DYNAMIC BOUNDARY CONDITIONS IN CFD

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

Centro Internacional de Métodos
Computacionales en Ingeniería (CIMEC),
INTEC(CONICET-UNL), Gümes 3450, (S3000GLN) Santa Fe, Argentina
mailto:mstorti@intec.unl.edu.ar,
http://www.cimec.org.ar/mstorti.

Key Words: finite elements, computational fluid dynamics, absorbing boundary conditions

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 input/output 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 normal 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 still harder when the number of incoming characteristics varies during the computation, and to correctly treat these cases poses both mathematical and practical problems. One example considered here is 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.*

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 taken 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 one has to introduce an artificial boundary distant as far as possible from the region of interest. The simplest choice is to impose a boundary condition assuming that the flow far from the region of interest is undisturbed. However, one has the freedom of choosing the boundary condition 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 (like Dirichlet or Neumann) condition may lead to a lack of convergence of the problem, because these conditions are completely reflecting and 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 condition 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 condition exist and are popular for the simpler linear operators, like potential flow problems, frequency domain analysis of wave problems like the Helmholtz equations for acoustics or the Maxwell equations.[1,2,3,4,5,6]

On the other hand the discrete operator for local absorbing boundary conditions is usually sparse but has a lower order accuracy and, in general, it is needed to bring the artificial boundary condition to infinity while refining 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.[7]

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 conditions to none.

In more complex problems it can go through the whole possible combinations of regimes: 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 fall of tree leaves. During the fall, the speed of the object varies periodically, accelerating when the angle of attach is smaller and the body experiences less 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 system of coordinates 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 exterior wind. Assume that the concentration of solid particles or contaminant is so low that its influence on the fluid is negligible so that we can solve first the movement of the fluid inside the building and then a 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 direction.

Changing 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 changing 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 papers is to explain how these variable boundary conditions may be implemented through Lagrange multipliers or penalization techniques, to discuss numerical aspects relative 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

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

Here $\mathbf{U} \in \mathbb{R}^n$ is the state vector, t is time, $\mathcal{F}_{c,j}, \mathcal{F}_{d,j}$ 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 the spatial coordinates.

The notation is standard, except perhaps for the "*generic enthalpy function*" $\mathcal{H}(\mathbf{U})$. The inclusion of the enthalpy function allows the inclusion of conservative equations in term 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

For instance, the heat advection-diffusion equation in terms of temperature can be put in this form through the definitions

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

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

2.2 Gas dynamics equations

The compressible flow, gas dynamics equations, can be put in conservative form with the following definitions

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

Note that 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 $\mathbf{U}_p = [\rho, \mathbf{u}, p]^T$. 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 we introduce the *"enthalpy function"* $\mathcal{H}(\mathbf{U}_p) = \mathbf{U}_c(\mathbf{U}_p)$.

2.3 Shallow water equations

Shallow water equations describes the open flow of fluids over regions whose characteristic dimensions are much larger than the depth.

$$\begin{aligned} \mathbf{U}_p &= [h, \mathbf{u}]^T, \\ \mathbf{U} = \mathbf{U}_c &= [h, h\mathbf{u}]^T, \\ \mathcal{H}(\mathbf{U}) &= \mathbf{U}, \\ \mathcal{F}_{c,j} n_j &= \begin{bmatrix} h(\mathbf{u}\cdot\hat{\mathbf{n}}) \\ h(\mathbf{u}\cdot\hat{\mathbf{n}})\,\mathbf{u} + \frac{1}{2} g h^2\,\mathbf{I} \end{bmatrix}. \end{aligned} \tag{4}$$

where h is the fluid depth, $\mathbf{u}$ the velocity vector, $\mathbf{U}_p$, $\mathbf{U}_c$ the primitive and conservative variables, g the gravity acceleration. We assume that the height of the bottom with respect to a fixed datum is constant. If this is not so, additional terms must be included in the source term $\mathbf{G}$, but this is irrelevant for the absorbing boundary condition issue.

2.4 Channel flow

Flow in a channel can be cast in advective form as follows

$$\begin{aligned} \mathbf{U}_p &= [h, u]^T, \\ \mathbf{U} = \mathbf{U}_c &= [A, Q]^T, \\ \mathcal{H}(\mathbf{U}) &= \mathbf{U}, \\ \mathcal{F} &= \begin{bmatrix} Q \\ Q^2/A + F \end{bmatrix}. \end{aligned} \tag{5}$$

where h and u are water depth and velocity (as in the shallow water equations). $A(h)$ is the section of the channel occupied by water for a given water height h and then defines the geometry of the channel. For instance

- Rectangular channels: $A(h) = wh$, w=width.
- Triangular channels: $A(h) = 2h^2 \tan\theta/2$; with θ=angle opening.
- Circular channel:

$$\begin{aligned} A(h) &= \int_{h'=0}^{h} \sqrt{2Rh - h^2}\,\mathrm{d}h \\ &= \theta R^2 - w(h)(R-h)/2 \end{aligned} \tag{6}$$

where R is the radius of the channel, $w(h) = 2\sqrt{2Rh - h^2}$ is the waterline for a given water height and $\theta = \mathrm{atan}[w/(2(R-h))]$ is the angular aperture.

$Q = Au$ is the water flow rate. $F(h)$ is a function defined by

$$F(h) = \int_{h'=0}^{h} A(h')\,\mathrm{d}h'. \tag{7}$$

Again, for the sake of simplicity, we restrict to the case of constant channel section an channel depth. For more general situations, other terms that can be included in the source and diffusive terms are present, not needed for the discussion of absorbing boundary conditions. For rectangular channels the equations reduce to those for one dimensional shallow water equations.

Channel flow is very interesting since it is in fact a family of different 1D hyperbolic systems depending on the area function $A(h)$.

3 VARIATIONAL FORMULATION

The weighted variational form for this kind of systems is to find $\mathbf{U}^h \in \mathcal{S}^h$ such that, for every $\mathbf{W}^h \in \mathcal{V}^h$,

$$\begin{aligned}\int_\Omega \mathbf{W}^h \cdot \left(\frac{\partial \mathcal{H}(\mathbf{U}^h)}{\partial t} + \frac{\partial \mathcal{F}_{c,j}}{\partial x_j} - \mathbf{G}\right)\mathrm{d}\Omega + \int_\Omega \frac{\partial \mathbf{W}^h}{\partial x_j}\mathcal{F}_{d,j}\,\mathrm{d}\Omega - \int_{\Gamma_h} \mathbf{W}^h \cdot \mathbf{H}^h\,\mathrm{d}\Gamma \\ + \sum_{e=1}^{n_{elem}} \int_\Omega \tau_e\, \mathbf{A}_k^T \frac{\partial \mathbf{W}^h}{\partial x_k} \cdot \left(\frac{\partial \mathcal{H}(\mathbf{U})}{\partial t} + \frac{\partial \mathcal{F}_{c,j}(\mathbf{U})}{\partial x_j} - \frac{\partial \mathcal{F}_{d,j}(\mathbf{U}, \nabla\mathbf{U})}{\partial x_j} - \mathbf{G}\right)\mathrm{d}\Omega = 0\end{aligned} \tag{8}$$

where

$$\begin{aligned}\mathcal{S}^h &= \left\{\mathbf{U}^h | \mathbf{U}^h \in [\mathbf{H}^{1h}(\Omega)]^m,\, \mathbf{U}^h\big|_{\Omega^e} \in [P^1(\Omega^e)]^m,\ \ \mathbf{U}^h = \mathbf{g} \text{ at } \Gamma_g\right\} \\ \mathcal{V}^h &= \left\{\mathbf{W}^h | \mathbf{W}^h \in [\mathbf{H}^{1h}(\Omega)]^m,\, \mathbf{U}^h\big|_{\Omega^e} \in [P^1(\Omega^e)]^m,\ \ \mathbf{U}^h = 0 \text{ at } \Gamma_g\right\}\end{aligned} \tag{9}$$

are the space of interpolation and weight function respectively, τ_e are stabilization parameters (a.k.a. *"intrinsic times"*), Γ_g is the Dirichlet part of the boundary, where $\mathbf{U} = \mathbf{g}$ is imposed, and Γ_h is the Neumann part of the boundary where $\mathcal{F}_{d,j} n_j = \mathbf{H}$ is imposed.

4 ABSORBING BOUNDARY CONDITIONS

For steady simulations using time-marching algorithms, it can be shown that the error towards the steady state propagates as waves, so that absorbing boundary conditions help in eliminating 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.[8]

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 exterior region with the internal region.

In this paper we will concentrate in the use of absorbing boundary conditions 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. $\mathcal{F}_{d,j} \equiv 0$

$$\frac{\partial \mathcal{H}(\mathbf{U})}{\partial t} + \frac{\partial \mathcal{F}_{c,x}(\mathbf{U})}{\partial x} = 0, \text{ in } [0, L]. \tag{10}$$

If the system is *"linear"*, i.e. $\mathcal{F}_{c,x}(\mathbf{U}) = \mathbf{A}\mathbf{U}$, $\mathcal{H}(\mathbf{U}) = \mathbf{C}\mathbf{U}$ then we obtain a first order linear system

$$\mathbf{C}\,\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}\,\frac{\partial \mathbf{U}}{\partial x} = 0. \tag{11}$$

The system is *"hyperbolic"* if $\mathbf{C}$ is inversible, $\mathbf{C}^{-1}\mathbf{A}$ is diagonalizable and has real eigenvalues. If this is so we can make the following eigenvalue decomposition for $\mathbf{C}^{-1}\mathbf{A}$

$$\mathbf{C}^{-1}\mathbf{A} = \mathbf{S}\mathbf{\Lambda}\mathbf{S}^{-1}. \tag{12}$$

where $\mathbf{S}$ is real and inversible and $\mathbf{\Lambda}$ is real and diagonal. If we define new variables $\mathbf{V} = \mathbf{S}^{-1}\mathbf{U}$, then (11) becomes

$$\frac{\partial \mathbf{V}}{\partial t} + \mathbf{\Lambda}\frac{\partial \mathbf{V}}{\partial x} = 0. \tag{13}$$

Now, each equation is a linear scalar advection equation

$$\frac{\partial v_k}{\partial t} + \lambda_k \frac{\partial v_k}{\partial x} = 0, \quad \text{(no summation over } k\text{)}. \tag{14}$$

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

4.2 Linear 1D absorbing boundary conditions

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

$$\begin{aligned} &\text{if } \lambda_k > 0\text{: } v_k(0) = \bar{v}_{k0}; \quad \text{no boundary condition at } x = L \\ &\text{if } \lambda_k < 0\text{: } v_k(L) = \bar{v}_{kL}; \quad \text{no boundary condition at } x = 0 \end{aligned} \tag{15}$$

This can be put in compact form as

$$\begin{aligned} \mathbf{\Pi}_V^+(\mathbf{V} - \bar{\mathbf{V}}_0) = 0; \qquad &\text{at } x = 0 \\ \mathbf{\Pi}_V^-(\mathbf{V} - \bar{\mathbf{V}}_L) = 0; \qquad &\text{at } x = L \end{aligned} \tag{16}$$

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

$$\mathbf{\Pi}^{+}_{V,jk} = \begin{cases} 1; & \text{if } j = k \text{ and } \lambda_k > 0 \\ 0; & \text{otherwise,} \end{cases} \tag{17}$$
$$\mathbf{\Pi}^{+} + \mathbf{\Pi}^{-} = \mathbf{I}.$$

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, we have

$$\mathbf{\Pi}_V^{-}\,\mathbf{S}^{-1}(\mathbf{U} - \bar{\mathbf{U}}_L) = 0 \tag{18}$$

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

$$\mathbf{\Pi}_U^{\pm}\,(\mathbf{U} - \bar{\mathbf{U}}_{0,L}) = 0, \quad \text{at } x = 0, L, \tag{19}$$

where

$$\mathbf{\Pi}_U^{\pm} = \mathbf{S}\,\mathbf{\Pi}_V^{\pm}\,\mathbf{S}^{-1}, \tag{20}$$

are the projection matrices in the $\mathbf{U}$ basis. These conditions are completely absorbing for 1D linear advection (11).

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

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

$$\mathbf{U}(x, t = 0) = \mathbf{U}_{\text{ref}} + \Delta\mathbf{U}\,\mathrm{e}^{(x-x_0)/\sigma^2}, \tag{21}$$

where $\rho_{\text{ref}} = 1$, $u_{\text{ref}} = 0.5$, $p_{\text{ref}} = 0.714$, ($\text{Ma}_{\text{ref}} = 0.5$) $\delta\rho = \delta p = 0$, $\delta u = 0.1$, $\mathrm{R} = 1$, $x_0 = 0.8$, $\sigma = 0.3$. The evolution of this perturbation is simulated using $N = 50$ equispaced finite elements ($h = L/N = 0.08$) with SUPG stabilization, Crank-Nicholson temporal scheme with $\Delta t = 0.05$ (CFL number ≈ 0.84). As the flow is subsonic we have to impose two conditions at inlet and one at outlet. We will compare the results using standard and absorbing boundary conditions at outlet ($x = L$), while imposing non-absorbing $\rho = \rho_{\text{ref}}$ and $u = u_{\text{ref}}$ at inlet ($x = 0$). In figure 1 we see the evolution in time (in the form of an elevation view) of the velocity when using the condition $p = p_{\text{ref}}$ at outlet, while in figure 3 we see the results when using first order linear absorbing boundary conditions based on the unperturbed state. We see that without absorbing boundary condition the perturbation reflects at both boundaries. Even after $t = 40$

a significant amount of perturbation is still in the domain. At this point the perturbation has reflected four times at the boundaries. On the other hand, 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.

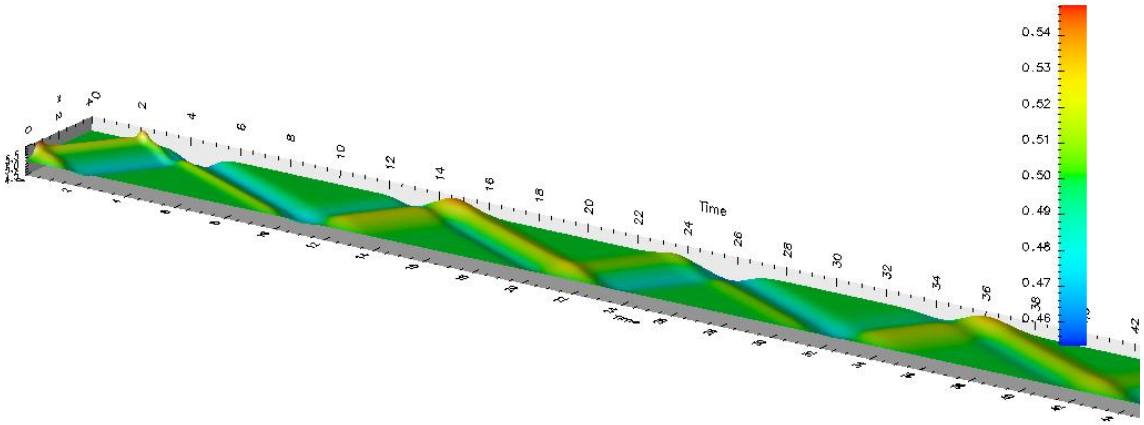

Figure 1: Temporal evolution of axial velocity in 1D gas dynamics problem *without* absorbing boundary condition at outlet.

4.3 Multidimensional problems

For multidimensional problems we can make a simplified 1D analysis in the direction normal to the local boundary and results in that the flux Jacobian $\mathbf{A}$ in (12) must be replaced with the projection of the flux Jacobians onto the exterior normal $\hat{\mathbf{n}}$, as follows

$$
\begin{aligned}
&\mathbf{\Pi}_n^-(\mathbf{U}-\bar{\mathbf{U}}) = 0,\\
&\mathbf{\Pi}_n^- = \mathbf{S}_n\,\mathbf{\Pi}_{Vn}^-\,\mathbf{S}_n^{-1},\\
&(\mathbf{\Pi}_{Vn}^-)_{jk} = \begin{cases} 1; & \text{if } j=k \text{ and } \lambda_j < 0,\\ 0; & \text{otherwise.}\end{cases}\\
&\mathbf{C}^{-1}\mathbf{A}_n = \mathbf{S}_n\mathbf{\Lambda}_n\mathbf{S}_n^{-1}, \quad (\mathbf{\Lambda}_n \text{ diagonal}),\\
&\mathbf{A}_n = \mathbf{A}_l n_l.
\end{aligned}
\tag{22}
$$

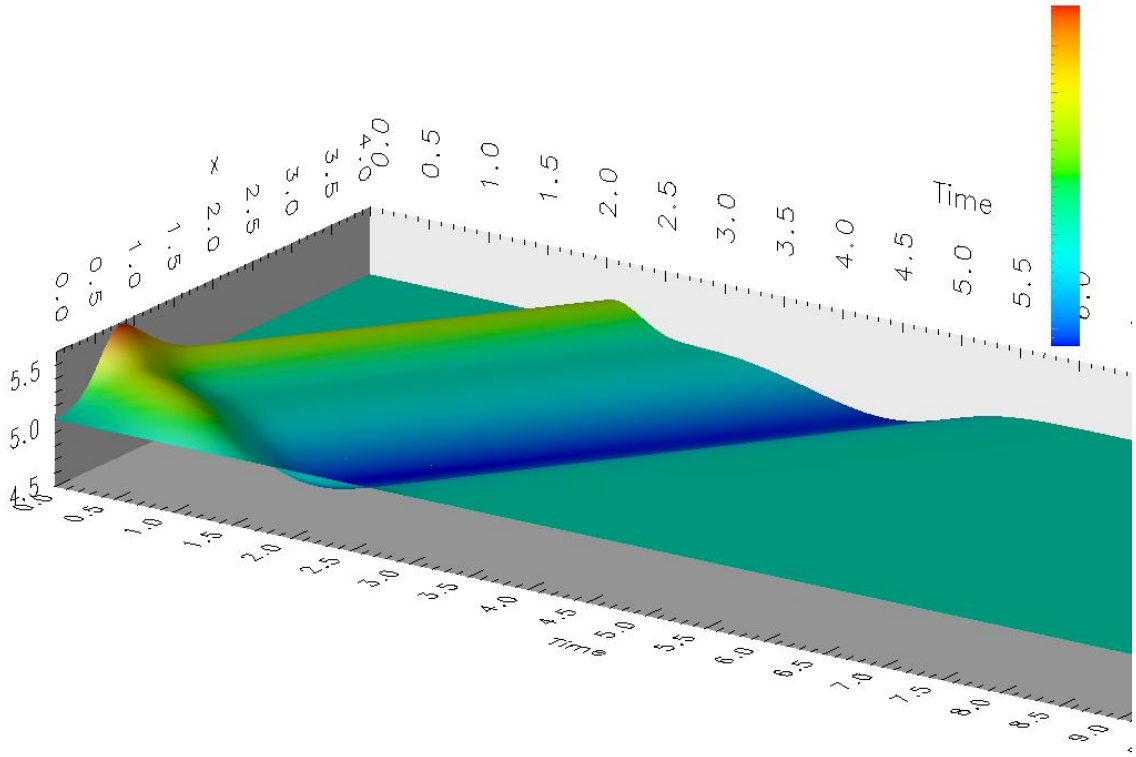

Figure 2: Temporal evolution of axial velocity in 1D gas dynamics problem *with* absorbing boundary condition at outlet.

This 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 $\mathbf{\Pi}^{\pm}$. If we can assume that the flow is composed of small perturbations around a reference state $\mathbf{U}_{\text{ref}}$, then we can compute the projection matrix at the state $\mathbf{U}_{\text{ref}}$

$$\mathbf{\Pi}(\mathbf{U}_{\text{ref}})_n^- \left(\mathbf{U} - \mathbf{U}_{\text{ref}}\right) = 0. \tag{23}$$

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

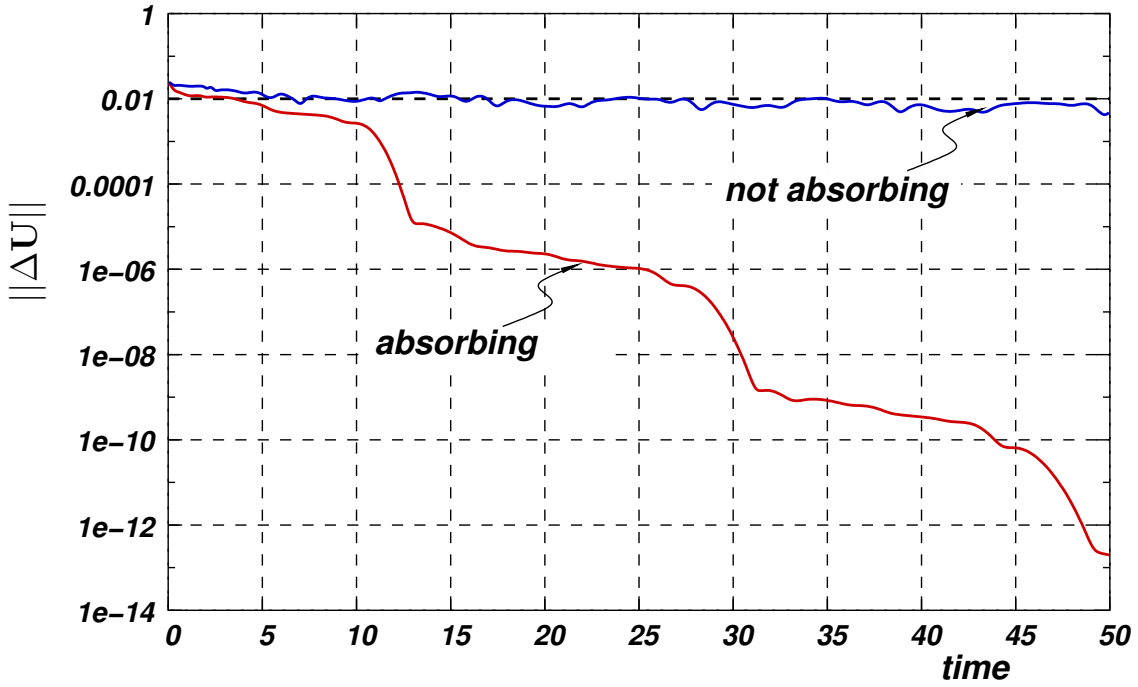

Figure 3: Rate of converge of 1D gas dynamics problem *with* and *without* absorbing boundary conditions.

4.4.1 Numerical example. Varying section compressible 1D flow.

Consider the 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

$$A(x) = A_0 \left(1 - C\,\frac{\tanh(x - Lx/2)}{L_c}\right). \tag{24}$$

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, $L_c = 0.136$ is a parameter controlling the width of the transition. We impose ρ and u at inlet and consider different outlet conditions, namely

- *non-absorbing*, $p =$cnst,
- *absorbing linear* (see (19)), and
- *absorbing non-linear* (see (23)).

In figure 4 we see the evolution of the state vector increment ($\|\Delta \mathbf{U}\|$) as

4.5 Riemann based absorbing boundary conditions

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

$$\mathbf{\Pi}^-(\mathbf{U}(t))\,(\mathbf{U}(t') - \mathbf{U}(t)) = 0. \tag{25}$$

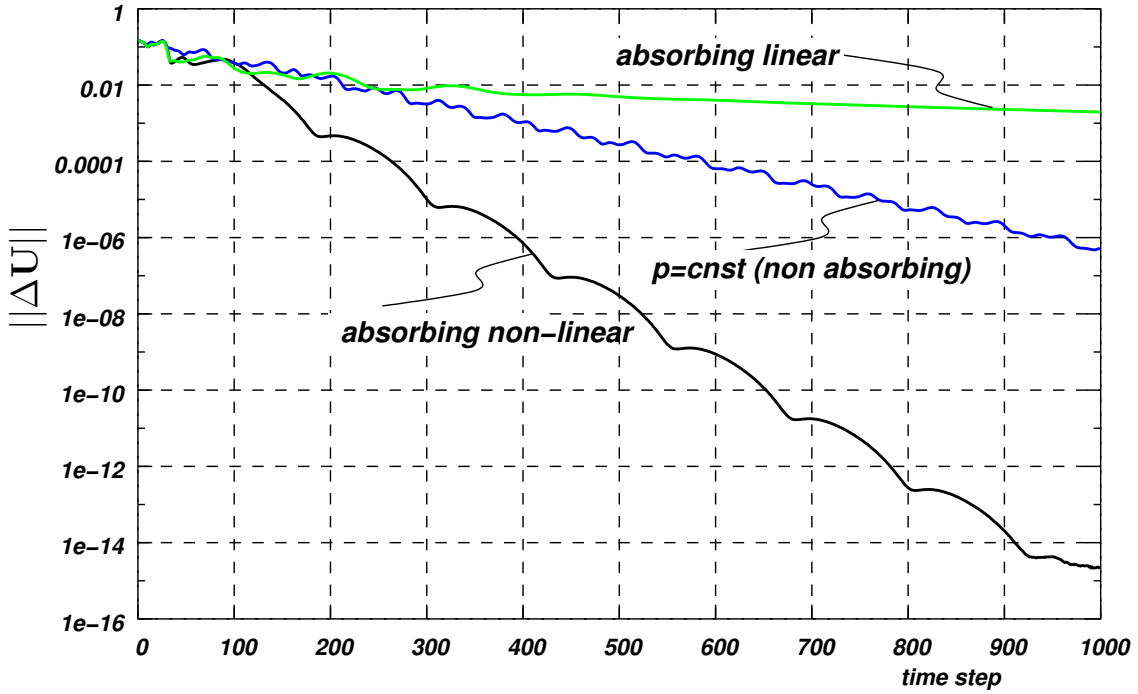

Figure 4: Rate of converge of 1D gas dynamics problem in full non-linear regime with diffrent kind of absorbing boundary conditions.

But regarding the equivalent expression (18) we can see that it can be written as

$$\mathbf{l}_j(\mathbf{U}) \cdot \mathrm{d}\mathbf{U} = 0, \quad \text{if } \lambda_j < 0, \tag{26}$$

where $\mathbf{l}_j$ is the j-th 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 *total differential*, i.e.

$$\mu(\mathbf{U})\, \mathbf{l}_j(\mathbf{U}) \cdot \mathrm{d}\mathbf{U} = \mathrm{d}w_j(\mathbf{U}), \tag{27}$$

for some non-linear function w_j and an "*integration factor*" $\mu(\mathbf{U})$, then we could impose

$$w_j(\mathbf{U}) = w_j(\mathbf{U}_{\mathrm{ref}}), \qquad (\text{for } w_j \text{ an } \textit{incoming} \text{ char.}) \tag{28}$$

which would be an absorbing boundary condition for the whole nonlinear 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 (reference...). 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 (27) along a path in state space, but the integration factor is not known.

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

$$w_\pm = u \pm \frac{2c}{\gamma - 1}. \tag{29}$$

Riemann invariants are known for the Riemann equations

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

Riemann invariants for channel flow are known only for rectangular and triangular shape and for triangular

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

4.6 Absorbing boundary conditions based on last state

While integrating the discrete equations in time, we can take the state of the fluid in the previous state as the reference state

$$\mathbf{\Pi}^-(\mathbf{U}^n)\,(\mathbf{U}^{n+1} - \mathbf{U}^n) = 0. \tag{32}$$

It is clear that the assumption of linearization is well justified, since in the limit of $\Delta t \to 0$ we should have $\mathbf{U}^{n+1} \approx \mathbf{U}^n$. In fact, (32) is equivalent, for $\Delta t \to 0$ to (26), so that if Riemann invariants exist, then this scheme preserves them, except for numerical errors. We call this strategy 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 if we look for a steady state at a certain regime, one has no way to guarantee that that regime will be obtained. For instance, if we want to obtain the steady flow around an aerodynamic profile at a certain Mach number, then we can set the initial state with a non perturbed constant flow at that conditions, but we can't assure that the final steady flow will preserve that Mach number. In practice we often use a mix of the strategies, 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 §4.2.1, with $\rho_{\text{ref}} = 1$, $u_{\text{ref}} = 0.2$, $p_{\text{ref}} = 0.714$, ($\text{Ma}_{\text{ref}} = 0.2$), $\delta\rho = \delta p = 0$, $\delta u = 0.6$, $\text{R} = 1$, $x_0 = 0.5L = 2$, $\sigma = 0.3$. 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 $N = 200$ equispaced finite elements ($h = L/N = 0.08$) with SUPG stabilization, Crank-Nicholson temporal scheme with $\Delta t = 0.02$ (CFL number ≈ 1.2). Absorbing boundary conditions based on the ULSAR strategy are applied at both ends $x = 0, L$. The values of the Riemann (29) are computed there and they are plotted in figure 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 end $x = L$. Convergence history is shown in figure 6. Note that absorption is very good, despite the full non-linear character of the flow.

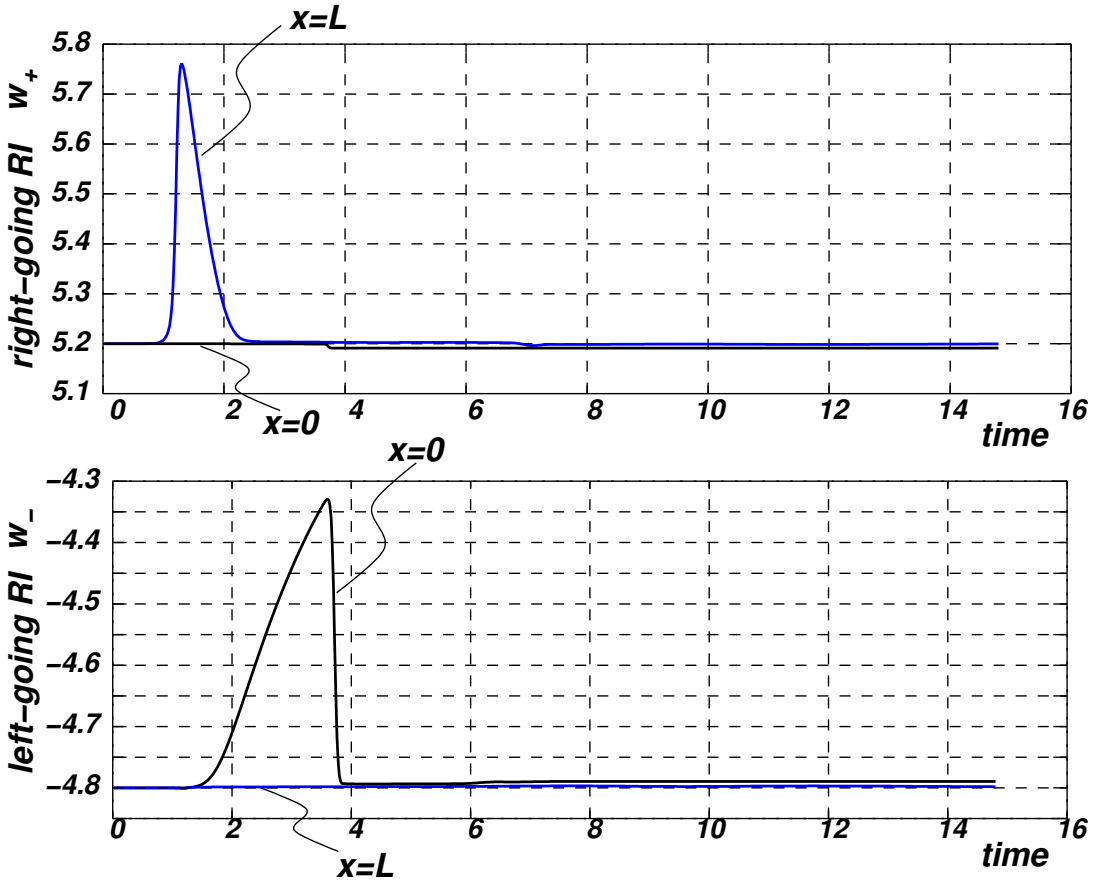

Figure 5: Riemann invariants at boundaries with ULSAR a.b.c.'s

4.7 Imposing non-linear absorbing boundary conditions

In this section we discuss how the absorbing boundary conditions can be integrated in a numerical code. For linear systems, the discrete version of equation (11) is of the form

$$\begin{aligned} \mathbf{C}\frac{\mathbf{U}_0^{n+1}-\mathbf{U}_0^n}{\Delta t}+\mathbf{A}\,\frac{\mathbf{U}_1^{n+1}-\mathbf{U}_0^n}{h}&=0;\\ \mathbf{C}\frac{\mathbf{U}_k^{n+1}-\mathbf{U}_k^n}{\Delta t}+\mathbf{A}\,\frac{\mathbf{U}_{k+1}^{n+1}-\mathbf{U}_{k-1}^n}{2h}&=0,\quad k\geq 1 \end{aligned} \tag{33}$$

where $\mathbf{U}_k^n$ is the state at grid point k at time $t^n = n\Delta t$. We assume a constant mesh step size of h, i.e. $x_k = kh$, and assume a boundary at mesh node $x_0 = 0$. We have made a lot of simplifications here, no source or upwind terms, and a simple discretization based on centered finite differences. Alternatively, it can be thought as a pure Galerkin FEM discretization with mass lumping.

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

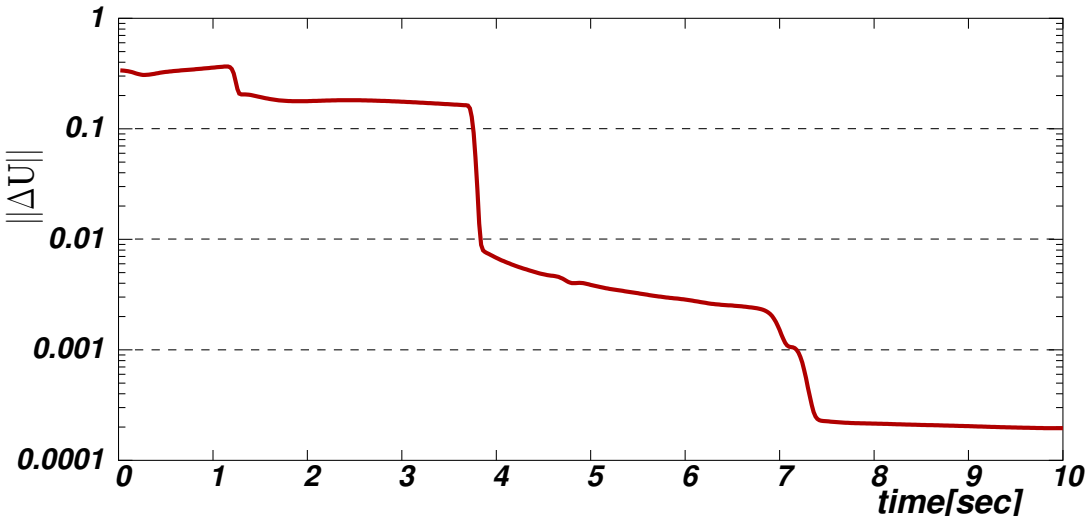

Figure 6: Convergence history when using with ULSAR a.b.c.'s

happens for instance in a supersonic inlet for gas dynamics or an inlet boundary for linear advection. In this case we simply replace the balance equation for the boundary node (the first equation in (33)) with the absorbing condition $\mathbf{U} = \mathbf{U}_{\text{ref}}$, keeping the balance between equations and unknowns.

Conversely, if the 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 we simply discard the absorbing condition $\mathbf{U} = \mathbf{U}_{\text{ref}}$. Again the number of equations and unknowns is maintained.

The case is more complicated when $0 < n_+ < n$ we can't simply add the absorbing condition (either (19), (28) or (32)), because we can neither discard the boundary balance equation nor keep it.

There are at least two strategies for imposing this non-linear boundary conditions. One 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 can not 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 we will describe in detail this second strategy.

In the base of the characteristic variables $\mathbf{V}$ (33) can be written as

$$\begin{aligned} \frac{\mathbf{V}_0^{n+1}-\mathbf{V}_0^n}{\Delta t}+\mathbf{\Lambda}\,\frac{\mathbf{V}_1^{n+1}-\mathbf{V}_0^n}{h}&=0;\\ \frac{\mathbf{V}_k^{n+1}-\mathbf{V}_k^n}{\Delta t}+\mathbf{\Lambda}\,\frac{\mathbf{V}_{k+1}^{n+1}-\mathbf{V}_{k-1}^n}{h}&=0,\quad k\geq 1. \end{aligned} \tag{34}$$

For the linear absorbing boundary conditions (19) we should impose

$$\mathbf{\Pi}_V^+(\mathbf{V}_{\rm ref})\,(\mathbf{V}_0-\mathbf{V}_{\rm ref})=0. \tag{35}$$

while discarding the equations corresponding to the incoming waves in the first rows of (34). Here $\mathbf{U}_{\rm ref}/\mathbf{V}_{\rm ref}$ is the state about which we make the linearization.

4.7.1 Using Lagrange multipliers

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

$$\begin{aligned} \mathbf{\Pi}_V^+(\mathbf{V}_{\rm ref})\,(\mathbf{V}_0-\mathbf{V}_{\rm ref})+\mathbf{\Pi}_V^-(\mathbf{V}_{\rm ref})\,\mathbf{V}_{lm}&=0,\\ \frac{\mathbf{V}_0^{n+1}-\mathbf{V}_0^n}{\Delta t}+\mathbf{\Lambda}\,\frac{\mathbf{V}_1^{n+1}-\mathbf{V}_0^n}{h}+\mathbf{\Pi}_V^+(\mathbf{V}_{\rm ref})\,\mathbf{V}_{lm}&=0;\\ \frac{\mathbf{V}_k^{n+1}-\mathbf{V}_k^n}{\Delta t}+\mathbf{\Lambda}\,\frac{\mathbf{V}_{k+1}^{n+1}-\mathbf{V}_{k-1}^n}{2h}&=0,\quad k\geq 1. \end{aligned} \tag{36}$$

where $\mathbf{V}_{lm}$ are the Lagrange multipliers for the imposition of the new conditions. Note that, if j is an incoming wave ($\lambda_j\geq 0$), then the equation is of the form

$$\begin{aligned} v_{j0}-v_{\rm ref0}&=0\\ \frac{v_{j0}^{n+1}-v_{j0}^n}{\Delta t}+\lambda_j\frac{v_{j1}^{n+1}-v_{j0}^n}{h}+v_{j,lm}&=0\\ \frac{v_{jk}^{n+1}-v_{jk}^n}{\Delta t}+\lambda_j\frac{v_{j,k+1}^{n+1}-v_{jk}^n}{2h}&=0,\quad k\geq 1 \end{aligned} \tag{37}$$

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

On the other hand, for the outgoing waves ($\lambda_j<0$), we have

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

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

Coming back to the $\mathbf{U}$ basis, we have

$$\begin{aligned} \mathbf{\Pi}_U^+(\mathbf{U}_{\text{ref}})\,(\mathbf{U}_0 - \mathbf{U}_{\text{ref}}) + \mathbf{\Pi}_U^-(\mathbf{U}_{\text{ref}})\,\mathbf{U}_{lm} = 0, \\ \mathbf{C}\frac{\mathbf{U}_0^{n+1} - \mathbf{U}_0^n}{\Delta t} + \mathbf{A}\,\frac{\mathbf{U}_1^{n+1} - \mathbf{U}_0^n}{h} + \mathbf{C}\mathbf{\Pi}_U^+(\mathbf{U}_{\text{ref}})\,\mathbf{U}_{lm} = 0; \\ \mathbf{C}\frac{\mathbf{U}_k^{n+1} - \mathbf{U}_k^n}{\Delta t} + \mathbf{A}\,\frac{\mathbf{U}_{k+1}^{n+1} - \mathbf{U}_{k-1}^n}{2h} = 0, \quad k \geq 1. \end{aligned} \tag{39}$$

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 (39)

$$\begin{aligned} -\epsilon\mathbf{U}_{lm} + \mathbf{\Pi}_U^+\,(\mathbf{U}_0 - \mathbf{U}_{\text{ref}}) + \mathbf{\Pi}_U^-\,\mathbf{U}_{lm} = 0, \\ \mathbf{C}\frac{\mathbf{U}_0^{n+1} - \mathbf{U}_0^n}{\Delta t} + \mathbf{A}\,\frac{\mathbf{U}_1^{n+1} - \mathbf{U}_0^n}{h} + \mathbf{\Pi}_U^+\,\mathbf{U}_{lm} = 0; \end{aligned} \tag{40}$$

where, for the moment, we dropped the dependence of the projectors on $\mathbf{U}_{\text{ref}}$. Eliminating $\mathbf{U}_{lm}$ from the first and second rows we obtain

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

Now, using projection algebra we can show that

$$(\mathbf{\Pi}_U^- + \epsilon\mathbf{I})^{-1} = (\frac{1}{\epsilon}\mathbf{\Pi}_U^+ + \frac{1}{1+\epsilon}\mathbf{\Pi}_U^-) \tag{42}$$

so that the last term in (41) reduces to $\mathbf{\Pi}_U^+(\mathbf{U}_0 - \mathbf{U}_{\text{ref}})$ and the whole equation is

$$\mathbf{C}\frac{\mathbf{U}_0^{n+1} - \mathbf{U}_0^n}{\Delta t} + \mathbf{A}\,\frac{\mathbf{U}_1^{n+1} - \mathbf{U}_0^n}{h} + \frac{1}{\epsilon}\mathbf{C}\mathbf{\Pi}_U^+(\mathbf{U}_0 - \mathbf{U}_{\text{ref}}) = 0. \tag{43}$$

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

5 DYNAMICALLY VARYING BOUNDARY CONDITIONS

5.1 Varying boundary conditions in external aerodynamics

During flow computation it the number of incoming characteristics n_+ may change. This can occur due to the flow changing regime (i.e. from subsonic to supersonic) or due to the flow changing sense (flow reversal). A typical case is the external flow around an aerodynamic body as shown in figure 7. 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 ($n_+ = 1$). We can determine whether the boundary is inlet or outlet by looking at the projection of the unperturbed flow velocity $\mathbf{u}_\infty$ with the local normal $\hat{\mathbf{n}}$. For the steady supersonic case the situation is very different. A bow shock develops in from 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 non-perturbed flow. At the boundary we have now a supersonic inlet region, and on the outlet region we have both subsonic and supersonic parts. The point where the flow at outlet changes from subsonic to supersonic may be estimated from the Mach angle, but it may very inaccurate if the boundary is close to the body. Having a boundary condition that can automatically adapt to the whole possibilities can be of great help in such a case. Now, consider the unsteady case, for instance a body accelerating slowly 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 point of separation between both will be changing position, following approximately the instantaneous Mach angle.

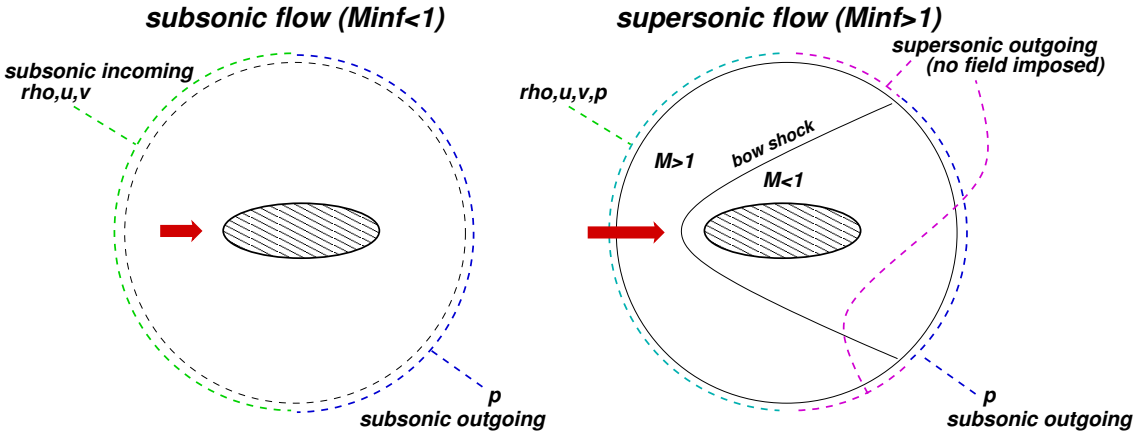

Figure 7: Number of incoming characteristics changing on accelerating body.

5.2 Aerodynamics of falling objects

For instance, one interesting case is the aerodynamics of a falling body.[9,10,11,12,13] Consider, for simplicity, a two dimensional case of an homogeneous ellipse in free fall. As the body accelerates, the pitching moments tend to increase the angle of attack until it stalls (A), and then the body starts to fall towards its other end accelerating while its main axis aligns with gravity (B). As the body accelerates the pitching moment growths until it eventually stalls again (c), and so on... This kind of falling mechanism is typical of slender bodies with relatively small moment of inertia like a sheet of paper and is called "*flutter*". However, depending of several parameters, but mainly depending of the moment of inertia of the body, if it has a large angular moment at (B) then it may happen that it rolls on itself, keeping always the same sense of

rotation. This kind of falling mechanism is called tumbling and is characteristic of less slender and more 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.

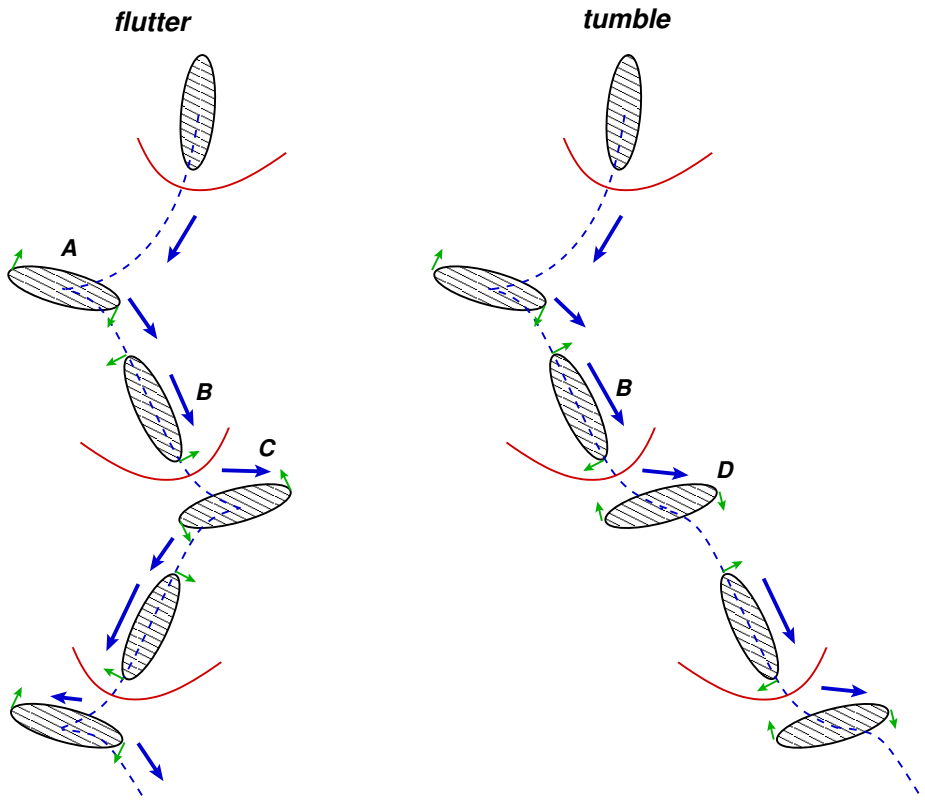

Figure 8: Falling ellipse

As the body falls it accelerates and can reach supersonic speeds. This depends on the density of the body relative to the surrounding atmosphere and 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.

One can model a falling body in several ways. In order to avoid the use of deforming meshes, a fixed mesh attached to the body can be used. Then one can choose to perform the computation in a *non-inertial* frame moving with the body or to perform the computation in an inertial frame using 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 we choose to use the first strategy.

The computation of the flow is linked to the dynamics of the falling object. The strategy is a typically staggered fluid/solid interaction process.[14,15,16] Basically we solve the fluid problem

in an non-inertial frame with inertial terms computed with the actual state of the body (linear acceleration $\mathbf{a}$, angular rotation velocity ω and angular rotation acceleration $\dot{\omega}$). Also boundary conditions in the non-inertial frame at infinity must take into account the actual linear and angular velocity of the object. The fluid solver updates the state of the fluid from t^n to t^{n+1}. Then, with the state of the fluid at t^{n+1} the forces exerted by the fluid on the body are computed. With this forces, the equations for the rigid motion of the body are solved (six degrees of freedom, two linear position and velocities, rotation angle and its derivative).

Coming back to the boundary conditions issue, we have now in addition to the fact that the body can accelerate and decelerate, and going back and forth from subsonic to supersonic speeds, 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 non-inertial frame fixed to the body.

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 $e = \sqrt{1 - b^2/a^2} = 0.8$),
- $m = 20$, (mass),
- $I = 50$, (moment of inertia),
- $g = 0.15$, (acceleration of gravity),
- $\rho_a = 1$, (atmosphere density),
- $p = 1$, (atmosphere pressure),
- $\gamma = 1.4$, (gas adiabatic index $\gamma = C_p/C_v$),

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

$$F_{\text{aero}} + W + F_{\text{float}} = C_D \rho_a v^2 A - \rho_s g V + \rho_a g V \tag{44}$$

where $V = \pi a b$ 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 $\rho_s = m/V, \rho_a$ the densities of solid and atmosphere respectively. For the data above this estimation gives a limit speed of $v = 3.1$ approximately. As 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 $(0, -2)$, and an angle of its major axis of 10° with respect with the vertical, the fluid is initially at rest. The computed trajectory until $t = 1.83$ time units is shown in figure 9. The mean vertical velocity during this period was -1.8. The computed trajectory is shown in a reference system falling at velocity -1.46 (this is done in order to reduce the vertical span of the plot). In figures 10 we see colormaps of Mach number at four instants, in the non inertial frame fixed to the body. The instants are marked

as A, B, C, D 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 cope 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. (An animation and further material can be found at the author home page).

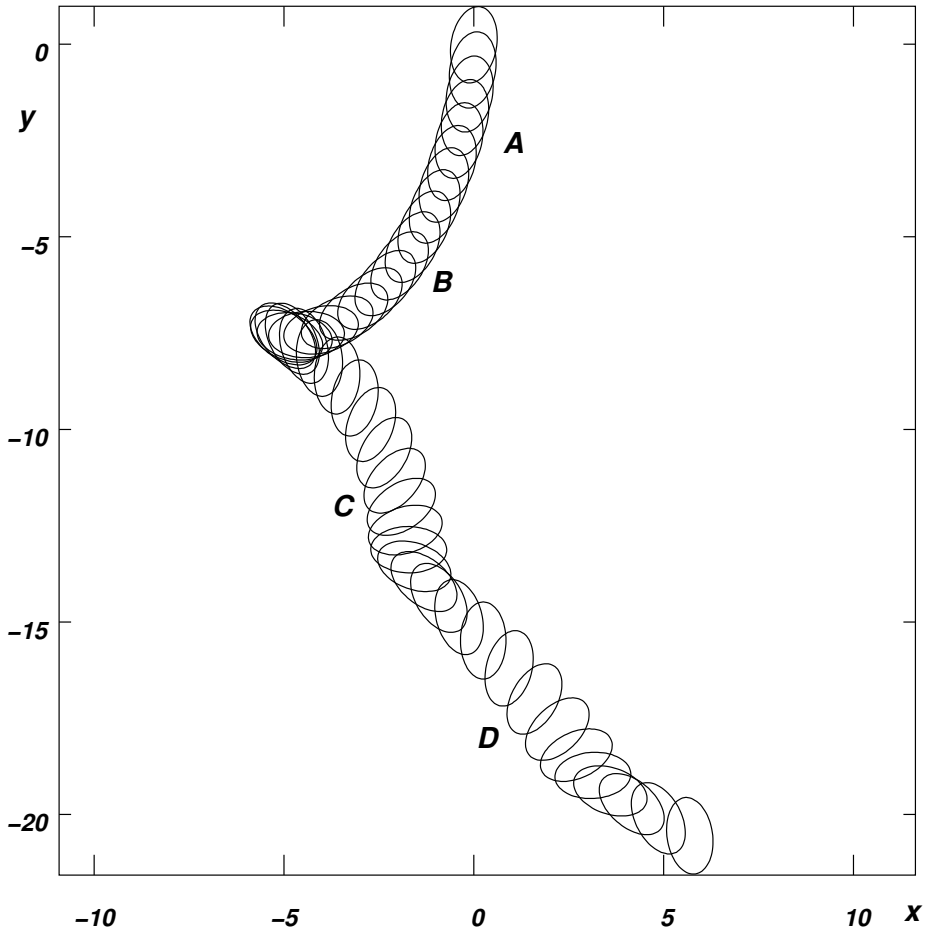

Figure 9: Computed trajectory of falling ellipse

6 CONCLUSIONS

Absorbing boundary conditions reduces computational cost by allowing to put the artificial exterior boundary nearer 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. For 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

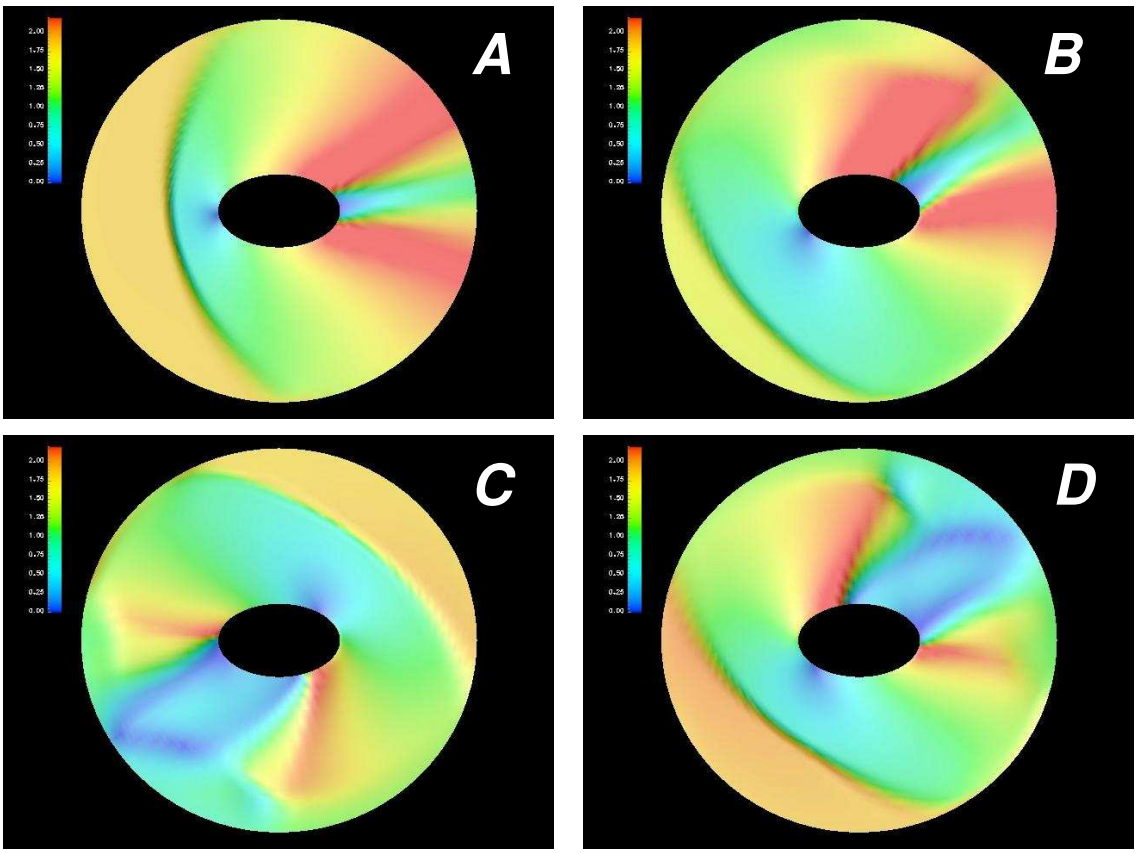

Figure 10: Ellipse falling at supersonic speeds. Colormaps of Mach number.

penalization techniques.

7 ACKNOWLEDGMENT

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). We made extensive use of freely distributed software as GNU/Linux OS, MPI, PETSc, GCC compilers, Octave, Open-DX among many others.

REFERENCES

[1] D. Givoli and J.B. Keller. Non-reflecting boundary conditions for elastic waves. *Wave Motion*, **12**, 261–279 (1990).

[2] D. Givoli and J.B.; Keller. A finite element method for large domains. *Computer Methods in Applied Mechanics and Engineering* , **76**, 41–66 (1989).

[3] J. Broeze and J.E. Romate. Absorbing boundary conditions for free surface wave simulations with a panel method. *Journal of Computational Physics* , **99**, 146 (1992).

[4] I. Harari and T.J.R. Hughes. Galerkin least-squares finite element methods for the reduced wave equation with non-reflecting boundary conditions in unbounded domains. *Computer Methods in Applied Mechanics and Engineering* , **98**, 411–454 (1992).

[5] M. Storti, J. D'Elía, and S. Idelsohn. Algebraic discrete non-local (dnl) absorbing boundary condition for the ship wave resistance problem. *Journal of Computational Physics* , **146**, 570–602 (1997).

[6] T Hagstrom. Boundary conditions at outflow for a problem with transport and diffusion. *Journal of Computational Physics* , **69**, 69–80 (1987).

[7] S.V. Tsynkov. Numerical solution of problems on unbounded domains. a review. *Applied Numerical Mathematics*, **27**, 465–532 (1998).

[8] C. Baumann, Storti M., and S Idelsohn. Improving the convergence rate of the petrov-galerkin techniques for the solution of transonic and supersonic flows. *International Journal for Numerical Methods in Engineering*, **34**, 543–568 (1992).

[9] S. Field, M. Klaus, M. Moore, , and Franco Nori. Instabilities and chaos in falling objects. *Nature*, (388), 252–254 (1997).

[10] A. Belmonte. Flutter and tumble in fluids. *Physics World*, (1999).

[11] J.Y. Huang. Trajectory of a moving curveball in viscid flow. In *Proceedings of the Third International Conference: Dynamical Systems and Differential Equations*, pages 191–198, (2000).

[12] J.Y. Huang. *Moving Boundaries VI*, chapter Moving Coordinates Methods and Applications to the Oscillations of a Falling Slender Body, pages 73–82. WIT Press, (2001).

[13] J.Y. Huang. *Advances in Fluid Mechanics IV*, chapter Aerodynamics of a Moving Curveball in Newtonian Flow, pages 597–608. WIT Press, (2002).

[14] J. Cebral. *Loose Coupling Algorithms for fluid structure interaction.* PhD thesis, Institute for Computational Sciences and Informatics, George Mason University, (1996).

[15] R. Löhner, C. Yang, J. Cebral, J. Baum, H. Luo, D. Pelessone, and C. Charman. Fluid-structure interaction using a loose coupling algorithm and adaptive unstructured grids. *AIAA paper AIAA-98-2419*, (1998).

[16] R. Lohner and J. R. Cebral. Fluid-structure interaction in industry: Issues and outlook. In *Proc. World User Association in Applied Computational Fluid Dynamics, 3rd World Conference in Applied Computational Fluid Dynamics, Germany, May 19-23*, (1996).