{\partial t} + \mathbf{A}^h_i\frac{\partial\mathbf{U}^h}{\partial x_i} - \frac{\partial}{\partial x_i}\left(\mathbf{K}^h_{ij}\frac{\partial\mathbf{U}^h}{\partial x_j}\right) - \mathbf{G}\right\}\mathrm{d}\Omega\\
&\quad + \sum_{e=1}^{n_{\text{elem}}}\int_{\Omega^e}\delta_{\text{shc}}\frac{\partial\mathbf{W}^h}{\partial x_i}\cdot\frac{\partial\mathbf{U}^h}{\partial x_i}\mathrm{d}\Omega = \mathbf{0},
\end{aligned}
\tag{4}
$$

where

$$
\begin{aligned}
\mathscr{S}^h &= \{\mathbf{U}^h | \mathbf{U}^h \in [\mathbf{H}^{1h}(\Omega)]^{n_{\text{dof}}}, \mathbf{U}^h|_{\Omega^e} \in [P^1(\Omega^e)]^{n_{\text{dof}}},\\
&\qquad \mathbf{U}^h = \mathbf{g} \quad \text{on } \Gamma_g\},\\
\mathscr{V}^h &= \{\mathbf{W}^h | \mathbf{W}^h \in [\mathbf{H}^{1h}(\Omega)]^{n_{\text{dof}}}, \mathbf{W}^h|_{\Omega^e} \in [P^1(\Omega^e)]^{n_{\text{dof}}},\\
&\qquad \mathbf{W}^h = \mathbf{0} \quad \text{on } \partial\Omega_g\},
\end{aligned}
\tag{5}
$$

and where matrices $\mathbf{A}_i$ and $\mathbf{K}_{ij}$ are defined as

$$
\frac{\partial\mathscr{F}^a}{\partial x_i} = \frac{\partial\mathscr{F}^a}{\partial\mathbf{U}}\frac{\partial\mathbf{U}}{\partial x_i} = \mathbf{A}_i\frac{\partial\mathbf{U}}{\partial x_i}
\tag{6}
$$

and

$$
\frac{\partial\mathscr{F}^d}{\partial x_i} = \frac{\partial\mathscr{F}^d}{\partial\mathbf{U}}\frac{\partial\mathbf{U}}{\partial x_i} = \mathbf{K}_{ij}\frac{\partial\mathbf{U}}{\partial x_i}.
\tag{7}
$$

The first three terms inside the first two integrals in the variational formulation (4) constitute the Galerkin formulation of the problem and the third integral accounts for the Neumann boundary conditions. The first series of element level integrals in (4) are the SUPG stabilization terms added to prevent spatial oscillations in the advection-dominated range. The second series of element level integrals in (4) are the shock capturing terms added to assure the stability at high Mach and Reynolds number flows, specially to suppress spurious overshoot and undershoot effects in the vicinity of discontinuities.

Various options for calculating the stabilization parameters and defining the shock capturing terms in the context of the SUPG formulation were introduced in [12]. In this section some of these options are described. The first one is the standard SUPG intrinsic time tensor τ that was given in [13] as a slightly modified version of the stabilization parameters introduced in [9,10,14]. In this case, this matrix is defined as $\tau = \max[\mathbf{0},\tau_a - \tau_d - \tau_\delta]$, with each τ_x taking into account the advective and diffusive effects and also avoiding the duplication of the shock capturing operator

and the streamline upwind operator. These matrices are defined as

$$\tau_a = \frac{h}{2(c+|\mathbf{u}|)}\mathbf{I}, \quad \tau_d = \frac{\sum_{j=1}^{n_{sd}} \beta_j^2 \mathrm{diag}(\mathbf{K}_{jj})}{(c+|\mathbf{u}|)^2}\mathbf{I},$$
$$\tau_\delta = \frac{\delta_{shc}}{(c+|\mathbf{u}|)^2}\mathbf{I}, \tag{8}$$

where c is the acoustic speed, $h = 2|\mathbf{u}|\left(\sum_{a=1}^{n_{en}} |\ \mathbf{u}\cdot\nabla N_a|\right)^{-1}$ is the element size computed here as the element length in the direction of velocity using for its definition the multi-lineal trial function N_a, $\beta = \nabla\|\mathbf{U}\|^2 / \|\ \nabla\|\mathbf{U}\|^2\|$ and δ_{shc} is the shock capturing parameter defined in the next paragraph. The τ matrix computation is already an open problem because it is not possible to diagonalize the system of equations. It follows some heuristics arguments based on the maximum value of the set of eigenvalues of the advective Jacobian matrices for the characteristic velocity, some measure of the element size that may not be very well justified but is equivalent to any other element size and some mechanism able to remove stabilization when physical diffusion is present.

The design of the shock capturing operator is also an open problem. Two versions are presented here: an isotropic operator and an anisotropic one, both proposed by Tezduyar et al. in [15]. A unit vector oriented with the density gradient is defined as $\mathbf{j} = \nabla\rho^h / |\nabla\rho^h|$ and a characteristic length as $h_{JGN} = 2\left(\sum_{a=1}^{n_{en}} |\mathbf{j}\cdot\nabla N_a|\right)^{-1}$, where N_a is the finite element shape function corresponding to the node a. The above cited isotropic shock capturing factor included in (4) is then defined as

$$\delta_{shc} = \frac{h_{JGN}}{2} u_{char} \left(\frac{|\nabla\rho^h| h_{JGN}}{\rho_{ref}}\right)^{\beta}, \tag{9}$$

where $u_{char} = |\mathbf{u}| + c$ is the characteristic velocity defined as the addition of the flow velocity magnitude and the acoustic speed. Here ρ_{ref} is the Gaussian point interpolated density and β is usually taken in the range $1 \leqslant \beta \leqslant 2$ according to the sharpness of the discontinuity to be captured as suggested in Ref. [15]. $\beta = 1$ was used in this study.

The anisotropic version of the shock capturing term in (4) is changed as follows

$$\sum_{e=1}^{n_{elem}} \int_{\Omega^e} \frac{\partial \mathbf{W}^h}{\partial x_i} \mathbf{j}_i \delta_{shc} \mathbf{j}_k \frac{\partial \mathbf{U}^h}{\partial x_k} \mathrm{d}\Omega. \tag{10}$$

The anisotropic shock capturing term showed good behavior. Nevertheless, for some applications, both terms may be needed, the isotropic one weighted by a factor close to 0.2 or lower.

4. Absorbing boundary conditions

For steady simulations using time-marching algorithms, it can be shown that the error going towards the steady state propagates like waves, so that absorbing boundary conditions help to eliminate the error from the computational domain. In fact, it can be shown that for strongly advective problems absorption at the boundaries is usually the main mechanism of error reduction (the other mechanism is physical or numerical dissipation in the interior of the computational domain). It has been shown that in such cases the rate of convergence can be directly related to the "*transparency*" of the boundary condition [16]. In general, absorbing boundary conditions are based on an analysis of the characteristic waves. A key point is to determine which of them are *incoming* and which are *outgoing*. Absorbing boundary conditions exist from the simplest first order ones based on a plane wave analysis at a certain smooth portion of the boundary (as will be described below), to the more complex ones that tend to match a full analytic solution of the problem in the external region with that obtained in the internal region. In this paper the usage of absorbing boundary conditions is accomplished in situations where the conditions at the boundary change, so as the number of incoming and outgoing characteristic waves varies during the temporal evolution of the problem, or even when the conditions at the boundary are not well known *a priori*.

4.1. Advective diffusive systems in 1D

Consider a pure advective system of equations in 1D, i.e., $\mathscr{F}_{d,j} \equiv 0$

$$\frac{\partial \mathscr{H}(\mathbf{U})}{\partial t} + \frac{\partial \mathscr{F}_{c,x}(\mathbf{U})}{\partial x} = 0 \quad \text{in } [0, L]. \tag{11}$$

If the system is "*linear*", i.e., $\mathscr{F}_{c,x}(\mathbf{U}) = \mathbf{A}\mathbf{U}$, $\mathscr{H}(\mathbf{U}) = \mathbf{C}\mathbf{U}$ ($\mathbf{A}$ and $\mathbf{C}$ do not depend on $\mathbf{U}$), a first order linear system is obtained

$$\mathbf{C}\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{U}}{\partial x} = 0. \tag{12}$$

The system is "*hyperbolic*" if $\mathbf{C}$ is invertible, $\mathbf{C}^{-1}\mathbf{A}$ is diagonalizable with real eigenvalues. If this is the case, it is possible to make the following eigenvalue decomposition for $\mathbf{C}^{-1}\mathbf{A}$

$$\mathbf{C}^{-1}\mathbf{A} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1}, \tag{13}$$

where $\mathbf{S}$ is real and invertible and $\mathbf{\Lambda}$ is real and diagonal. If new variables are defined $\mathbf{V} = \mathbf{S}^{-1}\mathbf{U}$, then Eq. (12) becomes

$$\frac{\partial \mathbf{V}}{\partial t} + \mathbf{\Lambda}\frac{\partial \mathbf{V}}{\partial x} = 0. \tag{14}$$

Now, each equation is a linear scalar advection equation

$$\frac{\partial v_k}{\partial t} + \lambda_k \frac{\partial v_k}{\partial x} = 0 \quad \text{(no summation over } k\text{)}. \tag{15}$$

v_k are the "*characteristic components*" and λ_k are the "*characteristic velocities*" of propagation.

4.2. Linear 1D absorbing boundary conditions

Assuming $\lambda_k \neq 0$, the absorbing boundary conditions are, depending on the sign of λ_k,

$$\begin{aligned} &\text{if } \lambda_k > 0 : v_k(0) = \bar{v}_{k0}; \quad \text{no boundary condition at } x = L, \\ &\text{if } \lambda_k < 0 : v_k(L) = \bar{v}_{kL}; \quad \text{no boundary condition at } x = 0. \end{aligned} \tag{16}$$

This can be put in compact form as

$$\begin{aligned} &\mathbf{\Pi}_V^+(\mathbf{V} - \overline{\mathbf{V}}_0) = 0; \quad \text{at } x = 0, \\ &\mathbf{\Pi}_V^-(\mathbf{V} - \overline{\mathbf{V}}_L) = 0; \quad \text{at } x = L, \end{aligned} \tag{17}$$

where $\mathbf{\Pi}_V^{\pm}$ are the *projection matrices onto the right/left-going characteristic modes* in the $\mathbf{V}$ basis,

$$\begin{aligned} &\mathbf{\Pi}_{V,jk}^+ = \begin{cases} 1; & \text{if } j = k \text{ and } \lambda_k > 0, \\ 0; & \text{otherwise}, \end{cases} \\ &\mathbf{\Pi}^+ + \mathbf{\Pi}^- = \mathbf{I}. \end{aligned} \tag{18}$$

It can be easily shown that they are effectively *projection matrices*, i.e., $\mathbf{\Pi}^{\pm}\mathbf{\Pi}^{\pm} = \mathbf{\Pi}^{\pm}$ and $\mathbf{\Pi}^{+}\mathbf{\Pi}^{-} = 0$. Coming back to the boundary condition at $x = L$ in the $\mathbf{U}$ basis, it can be written

$$\mathbf{\Pi}_V^- \mathbf{S}^{-1}(\mathbf{U} - \overline{\mathbf{U}}_L) = 0 \tag{19}$$

or, multiplying by $\mathbf{S}$ at the left

$$\mathbf{\Pi}_U^{\pm}(\mathbf{U} - \overline{\mathbf{U}}_{0,L}) = 0, \quad \text{at } x = 0, L, \tag{20}$$

where

$$\mathbf{\Pi}_U^{\pm} = \mathbf{S}\mathbf{\Pi}_V^{\pm}\mathbf{S}^{-1} \tag{21}$$

are the projection matrices in the $\mathbf{U}$ basis. These conditions are completely absorbing for 1D linear advection system of Eq. (12).

The rank of $\mathbf{\Pi}^+$ is equal to the number n_+ of positive eigenvalues, i.e., the number of right-going waves. Recall that the right-going waves are incoming at the $x = 0$ boundary and outgoing at the $x = L$ boundary. Conversely, the rank of $\mathbf{\Pi}^-$ is equal to the number n_- of negative eigenvalues, i.e., the number of left-going waves (incoming at $x = L$ and outgoing at the $x = 0$ boundary).

4.2.1. Numerical example. 1D compressible flow

The solution of 1D compressible flow in $0 \leqslant x \leqslant L = 4$ is considered. The undisturbed flow has a Mach number of 0.5 and at $t = 0$ there is a perturbation in the form of a Gaussian as follows

$$\mathbf{U}(x, t = 0) = \mathbf{U}_{\text{ref}} + \Delta\mathbf{U}\, e^{(x - x_0)/\sigma^2}, \tag{22}$$

where $\rho_{\text{ref}} = 1$, $u_{\text{ref}} = 0.5$, $p_{\text{ref}} = 0.714$, $(Ma_{\text{ref}} = 0.5)$ $\Delta\rho = \Delta p = 0$, $\Delta u = 0.1$, $R = 1$, $x_0 = 0.8$ and $\sigma = 0.3$. The evolution of this perturbation is simulated using $N = 50$ equal-spaced finite elements ($h = L/N = 0.08$) with SUPG stabilization and Crank–Nicholson temporal scheme with $\Delta t = 0.05$ (CFL number ≈ 0.84). As the flow is subsonic two conditions at inlet and one at outlet must be imposed. The results using standard ($p = p_{\text{ref}}$) and absorbing boundary conditions at outlet ($x = L$) are compared. In both cases, non-absorbing conditions ($\rho = \rho_{\text{ref}}$ and $u = u_{\text{ref}}$) were imposed at inlet ($x = 0$). Fig. 1 shows the evolution in time (in the form of an elevation view) of the velocity Fig. 2 when using the condition $p = p_{\text{ref}}$ at outlet, while Fig. 3 shows the results when using first order linear absorbing boundary conditions based on the unperturbed state. It can be seen that without absorbing boundary condition the perturbation reflects at both boundaries. Even after $t = 40$ a significant amount of perturbation is still inside the domain. At this point the perturbation has reflected four times at the boundaries. When using the absorbing boundary condition the perturbation is almost completely absorbed after it hits the outlet boundary. Note that the absorption is performed in two steps. First the perturbation splits in two components, one propagating downstream an another upstream. The first hits the outlet boundary and is absorbed, the other travels backwards, reflects at the inlet boundary and then travels to the outlet boundary, where it hits at $t = 4.5$. This shows that in 1D it is enough with only one absorbing boundary to have a strong dissipation of energy.

4.3. Multidimensional problems

For multidimensional problems a simplified 1D analysis can be done in the normal direction to the local boundary and with the flux Jacobian $\mathbf{A}$ in Eq. (13) replaced with its projection onto the exterior normal $\hat{\mathbf{n}}$, as follows

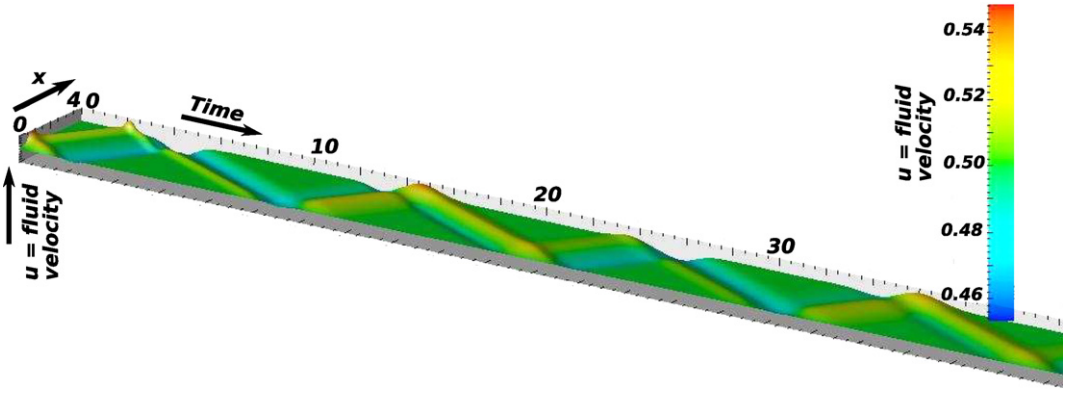

Fig. 1. Temporal evolution of axial velocity in 1D gas dynamics problem *without* absorbing boundary condition at outlet.

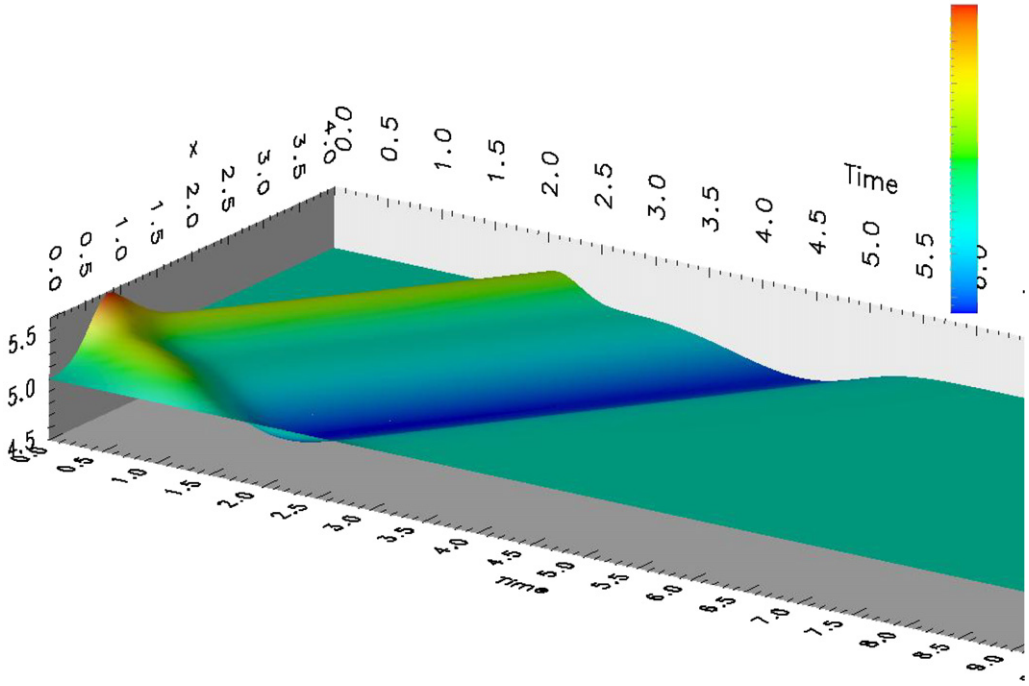

Fig. 2. Temporal evolution of axial velocity in 1D gas dynamics problem *with* absorbing boundary condition at outlet.

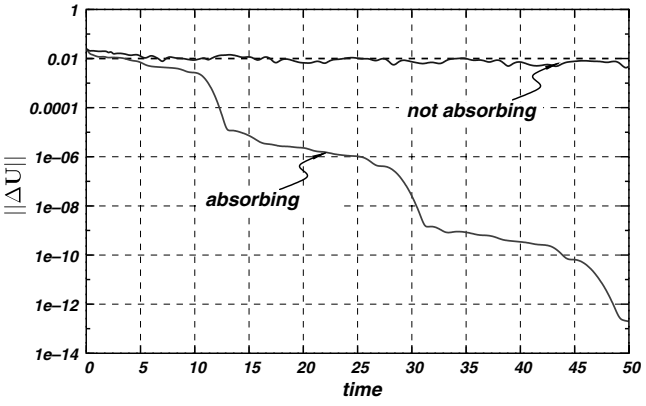

Fig. 3. Rate of converge of 1D gas dynamics problem *with* and *without* absorbing boundary conditions.

$$
\begin{aligned}
&\boldsymbol{\Pi}_n^-(\mathbf{U}-\overline{\mathbf{U}})=0,\\
&\boldsymbol{\Pi}_n^- = \mathbf{S}_n\boldsymbol{\Pi}_{Vn}^-\mathbf{S}_n^{-1},\\
&(\boldsymbol{\Pi}_{Vn}^-)_{jk} = \begin{cases} 1; & \text{if } j=k \text{ and } \lambda_j<0,\\ 0; & \text{otherwise,}\end{cases}\\
&\mathbf{C}^{-1}\mathbf{A}_n = \mathbf{S}_n\boldsymbol{\Lambda}_n\mathbf{S}_n^{-1}, \quad (\boldsymbol{\Lambda}_n \text{ diagonal}),\\
&\mathbf{A}_n = \mathbf{A}_l n_l.
\end{aligned}
\tag{23}
$$

These conditions are perfectly absorbing for perturbations reaching the boundary normal to the surface. For perturbations not impinging normally, the condition is partially absorbing, with a reflection coefficient that increases from 0 at normal incidence to 1 for tangential incidence.

4.4. Absorbing boundary conditions for non-linear problems

If the problem is non-linear, as the gas dynamics or shallow water equations, then the flux Jacobian $\mathbf{A}$ is a function of the state of the fluid, and then the same happens for the projection matrices $\boldsymbol{\Pi}^{\pm}$. If it is assumed that the flow is composed of small perturbations around a reference state $\mathbf{U}_{\text{ref}}$, then the projection matrix at the state $\mathbf{U}_{\text{ref}}$ can be computed

$$\boldsymbol{\Pi}(\mathbf{U}_{\text{ref}})_n^-(\mathbf{U}-\mathbf{U}_{\text{ref}})=0. \tag{24}$$

However, as long as the fluid state departs from the reference value the condition becomes less and less absorbing.

4.4.1. Numerical example. Varying section compressible 1D flow

Consider a one-dimensional flow in a tube with a contraction of 2:1. The inlet Mach number is 0.2 and the variation of area along the tube axis is

$$A(x) = A_0\left(1 - C\frac{\tanh(x - Lx/2)}{L_c}\right), \tag{25}$$

where A_0 is some (irrelevant) reference area, C is a constant given by $C=(\alpha-1)/(\alpha+1)$, $\alpha = A_{\text{in}}/A_{\text{out}}$ is the area ratio and $L_c = 0.136$ is a parameter controlling the width of the transition. Variables ρ and u are imposed at the inlet and consider different outlet conditions, namely

- *non-absorbing*, $p = \text{cnst}$,
- *absorbing linear* (see (20)), and
- *absorbing non-linear* (see (24)).

Figs. 3 and 4 show the evolution in time of the state vector increment ($\|\Delta\mathbf{U}\|$) for different absorbing and non-absorbing boundary conditions. Note that the absorbing linear condition behaves worst than the non-absorbing one, due to the fact that the state at the boundary has departed from the initial state, with which the projectors have been computed. This does not happens with the absorbing non-linear condition since it uses always the last computed state for the computation of the projection matrices.

4.5. Riemann based absorbing boundary conditions

Suppose that for a small interval $t \leqslant t' \leqslant t + \Delta t$ the state $\mathbf{U}(t)$ is taken as the reference state, then, during this interval $\mathbf{\Pi}^-(\mathbf{U}(t))$ is taken as the projection operator onto the incoming characteristics and the absorbing boundary conditions are

$$\mathbf{\Pi}^-(\mathbf{U}(t))\,(\mathbf{U}(t') - \mathbf{U}(t)) = 0. \tag{26}$$

But regarding the equivalent expression (19) it can be written as

$$\mathbf{l}_j(\mathbf{U}) \cdot \mathrm{d}\mathbf{U} = 0, \quad \text{if } \lambda_j < 0, \tag{27}$$

where $\mathbf{l}_j$ is the jth left eigenvalue of the normal flux Jacobian. Note that, as $\mathbf{l}_j$ is a function of $\mathbf{U}$, this is a differential form on the variable $\mathbf{U}$. If it happens that this is a *exact differential*, i.e.,

$$\mu(\mathbf{U})\,\mathbf{l}_j(\mathbf{U}) \cdot \mathrm{d}\mathbf{U} = \mathrm{d}w_j(\mathbf{U}) \tag{28}$$

for some non-linear function w_j and an "*integration factor*" $\mu(\mathbf{U})$, then it can be imposed

$$w_j(\mathbf{U}) = w_j(\mathbf{U}_{\mathrm{ref}}) \quad (\text{for } w_j \text{ an incoming char.}) \tag{29}$$

which would be an absorbing boundary condition for the whole non-linear regime. The functions w_j are often referred as "*Riemann invariants*" (RI) for the flux function.

For the 2D shallow water equations, the Riemann invariants are well known (see Ref. [17]). For 1D channel flow, Riemann invariants are known for a few channel shapes (rectangular and triangular). For general channel sections they are not known and in addition there is not a general numerical method for computing them. They could be computed by numerical integration of Eq. (28) along a path in state space, but the integration factor is not known.

Riemann invariants are known for the shallow water equations

$$w_\pm = \mathbf{u} \cdot \hat{\mathbf{n}} \pm 2\sqrt{gh}, \tag{30}$$

and for channel flow they, are known only for rectangular and triangular channel shapes. For the triangular case, RI are

$$w_\pm = \mathbf{u} \cdot \hat{\mathbf{n}} \pm 4\sqrt{gh}. \tag{31}$$

For the gas dynamics equations, the well known Riemann invariants are invariant only under isentropic conditions, so that they are not truly invariant. They are

$$w_\pm = u \pm \frac{2c}{\gamma - 1}. \tag{32}$$

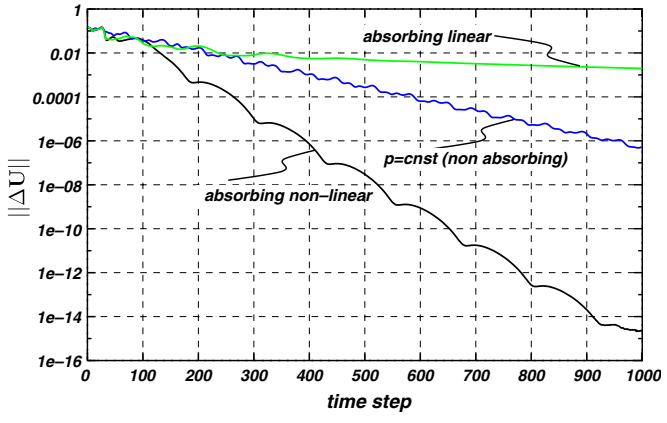

Fig. 4. Rate of converge of 1D gas dynamics problem in full non-linear regime with different kind of absorbing boundary conditions.

4.6. Absorbing boundary conditions based on last state

While integrating the discrete equations in time, the state of the fluid in the previous state can be taken as the reference state

$$\mathbf{\Pi}^-(\mathbf{U}^n)\,(\mathbf{U}^{n+1} - \mathbf{U}^n) = 0. \tag{33}$$

It is clear that the assumption of linearization is well justified, since in the limit of $\Delta t \to 0$ it should be $\mathbf{U}^{n+1} \approx \mathbf{U}^n$. In fact, (33) is equivalent, for $\Delta t \to 0$ to (27), so that if Riemann invariants exist, then this scheme preserves them in the limit $\Delta t \to 0$ and $\Delta x \to 0$. This strategy is called ULSAR (for *Use Last State As Reference*).

However, if this scheme is used in the whole boundary, then the flow in the domain is only determined by the initial condition, and it can drift in time due to numerical errors. Also, in a steady state of a certain regime, there is no way to guarantee that the regime will be obtained. For instance, to obtain the steady flow around an aerodynamic profile at a certain Mach number, the initial state with a non perturbed constant flow at that condition can be stated, but, it cannot be assured that the final steady flow will preserve that Mach number. In practice, a mix of the strategies are often used, with linear boundary conditions imposed at inlet regions and absorbing boundary conditions based on last state on the outlet regions.

4.6.1. Numerical example. ULSAR strategy keeps RI constant

Consider a 1D compressible flow example, as in Section 4.2.1, with $\rho_{\mathrm{ref}} = 1$, $u_{\mathrm{ref}} = 0.2$, $p_{\mathrm{ref}} = 0.714$, ($Ma_{\mathrm{ref}} = 0.2$), $\Delta\rho = \Delta p = 0$, $\Delta u = 0.6$, $R = 1$, $x_0 = 0.5L = 2$ and $\sigma = 0.3$. Note that this represents a perturbation in velocity that goes from $Ma = 0.2$ to 0.8, so that full non-linear effects are evidenced. The evolution of this perturbation is simulated using $N = 200$ equal-spaced finite elements

($h = L/N = 0.08$) with SUPG stabilization and Crank–Nicholson temporal scheme with $\Delta t = 0.02$ (CFL number ≈ 1.2). The values are dimensionless by selecting L, ρ_{ref} and u_{ref} as reference values for length, density and velocity. Absorbing boundary conditions based on the ULSAR strategy are applied at both ends $x = 0, L$. The values of the Riemann invariants (32) are computed there and they are plotted in Fig. 5. It can be seen that the incoming RI (the right going w_+) is kept approximately constant at the left boundary $x = 0$ and the same happens, *mutatis mutandis*, at the other boundary $x = L$. Convergence history is shown in Fig. 6. Note that absorption is very good, despite the full non-linear character of the flow.

4.7. Imposing non-linear absorbing boundary conditions

In this section, the integration of the absorbing boundary conditions in a numerical code is discussed. For linear systems, the discrete version of Eq. (12) is of the form

$$\begin{aligned} &\mathbf{C}\frac{\mathbf{U}_0^{n+1}-\mathbf{U}_0^n}{\Delta t}+\mathbf{A}\frac{\mathbf{U}_1^{n+1}-\mathbf{U}_0^{n+1}}{h}=0,\\ &\mathbf{C}\frac{\mathbf{U}_k^{n+1}-\mathbf{U}_k^n}{\Delta t}+\mathbf{A}\frac{\mathbf{U}_{k+1}^{n+1}-\mathbf{U}_{k-1}^{n+1}}{2h}0, \quad k \geqslant 1, \end{aligned} \tag{34}$$

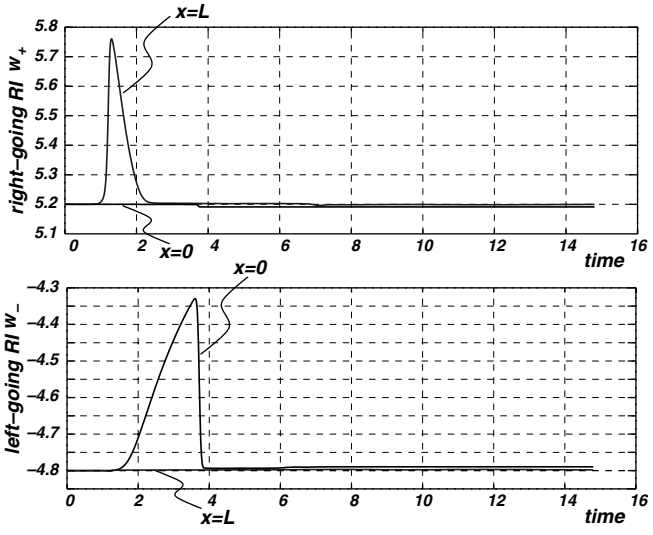

Fig. 5. Riemann invariants at boundaries with ULSAR ABC's.

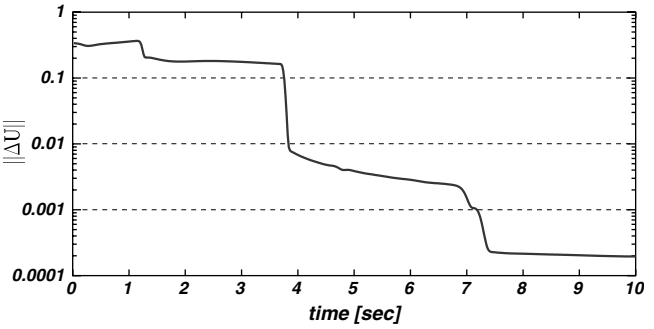

Fig. 6. Convergence history when using with ULSAR ABC's.

where $\mathbf{U}_k^n$ is the state at grid point k at time $t^n = n\Delta t$. It is assumed a constant mesh step size of h, i.e., $x_k = kh$, and the boundary located at the mesh node $x_0 = 0$. Several simplifications were assumed here, no source or upwind terms, and a simple discretization based on centered finite differences was used. Alternatively, it can be thought as a pure Galerkin FEM discretization with mass lumping. Also, backward Euler differencing in time is used.

If the projector onto incoming waves $\mathbf{\Pi}_U^+$ has rank $n_+ = n$, then $\mathbf{\Pi}_U^+ = \mathbf{I}$ and the absorbing boundary condition reduce to $\mathbf{U} = \mathbf{U}_{\text{ref}}$ (being $\mathbf{U}_{\text{ref}}$ a given value or $\mathbf{U}_0^n$ for ULSAR). This happens for instance in a supersonic inlet for gas dynamics or an inlet boundary for linear advection. In this case it is replaced the balance equation for the boundary node (the first equation in (34)) with the absorbing condition $\mathbf{U} = \mathbf{U}_{\text{ref}}$, keeping the balance between equations and unknowns.

Conversely, if the projector onto incoming waves $\mathbf{\Pi}_U^+$ has rank $n_+ = 0$, then $\mathbf{\Pi}_U^+ = \mathbf{0}$ and the absorbing boundary condition reduce to not imposing anything. This happens for instance in a supersonic outlet for gas dynamics or an outlet boundary for linear advection. In this case the absorbing condition $\mathbf{U} = \mathbf{U}_{\text{ref}}$ is discarded. Again, the number of equations and unknowns is maintained.

The case is more complicated when $0 < n_+ < n$. It cannot be added the absorbing condition (either (20), (29) or (33)), because the boundary balance equation cannot be discarded or maintained.

There are at least two strategies for imposing this non-linear boundary conditions. One possibility is to replace the boundary balance equation for the outgoing waves with a null first derivative condition. Then a discrete version can be generated with finite difference approximations. (This requires, however, a structured mesh at least near the boundary.) The other is to resort to the use of Lagrange multipliers or penalization techniques. One advantage of using Lagrange multipliers or penalization is that not only the boundary conditions coefficients can easily be changed for non-linear problems, but also *the number* of imposed boundary conditions. This is important for problems where the number of incoming characteristics cannot be easily determined *a priori*, or for problems where the flow regime is changing from subsonic to supersonic, or the flow reverts. In the rest of this section the second strategy will be described in detail.

In the base of the characteristic variables $\mathbf{V}$, (34) can be written as

$$\begin{aligned} &\frac{\mathbf{V}_0^{n+1}-\mathbf{V}_0^n}{\Delta t}+\mathbf{\Lambda}\frac{\mathbf{V}_1^{n+1}-\mathbf{V}_0^{n+1}}{h}=0,\\ &\frac{\mathbf{V}_k^{n+1}-\mathbf{V}_k^n}{\Delta t}+\mathbf{\Lambda}\frac{\mathbf{V}_{k+1}^{n+1}-\mathbf{V}_{k-1}^{n+1}}{2h}=0, \quad k \geqslant 1. \end{aligned} \tag{35}$$

For the linear absorbing boundary conditions (20) it should be imposed

$$\mathbf{\Pi}_V^+(\mathbf{V}_{\text{ref}})(\mathbf{V}_0 - \mathbf{V}_{\text{ref}}) = 0, \tag{36}$$

while discarding the equations corresponding to the incoming waves in the first rows of (35). Here $\mathbf{U}_{\text{ref}}/\mathbf{V}_{\text{ref}}$ is the state about which the linearization is done.

4.7.1. Using Lagrange multipliers

This can be done, via Lagrange multipliers in the following way

$$\begin{aligned}
&\mathbf{\Pi}_V^+(\mathbf{V}_{\text{ref}})(\mathbf{V}_0-\mathbf{V}_{\text{ref}})+\mathbf{\Pi}_V^-(\mathbf{V}_{\text{ref}})\mathbf{V}_{Im}=0,\\
&\frac{\mathbf{V}_0^{n+1}-\mathbf{V}_0^n}{\Delta t}+\mathbf{\Lambda}\,\frac{\mathbf{V}_1^{n+1}-\mathbf{V}_0^{n+1}}{h}+\mathbf{\Pi}_V^+(\mathbf{V}_{\text{ref}})\mathbf{V}_{Im}=0,\\
&\frac{\mathbf{V}_k^{n+1}-\mathbf{V}_k^n}{\Delta t}+\mathbf{\Lambda}\,\frac{\mathbf{V}_{k+1}^{n+1}-\mathbf{V}_{k-1}^{n+1}}{2h}=0,\quad k\geqslant 1,
\end{aligned}\tag{37}$$

where $\mathbf{V}_{Im}$ are the Lagrange multipliers for the imposition of the new conditions. On one hand, if j is an incoming wave ($\lambda_j \geqslant 0$), then the equation is of the form

$$\begin{aligned}
&v_{j0}-v_{\text{ref}0}=0,\\
&\frac{v_{j0}^{n+1}-v_{j0}^n}{\Delta t}+\lambda_j\frac{v_{j1}^{n+1}-v_{j0}^{n+1}}{h}+v_{j,Im}=0,\\
&\frac{v_{jk}^{n+1}-v_{jk}^n}{\Delta t}+\lambda_j\frac{v_{j,k+1}^{n+1}-v_{jk}^{n+1}}{2h}=0,\quad k\geqslant 1.
\end{aligned}\tag{38}$$

Note that, due to the $v_{j,Im}$ Lagrange multiplier, it can be solved for the v_{jk} values from the first and last rows, while the value of the multiplier $v_{j,Im}$ "*adjusts*" itself in order to satisfy the equations in the second row.

On the other hand, for the outgoing waves ($\lambda_j < 0$), the equations is

$$\begin{aligned}
&v_{j,Im}=0,\\
&\frac{v_{j0}^{n+1}-v_{j0}^n}{\Delta t}+\lambda_j\frac{v_{j1}^{n+1}-v_{j0}^{n+1}}{h}=0,\\
&\frac{v_{jk}^{n+1}-v_{jk}^n}{\Delta t}+\lambda_j\frac{v_{j,k+1}^{n+1}-v_{jk}^{n+1}}{2h}=0,\quad k\geqslant 1.
\end{aligned}\tag{39}$$

So that the solution coincides with the unmodified original FEM equation, and the Lagrange multiplier is $v_{j,Im}=0$.

Coming back to the $\mathbf{U}$ basis, it can be written

$$\begin{aligned}
&\mathbf{\Pi}_U^+(\mathbf{U}_{\text{ref}})(\mathbf{U}_0-\mathbf{U}_{\text{ref}})+\mathbf{\Pi}_U^-(\mathbf{U}_{\text{ref}})\mathbf{U}_{Im}=0,\\
&\mathbf{C}\frac{\mathbf{U}_0^{n+1}-\mathbf{U}_0^n}{\Delta t}+\mathbf{A}\,\frac{\mathbf{U}_1^{n+1}-\mathbf{U}_0^{n+1}}{h}+\mathbf{C}\mathbf{\Pi}_U^+(\mathbf{U}_{\text{ref}})\mathbf{U}_{Im}=0,\\
&\mathbf{C}\frac{\mathbf{U}_k^{n+1}-\mathbf{U}_k^n}{\Delta t}+\mathbf{A}\frac{\mathbf{U}_{k+1}^{n+1}-\mathbf{U}_{k-1}^{n+1}}{2h}=0,\quad k\geqslant 1.
\end{aligned}\tag{40}$$

4.7.2. Using penalization

The corresponding formulas for penalization can be obtained by adding a diagonal term scaled by a small regularization parameter ϵ to the first equation in (40)

$$\begin{aligned}
&-\epsilon\mathbf{U}_{Im}+\mathbf{\Pi}_U^+(\mathbf{U}_0-\mathbf{U}_{\text{ref}})+\mathbf{\Pi}_U^-\mathbf{U}_{Im}=0,\\
&\mathbf{C}\frac{\mathbf{U}_0^{n+1}-\mathbf{U}_0^n}{\Delta t}+\mathbf{A}\,\frac{\mathbf{U}_1^{n+1}-\mathbf{U}_0^{n+1}}{h}+\mathbf{\Pi}_U^+\mathbf{U}_{Im}=0,
\end{aligned}\tag{41}$$

where, for the moment, the dependence of the projectors on $\mathbf{U}_{\text{ref}}$ is dropped. Eliminating $\mathbf{U}_{Im}$ from the first and second rows it is obtained

$$\begin{aligned}
&\mathbf{C}\frac{\mathbf{U}_0^{n+1}-\mathbf{U}_0^n}{\Delta t}+\mathbf{A}\,\frac{\mathbf{U}_1^{n+1}-\mathbf{U}_0^{n+1}}{h}\\
&\quad+\mathbf{\Pi}_U^+(\mathbf{\Pi}_U^-+\epsilon\mathbf{I})^{-1}\mathbf{\Pi}_U^+(\mathbf{U}_0-\mathbf{U}_{\text{ref}})=0.
\end{aligned}\tag{42}$$

Now, using projection algebra it can be shown that

$$(\mathbf{\Pi}_U^-+\epsilon\mathbf{I})^{-1}=\left(\frac{1}{\epsilon}\mathbf{\Pi}_U^+ + \frac{1}{1+\epsilon}\mathbf{\Pi}_U^-\right),\tag{43}$$

so that the last term in (42) reduces to $\mathbf{\Pi}_U^+(\mathbf{U}_0-\mathbf{U}_{\text{ref}})$ and the whole equation is

$$\mathbf{C}\frac{\mathbf{U}_0^{n+1}-\mathbf{U}_0^n}{\Delta t}+\mathbf{A}\,\frac{\mathbf{U}_1^{n+1}-\mathbf{U}_0^{n+1}}{h}+\frac{1}{\epsilon}\mathbf{C}\mathbf{\Pi}_U^+(\mathbf{U}_0-\mathbf{U}_{\text{ref}})=0.\tag{44}$$

Here $1/\epsilon$ can be thought as a large penalization factor.

4.8. Viscous compressible subsonic flow over a parabolic bump

In order to evaluate the absorption of waves impinging at fictitious boundaries a 2D test consisting of a compressible subsonic flow over a parabolic bump at $Ma_{\text{ref}}=0.5$ is considered (see Fig. 7). The idea is to assess how the length from bump trailing edge to the fictitious outflow (L_{out}) affects the predicted forces and their time evolution. Two set of simulation were carried out. One set considering non-absorbent boundary conditions where variables are imposed as specified in Fig. 7. At inlet wall the imposed conditions are $\rho=\rho_{\text{ref}}=1$, $u=u_{\text{ref}}=Ma_{\text{ref}}\sqrt{\gamma p_{\text{ref}}/\rho_{\text{ref}}}=0.5$ and $v=0$. At the outflow boundary pressure is imposed, i.e., $p=p_{\text{ref}}=1/\gamma$, where $\gamma=1.4$. The second set of simulations is considering ULSAR non-reflecting conditions at channel inlet and outlet. Initial state for both set of problems is $\mathbf{U}=(\rho_{\text{ref}},u_{\text{ref}},0,p_{\text{ref}})$. Parameters in Fig. 7 are: $L_{\text{in}}=1.4$, $L_{\text{bump}}=2$, $h_{\text{bump}}=0.1$ and $L_{\text{out}}=1,2,4,8$. The values are dimensionless by selecting L, ρ_{ref} and u_{ref} as reference values for length, density and velocity.

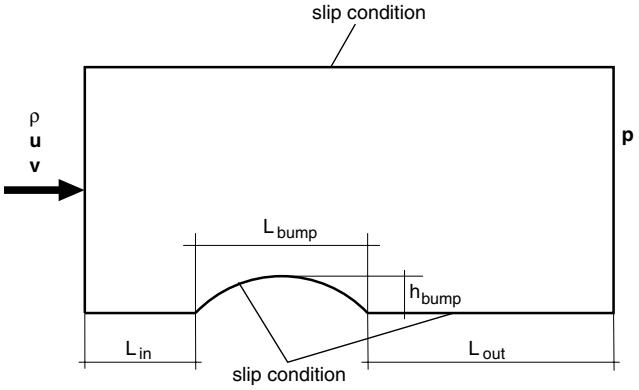

Fig. 7. Problem geometry.

Figs. 8 and 9 show how ULSAR conditions produce the wave absorption at fictitious boundaries.

5. Dynamically varying boundary conditions

5.1. Varying boundary conditions in external aerodynamics

During a flow computation the number of incoming characteristics n_+ may change. This can occur due to a flow regime changing (i.e., from subsonic to supersonic) or due to a flow sense changing (flow reversal). A typical case is the external flow around an aerodynamic body as shown in Fig. 10. Consider first a steady subsonic flow. The flow is normally subsonic at the whole infinite boundary, even if some supersonic pockets can develop at transonic speeds. Then the only two possible regimes are subsonic inlet ($n_+ = n_d + 1$, n_d is the spatial dimension) and subsonic outlet ($n_+ = 1$). By looking at the projection of the unperturbed flow velocity $\mathbf{u}_\infty$ onto the local normal $\hat{\mathbf{n}}$ it can be settled whether the boundary is inlet or outlet. For the steady supersonic case the situation is very different. A bow shock develops in front of the body and forms a subsonic region which propagates downstream. Far downstream the envelope of the subsonic region approaches a cone with an aperture angle equal to the Mach angle for the undisturbed flow. Now, the inlet region is supersonic and the outlet one is both, subsonic and supersonic. The point where the flow at outlet changes from subsonic to supersonic may be estimated from the Mach angle, but it may be very inaccurate if the boundary is close to the body. Having a boundary condition that can automatically adapt itself to the whole possibilities can be of great help in such a case. Now, consider the unsteady case, for instance a body slowly accelerated from subsonic to supersonic speeds. The inlet part will change at some point from subsonic to supersonic. At outlet, some parts will change also from subsonic to supersonic, and the separation between both parts will change its position, following approximately the instantaneous Mach angle.

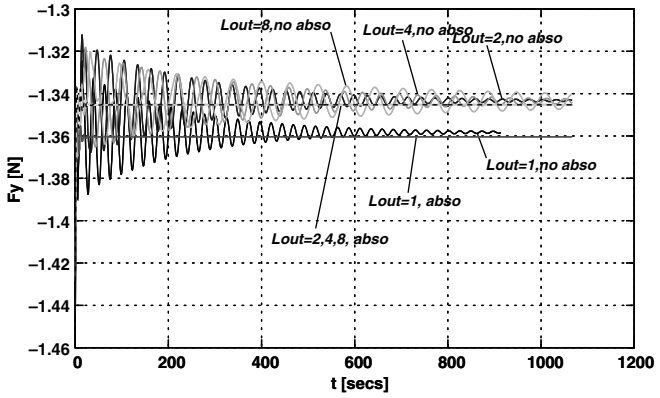

Fig. 8. y-Force.

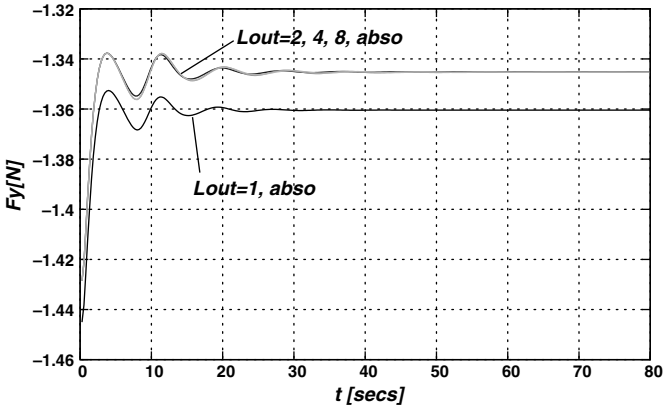

Fig. 9. y-Force evolution for absorbent conditions.

5.2. Aerodynamics of falling objects

An interesting case is the aerodynamics of a falling body [18–22]. Consider, for simplicity, a two-dimensional case of an homogeneous ellipse in free fall (Fig. 11). As the body accelerates, the pitching moments tend to increase the angle of attack until it stalls (A). Then, the body starts to fall towards its other end, and accelerates while its main axis aligns with gravity (B). As the body accelerates the pitching moment grows until it eventually stalls again (C). The pattern is repeated during the downfall. This kind of falling

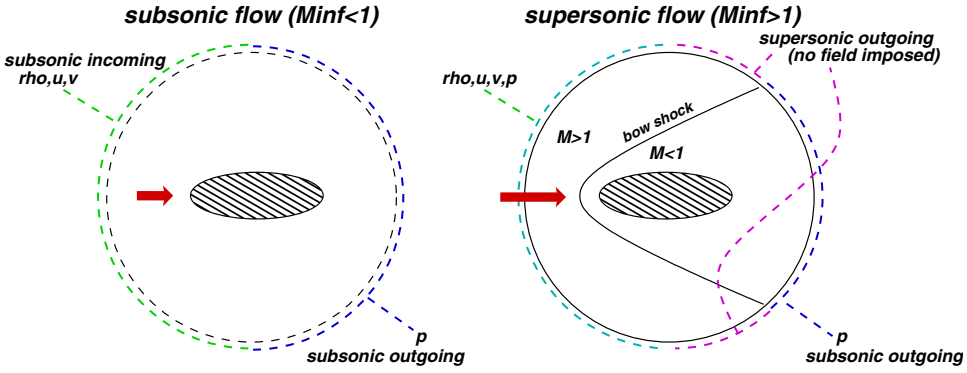

Fig. 10. Number of incoming/outgoing characteristics changing on an accelerating body.

mechanism is typical of slender bodies with relatively small moment of inertia like a sheet of paper and is called *'flutter'*. However, depending on several parameters, but mainly depending on the moment of inertia of the body, if it has a large angular moment at (B), it may happen that it rolls on itself, keeping always the same sense of rotation. This kind of falling mechanism is called *'tumble'* and is a typical pattern for thicker and massive objects. For massive objects (like a ballistic projectile, for instance) tumbling may convert a large amount of potential energy in the form of rotation, causing the object to rotate at very large speeds. As the body falls it accelerates and can reach supersonic speeds. This depends on the density of the body relative to the surrounding atmosphere, its dimensions and shape. As the weight of the body goes with $\propto L^3$, being L the characteristic length, while the drag force goes with $\propto L^2$, larger bodies tend to reach larger limit speeds and eventually reach supersonic regime.

The falling of a body can be modeled in several ways. In order to avoid the use of deforming meshes, a fixed mesh attached to the body can be used. Then, it is possible to perform the computation in a *non-inertial* frame moving with the body or using an inertial frame with a *moving but not deforming* mesh. In the first case "*inertial forces*" (Coriolis, centrifugal) must be added, while in the second case convective terms must take into account the mesh velocity as in the "*Arbitrary Lagrangian Eulerian (ALE)*" formulation. In this example the second strategy was used.

The computation of the flow is linked to the dynamics of the falling object. The strategy is a typically staggered fluid/solid interaction process [23–26]. First, a standard predictor is applied in order to obtain a guess for the position of the body at t^{n+1}. Then, the fluid solver updates the state of the fluid from t^n to t^{n+1} including the ALE terms. Then, with the state of the fluid at t^{n+1} the forces exerted by the fluid on the body are computed and the equations for the rigid motion of the body are solved (six degrees of freedom, accounting for two linear position and velocities, rotation angle and its derivative).

Coming back to the boundary conditions issue, added the fact that the body can accelerate and decelerate, and going back and forth from subsonic to supersonic speeds, it must be taken into account that the angle from which the unperturbed flow impinges on the body varies with time. So, as the body can rotate arbitrarily, the flow can impinge from any direction relative to the boundary.

5.2.1. Numerical example. Ellipse falling at supersonic speed

As an example consider the fall of an ellipse with the following physical data

- $a = 1$, $b = 0.6$ (major and minor semi-axes, eccentricity $e = \sqrt{1 - b^2/a^2} = 0.8$),
- $m = 2.885$ (mass of body),
- $g = 2.5$ (gravity),
- $r = 1$ (radius of inertia),
- c.m. $= (-0.15, 0.0)$ (center of mass),
- $\rho_a = 1$ (atmosphere density),
- $p = 1$ (atmosphere pressure),
- $\gamma = 1.4$ (gas adiabatic index $\gamma = C_p/C_v$),
- $R_{\text{ext}} = 10$ (radius of the fictitious boundary),
- $\mathbf{u}_{\text{ini}} = [0, 0, 1.39, 0, 1.3, 0]$ (ellipse initial state (position and velocity) $[x, y, \theta, u, v, \dot{\theta}]$).

These values are dimensionless by selecting a, ρ_a and c_0 as reference values for length, density and velocity, so that the non-dimensional quantities are $\rho'_a = 1$, $p' = 1/\gamma$, $u' = 0.5$ (in the following the prime indicating non-dimensional quantities is dropped). A coarse estimation of the

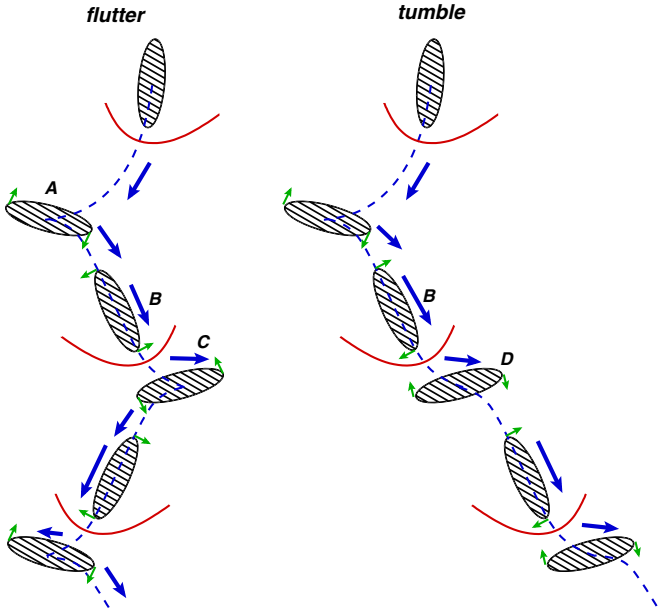

Fig. 11. Falling ellipse.

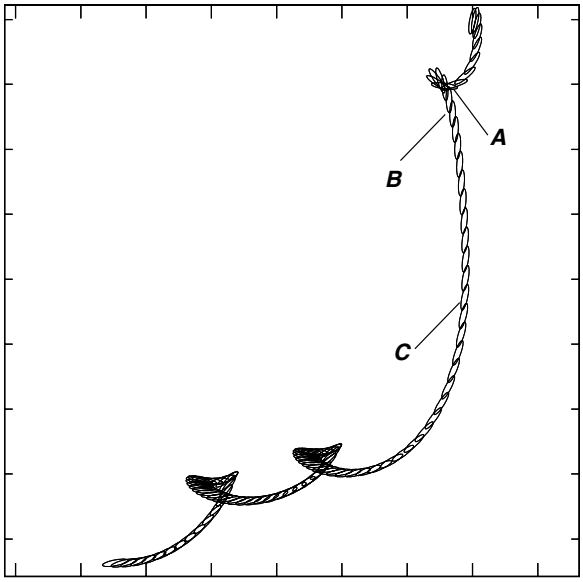

Fig. 12. Computed trajectory of falling ellipse.

limit speed v can be obtained by balancing the vertical forces on the body, i.e., the drag on the body (F_{aero}), the weight and the hydrostatic flotation

$$F_{\text{aero}} + W + F_{\text{float}} = \frac{1}{2} C_D \rho_a v^2 A - \rho_s g V + \rho_a g V = 0, \qquad (45)$$

where $V = \pi ab$ is the volume of the body (the area in 2D) and $A = 2b$ the area of the section facing the fluid (length in 2D). $C_D = 0.2$ is an estimation for the drag coefficient of the body and $\rho_s = m/V$, ρ_a the densities of solid and atmosphere respectively. For the data above, this estimation gives a limit speed of $v = 4.6$ approximately. The speed of sound of the atmosphere is $c = \sqrt{\gamma p/\rho_a} = 1.18$, so that it is expected that the body will reach supersonic speeds. Of course, if the body does reach supersonic speed, then the drag coefficient will be higher and probably the average speed will be lower than that one estimated above.

The initial conditions are the ellipse starting at velocity $(0, -1.39)$, null angular velocity, and an angle of its major axis of 80° with respect with the vertical. The fluid is initially at rest. The computed trajectory until $t = 50$ time units is shown in Fig. 12. The computed trajectory is shown

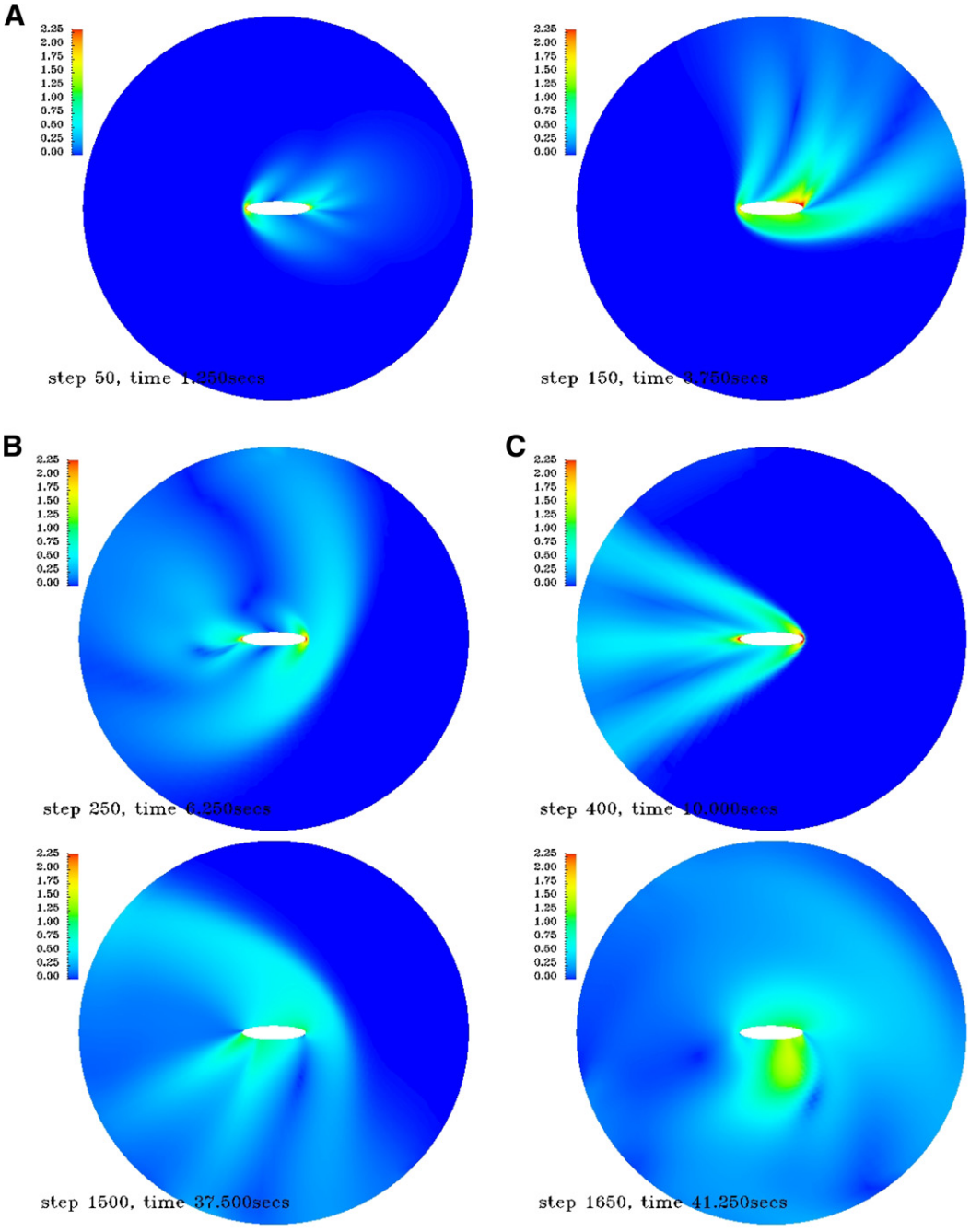

Fig. 13. Ellipse falling at supersonic speeds. Colormaps of Mach number. Top left: station A ($t = 3.75$), top right: station B ($t = 6.25$), bottom: station C ($t = 10$). Stations in the trajectory refer to Fig. 12. Results are shown in a non-inertial frame attached to the ellipse.

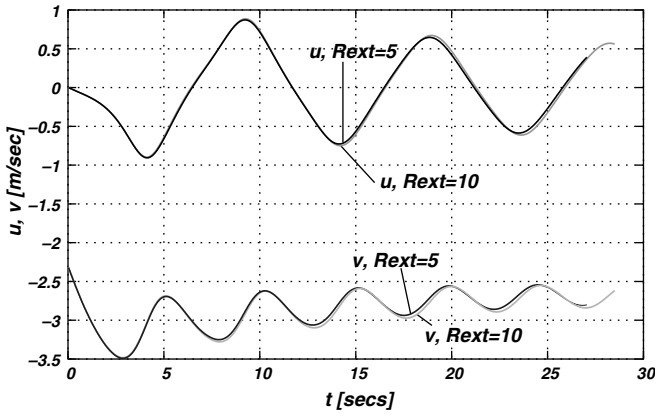

Fig. 14. Ellipse velocities for different external radius.

in a reference system falling at velocity $v = (-0.5, 0.5)$ (this is done in order to reduce the horizontal and vertical span of the plot). Fig. 13 show colormaps of Mach number at six instants, in the inertial frame fixed to the fluid. The instants are marked as A, B, C and identified in the trajectory. Note that as the ellipse rotates, each part of the boundary experiments all kind of regimes and the absorbing boundary condition copes with all of them. Note also that the artificial boundary is located very near to the body, the radius of the external circle is 3.25 times the major semi-axis of the ellipse (in the case simulated with the minor external radius, i.e., $R_{\text{ext}} = 5$).

In Fig. 14, the velocities of the ellipse are shown in order to evaluate the absorption of ULSAR conditions when waves reach boundaries as the ellipse falls and tumble/flutter when the fictitious boundary (exterior circle) is located at $R_{\text{ext}} = 5$ m and $R_{\text{ext}} = 10$ m and the size of finite elements remain constant.

6. Conclusions

Absorbing boundary conditions reduce computational cost by allowing to put the artificial exterior boundary closer to the region of interest. Extension to the non-linear cases can be done either by using Riemann invariants or by using the state at the previous time step as reference state for a linearized boundary condition. In complex simulations, the number of incoming characteristic waves may vary during the computation or may not be known a priori. In those cases, absorbing boundary conditions can be imposed with the help of Lagrange multipliers or penalization techniques.

Acknowledgements

This work has received financial support from Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET, Argentina, grants PIP 0198/98, PIP 02552/00, PIP 5271/05), Universidad Nacional del Litoral (UNL, Argentina, grants CAI+D 2000/43) and Agencia Nacional de Promoción Científica y Tecnológica (ANPCyT, Argentina, grants PICT 6973/99, PID-74/99, PICT Lambda 12-14573/2003, PME 209/2003). Extensive use of freely distributed software as GNU/Linux OS, MPI, PETSc, GCC compilers, Octave, Open-DX were made among many others.

References

[1] D. Givoli, J.B. Keller, Non-reflecting boundary conditions for elastic waves, Wave Motion 12 (1990) 261–279.
[2] D. Givoli, J.B. Keller, A finite element method for large domains, Comput. Methods Appl. Mech. Engrg. 76 (1989) 41–66.
[3] J. Broeze, J.E. Romate, Absorbing boundary conditions for free surface wave simulations with a panel method, J. Comput. Phys. 99 (1992) 146.
[4] I. Harari, T.J.R. Hughes, Galerkin least-squares finite element methods for the reduced wave equation with non-reflecting boundary conditions in unbounded domains, Comput. Methods Appl. Mech. Engrg. 98 (1992) 411–454.
[5] M. Storti, J. D'Elía, S. Idelsohn, Algebraic discrete non-local (dnl) absorbing boundary condition for the ship wave resistance problem, J. Comput. Phys. 146 (1997) 570–602.
[6] T. Hagstrom, Boundary conditions at outflow for a problem with transport and diffusion, J. Comput. Phys. 69 (1987) 69–80.
[7] M. Storti, J. D'Elia, R. Bonet, N. Nigro, S. Idelsohn, The DNL absorbing boundary condition. Applications to wave problems, Comput. Methods Appl. Mech. Engrg. 182 (3–4) (2000) 483–498.
[8] S.V. Tsynkov, Numerical solution of problems on unbounded domains. A review, Appl. Numer. Math. 27 (1998) 465–532.
[9] T.E. Tezduyar, T.J.R. Hughes, Finite element formulations for convection dominated flows with particular emphasis on the compressible euler equations, in: Proceedings of AIAA 21st Aerospace Sciences Meeting, Reno, Nevada, AIAA Paper 83-0125, 1983.
[10] T.J.R. Hughes, T.E. Tezduyar, Finite element methods for first-order hyperbolic systems with particular emphasis on the compressible euler equations, Comput. Methods Appl. Mech. Engrg. 45 (1984) 217–284.
[11] T.E. Tezduyar, Finite Element Methods for Fluid Dynamics with Moving Boundaries and Interfaces, Encyclopedia of Computational Mechanics, vol. 3, John Wiley and Sons, 2004.
[12] T.E. Tezduyar, Finite elements in fluids: stabilized formulations and moving boundaries and interfaces, Comput. Fluids 36 (2007) 191–206.
[13] S.K. Aliabadi, S.E. Ray, T.E. Tezduyar, SUPG finite element computation of viscous compressible flows based on the conservation and entropy variables formulations, Comput. Mech. 11 (1993) 300–312.
[14] G.J. Le Beau, S.E. Ray, S.K. Aliabadi, T.E. Tezduyar, Supg finite element computation of compressible flows with the entropy and conservation variables formulations, Comput. Methods Appl. Mech. Engrg. 104 (1993) 397–422.
[15] T.E. Tezduyar, M. Senga, Stabilization and shock-capturing parameters in supg formulation of compressible flows, Comput. Methods Appl. Mech. Engrg. 195 (2006) 1621–1632.
[16] C. Baumann, M. Storti, S. Idelsohn, Improving the convergence rate of the Petrov–Galerkin techniques for the solution of transonic and supersonic flows, Int. J. Numer. Methods Engrg. 34 (1992) 543–568.
[17] B.F. Sanders, High-resolution and non-oscillatory solution of the st. Venant equations in non-rectangular and non-prismatic channels, J. Hydraul. Res. 39 (3) (2001) 321–330.
[18] S. Field, M. Klaus, M. Moore, Franco Nori, Instabilities and chaos in falling objects, Nature (388) (1997) 252–254.
[19] A. Belmonte, Flutter and Tumble in Fluids, Physics World, 1999.
[20] J.Y. Huang, Trajectory of a moving curveball in viscid flow, in: Proceedings of the Third International Conference: Dynamical Systems and Differential Equations, 2000, pp. 191–198.

[21] J.Y. Huang, Moving Coordinates Methods and Applications to the Oscillations of a Falling Slender Body, Moving Boundaries VI, WIT Press, 2001, pp. 73–82.
[22] J.Y. Huang, Aerodynamics of a Moving Curveball in Newtonian Flow, Advances in Fluid Mechanics IV, WIT Press, 2002, pp. 597–608.
[23] R. Piperno, C. Farhat, Partitioned procedures for the transient solution of coupled aeroelastic problems. Part II: energy transfer analysis and three-dimensional applications, Comput. Methods Appl. Mech. Engrg. 190 (2001) 3147–3170.
[24] J. Cebral, Loose Coupling Algorithms for fluid structure interaction, Ph.D. Thesis, Institute for Computational Sciences and Informatics, George Mason University, 1996.
[25] R. Löhner, C. Yang, J. Cebral, J. Baum, H. Luo, D. Pelessone, C. Charman, Fluid–structure interaction using a loose coupling algorithm and adaptive unstructured grids, AIAA paper AIAA-98-2419, 1998.
[26] R. Lohner, J.R. Cebral, Fluid–structure interaction in industry: issues and outlook, in: Proceedings of the World User Association in Applied Computational Fluid Dynamics, 3rd World Conference in Applied Computational Fluid Dynamics, Germany, May 19–23, 1996.