A strategy to implement Dirichlet boundary conditions in the context of ADER finite volume schemes. One-dimensional conservation laws

Gino I. Montecinos[a,1,*]

[a] Center for Mathematical Modeling (CMM)
Universidad de Chile
Santiago, Chile

Abstract

ADER schemes are numerical methods, which can reach an arbitrary order of accuracy in both space and time. They are based on a reconstruction procedure and the solution of generalized Riemann problems. However, for general boundary conditions, in particular of Dirichlet type, a lack of accuracy might occur if a suitable treatment of boundaries conditions is not properly carried out. In this work the treatment of Dirichlet boundary conditions for conservation laws in the context of ADER schemes, is concerned. The solution of generalized Riemann problems at the extremes of the computational domain, provides the correct influence of boundaries. The reconstruction procedure, for data near to the boundaries, demands for information outside the computational domain, which is carried out in terms of ghost cells, which are provided by using the numerical solution of auxiliary problems. These auxiliary problems are hyperbolic and they are constructed from the conservation laws and the information at boundaries, which may be partially or totally known in terms of prescribed functions. The evolution of these problems, unlike to the usual manner, is done in space rather than in time due to that these problems are named here, *reverse problems*. The methodology can be considered as a numerical counterpart of the inverse Lax-Wendroff procedure for filling ghost cells. However, the use of Taylor series expansions, as well as, Lax-Wendroff procedure, are avoided. For the scalar case is shown that the present procedure preserve the accuracy of the scheme which is reinforced with some numerical results. Expected orders of accuracy for solving conservation laws by using the proposed strategy at boundaries, are obtained up to fifth-order in both space and time.

Keywords:
Inverse Lax-Wendroff procedure, Dirichlet boundary conditions, ADER finite volume schemes.

1. Introduction

In this article the treatment of Dirichle boundary conditions in the context of high-order finite volume schemes for conservation laws is concerned. To construct schemes of high-order of accuracy, the reconstruction procedure is a key, for which the so-called stencils are required. These

---

*Corresponding author
[1]Email address: gmontecinos@dim.uchile.cl

are sets of data, which allows the construction of a polynomial representation of the solution within computational cells. It is in general a well described process for cells within a computational domain, however, for cells near to boundaries is not so. In boundary treatment, it is usual, the use of ghost cells to provided the required information outside computational domains. Cells extrapolation techniques are usually employed to obtain ghost cells. Lagrangian extrapolation is often used. However, it only provides good approximations for smooth solutions. For the presence of shocks, a WENO extrapolation procedure has been proposed in [62]. However, the strategy proposed by Tan and collaborators [44, 45, 46], Huang et al. [28] and Xiong et al. [58], seems to be the most efficient in a wide range of solution regimes. In these references, ghost cells are computed by using Taylor series expansions in space around points in the boundary. To compose these Taylor expansions the governing equation is repeatedly used to provide space-derivatives in terms of time-derivatives similarly to the Cauchy-Kowalewskaya or Lax-Wendroff procedure, this process is known as the *inverse Lax-Wendroff procedure*. In these works, the inverse Lax-Wendroff procedure was implemented for inflow, whereas for outflow boundaries, extrapolation techniques were concerned. This treatment has been implemented for linear problems, Vilar et al. [56] as well as non linear problems [23] to mention but a few, where the global scheme, has resulted to be stable for any position of the boundaries. To apply this methodology, a manner to identify if the boundary is an inflow or an outflow boundary, is needed. A form to identify and apply the type of boundary condition, is given in terms of Riemann problems. This technique is similar to that proposed by Berger et al. [5, 6], where irregular cells are incorporated near to boundaries in order to provide new cells which allow to construct Riemann problems. In this way the inflow and outflow boundaries are automatically selected from the eigenstructure of the Riemann problem.

In this paper, the treatment of hyperbolic conservation laws with dirichelt boundary conditions in the context of ADER finite volume schemes is concerned. The ADER approach was first put forward by Toro et al. [52, 54, 47] for linear hyperbolic problems, see also [42]. Later, this method has also been successfully applied to solve several non-linear hyperbolic systems in Cartesian and unstructured meshes, [10, 29, 30, 31, 43, 49, 55]. ADER has also been extended in the framework of discontinuous Galerkin finite element methods by Dumbser and collaborators, see for example [19, 13, 20, 17, 18, 39, 40]. See also extensions of the ADER approach to the equations of magnetohydrodynamics by Balsara and collaborators [3, 1] and extensions to advection-diffusion-reaction type equations [48, 51, 27, 14, 26, 59, 36, 53]. More recent results of the ADER approach include [37, 38, 33, 21, 22, 16, 7, 9, 8, 2, 35]. The ADER type methods are based on two main steps to compute numerical solutions with arbitrary order of accuracy; i) a reconstruction procedure and ii) the solution of a Generalized Riemann Problem (GRP). The reconstruction procedure has to be non-linear in order to circumvent Godunov's theorem, [24, 50]. This procedure at each time step requires a set of data, called stencil. On the other hand, the solution of GRP's allows a correct wave propagation. See [11, 34, 38] and chapters 19 and 20 of [50] for a review of Generalized Riemann solvers.

Periodic boundary conditions, as well as, transmissive and reflective types, can easily be implemented in the context of ADER methods. A straightforward manner to define generalised Riemann problems, as well as, to generate stencils for the reconstruction procedure is available. However, for general Dirichlet boundary condition types, the reconstruction procedure requires information outside the computational domain in order to complete the required stencils. Additionally, the wave propagation at boundaries must be correctly described. Therefore, in this paper a strategy to provide the needed information in the reconstruction stage, as well as, the correct wave propagation at boundaries through a generalised Riemann problem, are presented.

Ghost cells, which allow to create stencils, are obtained from the solution of auxiliary problems, which are constructed from the governing equations and the information at boundaries. These auxiliary problems are hyperbolic and they may be solved by any finite volume scheme. This strategy can be considered as a numerical counter part to the inverse Lax-Wendroff procedure, but the use of Taylor series expansions and the Cauchy-Kowalewskaya procedure are avoided. At this point we identify two types of problems, the first one, is defined inside the computational domain and it is the problem of interest for us. This is called here, *interior problem*. The second type is given by the auxiliary problems which are defined outside the computational domain and allow to construct ghost cells. We remark that in opposite to the interior problem, for auxiliary problems the evolution is carried out in space rather than in time, due to that these auxiliary problems are called *reverse problems*. The methodology does not require the use of extrapolations in the usual manner and as we will show later, the methodology preserves the order of accuracy for numerical solutions of the interior problem.

The structure of the paper is as follows. In section 2, the interior problem and the numerical scheme is discussed. In section 3 the treatment of Dirichelt boundary conditions is introduced. In section 4 the numerical scheme for auxiliary problems is presented. Some theoretical results concerning the scalar case are obtained in section 5. In section 6, numerical results are shown. Finally, conclusion are drawn in section 7.

2. The interior problem and the numerical scheme

Let us consider hyperbolic conservation laws in the form

$$\left.\begin{array}{rll} \partial_t\mathbf{Q} + \partial_x\mathbf{F}(\mathbf{Q}) &=& \mathbf{0}\,, \qquad x_L \leq x \leq x_R\,, 0 \leq t \leq T\,, \\ \mathbf{Q}(x,0) &=& \mathbf{H}_0(x)\,, \end{array}\right\} \tag{1}$$

here, $\mathbf{H}_0(x)$ is the initial condition, $\mathbf{Q} \in \mathbb{R}^m$ is the vector of unknowns, $\mathbf{F}(\mathbf{Q}) \in \mathbb{R}^m$ is the physical flux. On boundaries x_L and x_R, inlet and outlet boundaries are observed. In the best case prescribed functions $\mathbf{G}_L(t) \in \mathbb{R}^m$ and $\mathbf{G}_R(t) \in \mathbb{R}^m$ are provided, that means $\mathbf{Q}(x_L,t) = \mathbf{G}_L(t)$ and $\mathbf{Q}(x_R,t) = \mathbf{G}_R(t)$. Of course, the prescribed functions influence the evolution at a given time t, according to the eigenstructure of the problem, the number of effective boundary conditions depends on the sign of eigenvalues. We remark that, in general cases, all components of $\mathbf{G}_L(t)$ or $\mathbf{G}_R(t)$ are not prescribed, so in the following sections we will deal the case in which component associated to outlet boundaries are not prescribed and approximate values will be provided.

Problem (1) is defined inside the computational domain, due to that it is referred as *the interior problem*. Let us derive the conventional finite volume formulation for (1). So, let us consider a partition of $[x_L, x_R]$ into N_{int} sub intervals $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ with uniform length $\Delta x = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$, $i = 1, ..., N_{int}$, where $x_{-\frac{1}{2}} = x_L$ and $x_{N_{int}+\frac{1}{2}} = x_R$. Therefore, integrating the governing equation in the space-time interval $I_i^n = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [t^n, t^{n+1}]$, we obtain the evolution formula

$$\mathbf{Q}_i^{n+1} = \mathbf{Q}_i^n - \frac{\Delta t}{\Delta x}\left[\mathbf{F}_{i+\frac{1}{2}} - \mathbf{F}_{i-\frac{1}{2}}\right]\,, \tag{2}$$

with

$$\mathbf{Q}_i^n = \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{Q}(x, t^n)dx\,,$$

$$\mathbf{F}_{i+\frac{1}{2}} = \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \mathbf{F}(\mathbf{Q}(x_{i+\frac{1}{2}}, t))dt\,, \tag{3}$$

3

where $\mathbf{Q}_i^n$ is the cell average and $\mathbf{F}_{i+\frac{1}{2}}$ is the numerical flux. A scheme is completely determined once the numerical flux is defined. In this paper we are interested in the family of ADER schemes, which can reach an arbitrary order of accuracy in space and time. These are based on two steps; i) a reconstruction procedure and ii) the solution of a Generalized Riemann Problem (GRP).

The reconstruction procedure of order M, provides a set of polynomials $\mathbf{P}_i(x)$, which are constructed from sets of data, called stencils. Thus, for example to construct the reconstruction polynomial inside the cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, the stencils are

$$
\begin{aligned}
\mathcal{S}_{-1,i}^M &= \{\mathbf{Q}_{i-M}^n, ..., \mathbf{Q}_i^n\} \,, \\
\mathcal{S}_{0,i}^M &= \{\mathbf{Q}_{i-M}^n, ..., \mathbf{Q}_i^n, ..., \mathbf{Q}_{i+M}^n\} \,, \\
\mathcal{S}_{+1,i}^M &= \{\mathbf{Q}_i^n, ..., \mathbf{Q}_{i+M}^n\} \,.
\end{aligned}
\tag{4}
$$

From each stencil $\mathcal{S}_{l,i}^M$, with $l = -1, 0, +1$, a polynomial of degree $M - 1$ denoted by $\mathbf{P}_{i,l}(x)$, is constructed. A polynomial reconstruction basis $\{\theta_k(\xi)\}_{k=1}^M$ defined in $[0, 1]$, is considered. So, each polynomial has the form

$$
\mathbf{P}_{i,l}(x) = \sum_{k=1}^M \gamma_k^l \theta_k \left(\frac{x - x_{i-\frac{1}{2}}}{\Delta x} \right) \,,
\tag{5}
$$

where γ_k^l are found by solving

$$
\mathbf{Q}_j^n = \sum_{k=1}^M \gamma_k^l \int_j^{j+1} \theta(\xi)_k d\xi \,, \quad \mathbf{Q}_j^n \in S_l^M \,,
\tag{6}
$$

which is a linear problem for γ_k^l. For $\mathcal{S}_0^M$ we have more elements than degrees of freedom, so an overestimated system has to be solved, which is carried out through a constrained least-squares technique, see [17]. So, the reconstruction polynomial with support in $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ is constructed as

$$
\mathbf{P}_i(x) = \omega_{-1} \mathbf{P}_{i,-1}(x) + \omega_0 \mathbf{P}_{i,0}(x) + \omega_{+1} \mathbf{P}_{i,+1}(x) \,,
\tag{7}
$$

where ω_l, $i = -1, 0, +1$ are weights, which are obtained as follows. Firstly, we compute

$$
\tilde{\omega}_l = \frac{\lambda_l}{(\epsilon + \sigma_l)^r} \,,
\tag{8}
$$

where $\lambda_{-1} = 1$, $\lambda_0 = 10^5$, $\lambda_{+1} = 1$, $\epsilon = 10^{-14}$, $r = 4$ and σ_l is computed as

$$
\sigma_l = \sum_{r=1}^M \sum_{k=1}^M \gamma_k^l \int_j^{j+1} \left(\frac{d^r}{d\xi^r} \theta_k(\xi) \right)^2 d\xi \,.
\tag{9}
$$

Secondly, we normalize these values

$$
\omega_l = \frac{\tilde{\omega}_l}{\tilde{\omega}_{-1} + \tilde{\omega}_0 + \tilde{\omega}_{+1}} \,,
\tag{10}
$$

with $l = -1, 0, +1$. See [25, 12, 17] for further details.

On the other hand, in ADER methods the numerical flux in (3), $\mathbf{F}_{i+\frac{1}{2}}$, is obtained from evaluating the integral

$$\mathbf{F}_{i+\frac{1}{2}} = \int_0^1 \mathbf{F}_h(\mathbf{q}_i(1,\tau), \mathbf{q}_{i+1}(0,\tau))d\tau, \tag{11}$$

where $\mathbf{F}_h(\mathbf{q}_i, \mathbf{q}_{i+1})$ denotes a Riemann solver, which depends on two arguments $\mathbf{q}_i(1,\tau)$ and $\mathbf{q}_{i+1}(0,\tau)$, which are the high order extrapolated values of the data on the left and right side of the interface $x_{i+\frac{1}{2}}$ at time τ, respectively. In particular the Rusanov solver has been implemented in this work. Here, the reconstruction polynomials are used to form the GRP's, as illustrated in [37], section 2.3. In addition, see chapters 19 and 20 of [50] for a review of Generalized Riemann solvers and [11, 34] for a comparison of GRP solvers and references therein.

Note that for cells near to boundaries, stencils should require values outside the computational domain. For example, to construct the reconstruction polynomial for the first cell of the computational domain, stencils $\mathcal{S}_{-1,1}^M$ and $\mathcal{S}_{0,1}^M$ should contain M values outside the left boundary. For some special boundary condition types, like periodic, the stencils are filled with information which is available inside the computational domain. But for the general case we most provide these values, in this case, these correspond to ghost cells.

The aim of this work is the treatment of boundary conditions, which is carried out in two steps. The first one, consists on the computation of the ghost cells to fill the stencils for the reconstruction procedure, the procedure is presented in detail in the following sections. The second issue regards the correct wave propagation coming from boundaries, it is carried out by using a commonly used strategy to estimate fluxes as boundaries as follows

$$
\begin{aligned}
\mathbf{F}_{-\frac{1}{2}} &= \int_0^1 \mathbf{F}_h(\mathbf{G}_L(t^n + \tau\Delta t), \mathbf{q}_1(0_+,\tau))d\tau \ , \\
\mathbf{F}_{N_{int}+\frac{1}{2}} &= \int_0^1 \mathbf{F}_h(\mathbf{q}_{N_{int}}(1_-,\tau), \mathbf{G}_R(t^n + \tau\Delta t))d\tau \ .
\end{aligned}
\tag{12}
$$

Here, $\mathbf{G}_L(t)$ and $\mathbf{G}_R(t)$ are functions defined on the interfaces, in which all their components are prescribed functions or they have to be approximated at each interval $[t^n, t^{n+1}]$. In this way the inflow and outflow boundaries are automatically selected by the Riemann solver.

3. Reverse problems for Dirichlet boundary conditions

The aims here are twofold; first, provide the data at the interface to build Riemann problems, and second the computation of ghost cells, which will allow to fill stencils for the construction procedure.

3.1. Computation of ghost cells

Here we present a strategy to complete the stencils (4) given by $\mathcal{S}_{l,i}^M$ with $l = -1, 0, +1$ at the left and right extremes $i = 1, ..., M$ and $i = N_{int} - M, ..., N_{int}$, respectively. Let us assume that the physical flux is invertible, in the sense that, there exists an operator $\mathbf{R}$ such that

$$\mathbf{R}(\mathbf{F}(\mathbf{Q})) = \mathbf{Q} \ . \tag{13}$$

Therefore, we note that the governing equation (1) can be written as

$$\partial_x \mathbf{U} + \partial_t \mathbf{R}(\mathbf{U}) = \mathbf{0} \ , \tag{14}$$

5

with $\mathbf{U} = \mathbf{F}(\mathbf{Q})$. Additionally, we assume that prescribed functions at both extremes of the domain, $\mathbf{G}_L(t)$ and $\mathbf{G}_R(t)$ are available at any time t. This allows us to build the following problems

$$\begin{aligned} \partial_x \mathbf{U} + \partial_t \mathbf{R}(\mathbf{U}) &= \mathbf{0} , \ x \le x_L , \\ \mathbf{U}(x_L, t) &= \mathbf{F}(\mathbf{G}_L(t)) , \end{aligned} \right\} \tag{15}$$

and

$$\begin{aligned} \partial_x \mathbf{U} + \partial_t \mathbf{R}(\mathbf{U}) &= \mathbf{0} , \ x \ge x_R , \\ \mathbf{U}(x_R, t) &= \mathbf{F}(\mathbf{G}_R(t)) . \end{aligned} \right\} \tag{16}$$

Notice that unlike to (1), the evolution of these problems is carried out for the x variable instead of t. Due to that, systems (15) and (16), are called here *reverse problems*. These problems allow to know approximations of the solution of interior problems at any position x outside computational domain at any time t. Additionally, the involvement of the governing equation of interior problems, incorporates a physical meaning to approximations. Note that these are hyperbolic problems, so a wide range of solution regimes can be expected, in this sense shock waves or discontinuous behaviours of solution can be captured by these reverse problems. Notice that any finite volume scheme of the form

$$\mathbf{U}_{i+1}^n = \mathbf{U}_i^n - \frac{\delta x}{\delta t} [\mathbf{R}_{n+\frac{1}{2}} - \mathbf{R}_{n-\frac{1}{2}}] , \tag{17}$$

with

$$\mathbf{U}_i^n = \tfrac{1}{\delta t} \int_{t^{n-\frac{1}{2}}}^{t^{n+\frac{1}{2}}} \mathbf{U}(x_i, t) dt , \quad \mathbf{R}_{n+\frac{1}{2}} = \tfrac{1}{\delta x} \int_{x_i}^{x_{i+1}} \mathbf{R}(\mathbf{U}(x, t^{n+\frac{1}{2}})) dx , \tag{18}$$

can be employed to solve these reverse problems. Of course, methods for interior problems may be different of those for reverse problems, even order of accuracy of both type of schemes can be different. In section 4, a simple second order method for reverse problems, is presented.

Remark 1. *Notice that in this section, two assumptions have been done; the physical flux $\mathbf{F}(\mathbf{Q})$ is assumed to be invertible; $\mathbf{G}_L(t)$ and $\mathbf{G}_R(t)$ are assumed to be prescribed. However, in general cases these assumptions are not longer valid. In the following sections these issues are concerned.*

The strategy could be considered as a numerical counterpart of the inverse Lax-Wendroff procedure, but, the use of Taylor expansions, the Cauchy-Kowaleski procedure and extrapolations like those in [44], are avoided.

3.2. A strategy to compute inverse functions $\mathbf{R}$ for the physical flux $\mathbf{F}$

Let us consider the physical flux $\mathbf{F}$ and a given $\mathbf{U}$, from the relationship between $\mathbf{F}$ and $\mathbf{U}$, we can find $\mathbf{Q}$ such that

$$\mathbf{U} = \mathbf{F}(\mathbf{Q}) . \tag{19}$$

This can be done by using a fixed point procedure

$$\mathfrak{H}(\mathbf{Q}) = \mathbf{U} - \mathbf{F}(\mathbf{Q}) . \tag{20}$$

6

The Jacobian of $\mathfrak{H}(\mathbf{Q})$ is given by

$$\frac{\partial \mathfrak{H}}{\partial \mathbf{Q}} = -\mathbf{A}(\mathbf{Q}) \, . \tag{21}$$

Then we can generate the iteration process

$$\mathbf{Q}^{l+1} = \mathbf{Q}^l - \delta^l \, , \tag{22}$$

where δ^l solves

$$\delta^l = min_\delta \|\mathbf{A}(\mathbf{Q}^l)\delta - \mathfrak{H}(\mathbf{Q}^l)\| \, , \tag{23}$$

here, l is an iteration index. Notice that $\mathbf{A}$ not require to be invertible. However, if $\mathbf{A}$ is invertible, then δ is uniquely determined by (23) and in addition, we can assume that (22) converges at least locally to some $\mathbf{Q}^*$, that means, $\mathbf{Q}^l \to \mathbf{Q}^*$ then

$$\mathbf{U} = \mathbf{F}(\mathbf{Q}^*) \tag{24}$$

and thus

$$\mathbf{R}(\mathbf{U}) = \mathbf{Q}^* \, . \tag{25}$$

We remark that this procedure can be carried out only if $\mathbf{R}$ cannot be provided analytically.

3.3. Treatment of boundary conditions when only the inflow boundary is prescribed

Here, the aim is to construct $\mathbf{G}_L(t)$ and $\mathbf{G}_R(t)$ in the case in which only the inflow boundary is prescribed. This case has been well described in [44] and the strategy described in section 2.4, can be applied to build $\mathbf{G}_L(t)$ and $\mathbf{G}_R(t)$. Alternatively, in this section we propose a simpler approach which avoid the solution of algebraic equations.

We note that inlet and outlet boundaries are characterized by the sign of eigenvalues of the Jacobian matrix of $\mathbf{F}(\mathbf{Q})$. By following the strategy in [44] we express variables at boundaries in terms of local characteristic variables. To do that, we are going to assume that Jacobian matrix $\mathbf{A} := \partial \mathbf{F}(\mathbf{Q})/\partial \mathbf{Q}$ has a decomposition

$$\mathbf{A} = \mathbf{L}^{-1}\mathbf{\Lambda}\mathbf{L} \, , \tag{26}$$

with $\mathbf{\Lambda} = diag(\lambda_j)$, $j = 1, ..., m$. In addition let us assume that $\lambda_i < \lambda_{i+1}$. Without loss of generality, let us focus on the left boundary so we are going to assume that there exists $m^* \leq m$ such that $\lambda_j(\mathbf{G}_L(t^n)) < 0$, $j = 1, ..., m^*$, whereas, $\lambda_j(\mathbf{G}_L(t^n)) > 0$, $j = m^* + 1, ..., m$. Then a local characteristic transformation can be carried out at boundaries. So a local decomposition can be obtained as follows

$$\mathbf{W}_L(t) = \mathbf{L}(\mathbf{G}_L(t))\mathbf{G}_L(t) \, ,$$

it also implies that eigenvalues can be expressed in terms of local characteristic variables. Similarly, once characteristic variables are available we can recover conserved variables as follows

$$\mathbf{G}_L(t) = \mathbf{L}^{-1}(\mathbf{W}_L(t))\mathbf{W}_L(t) \, . \tag{27}$$

Since $\mathbf{G}_L(t)$ and $\mathbf{G}_R(t)$ are not always prescribed, we replace $\mathbf{G}_L(t_n)$ by $\mathbf{Q}_1^n$ and $\mathbf{G}_R(t_n)$ by $\mathbf{Q}_{N_{int}}^n$. Hence, inflow boundaries on the left correspond to $\lambda_j(\mathbf{Q}_1^n) > 0$ and information at boundary is

required on the corresponding characteristic variables, that means $\mathbf{W}_{L,j}(t)$ needs to be defined for $j > m^*$ because this information is propagated into the computational domain. However, boundary information associated to $\lambda_j < 0$ is not required, because the corresponding information is propagated outside the domain, and it does not influence the numerical solution of interior problems. However, we note that this information is still required for the reconstruction procedure.

In this section we provide a simple interpolation to fill information at boundaries, due to the lack of a prescribed function for the inflow as well as for outflow boundaries. The strategy simply is based on the use of the record of numerical solutions at interior cell near to boundaries. At the time level t^n, we carry out the following first order interpolation in time

$$\mathbf{T}(t) = \mathbf{Q}_1^{n-1} + \frac{t - t^{n-1}}{\Delta t}(\mathbf{Q}_1^n - \mathbf{Q}_1^{n-1}) \, . \tag{28}$$

We start the procedure at t^0, with $\mathbf{Q}_1^{-1} := 3\mathbf{Q}_1^0 - 3\mathbf{Q}_2^0 + \mathbf{Q}_3^0$, which is a third order extrapolation. Then, we transform $\mathbf{T}(t)$ into characteristic variables in a local sense

$$\hat{\mathbf{T}}(t) = \mathbf{L}(\mathbf{Q}_1^n)\mathbf{T}(t) \, , \tag{29}$$

then, for component j where information is not provided, we assign $\mathbf{W}_{L,j}(t) = \hat{\mathbf{T}}_j(t)$. This is normal in the case of outflow, however, it can be employed for inflow boundaries when information is missed. Then the data on the left $\mathbf{G}_L(t)$ is completely defined by following (27) but with $\mathbf{W}_L(t_n)$ replaced by $\mathbf{W}_1^n$, the primitive variable associated to $\mathbf{Q}_1^n$, and thus reverse problems can be solved. The same procedure on the right boundary condition can be carried out, and thus $\mathbf{G}_R(t)$ can be obtained.

4. Numerical solution of reverse problems and ghost cell computations

In this section we propose a second order scheme to solve the reverse problems on the right, in a small domain centred in $t \in [t^n, t^{n+1}]$, the same approach will be valid for reverse problems on the left. This approach is based on the construction of a computational domain based on some few small cells around t such that the second order of accuracy on these small intervals is comparable with the high-order accuracy of the global scheme on coarse meshes. Let us construct a computation domain of $2\bar{M} - 1$ cells. We start by considering $\bar{M}$ barycentre values

$$\tau_{n+\bar{M}} = t + n\delta t \, , \tag{30}$$

with $n = -(\bar{M} - 1), ..., \bar{M} - 1$ and $\delta t = L\Delta t/(2\bar{M} - 1)$, where L is a constant length. Thus we define $2\bar{M} - 1$ cells $[t^{n-\frac{1}{2}}, t^{n+\frac{1}{2}}]$ with $t^{n+\frac{1}{2}} = \tau_n + \frac{\delta t}{2}$. By construction the central cell contains t, whereas, extremes of this domain are given by $t_L = t^{-(\bar{M}-1)-\frac{1}{2}}$ and $t_R = t^{(\bar{M}-1)+\frac{1}{2}}$. We will use a MUSCL type scheme for marching in space up to $x > x_R$ in N iterations, which is carried out by considering

$$\delta x = \frac{(x - x_R)}{N} \, , \tag{31}$$

such that $x = x_R + N\delta x$. The marching in space is achieved by using a one-step finite volume formula, as follows

$$\mathbf{U}_{i+1}^n = \mathbf{U}_i^n - \frac{\delta x}{\delta t}\left[\mathbf{R}_{n+\frac{1}{2}} - \mathbf{R}_{n-\frac{1}{2}}\right] \, , \tag{32}$$

8

with $i = 1, ..., N$ and $n = -(\bar{M} - 1), ..., \bar{M} - 1$ where

$$\mathbf{U}_0^n = \tfrac{1}{\Delta t} \int_{t^{n-\frac{1}{2}}}^{t^{n+\frac{1}{2}}} \mathbf{F}(\mathbf{G}_R(t)) dt \ . \tag{33}$$

The numerical flux is computed by using a MUSCL type scheme which is explained as follows. We start by considering the interpolation polynomial

$$\mathbf{P}^n(t) = \mathbf{U}_i^n + \frac{(t - \tau_n)}{\delta t} \mathbf{\Delta_n} \ , \tag{34}$$

where the slope $\mathbf{\Delta_n}$ is obtained through the MINMOD limiter, which in a component wise, has the form

$$\mathbf{\Delta}_{\mathbf{n}\,j} = \left\{ \begin{array}{ll} 0 \ , if \ , \mathbf{D}_j^{+,n} \mathbf{D}_j^{-,n} \leq 0 \ , \\ \mathbf{D}_j^{-,n} \ , if \ , |\mathbf{D}_j^{-,n}| < |\mathbf{D}_j^{+,n}| \ , \\ \mathbf{D}_j^{+,n} \ , if \ , |\mathbf{D}_j^{+,n}| < |\mathbf{D}_j^{-,n}| \ , \end{array} \right. \tag{35}$$

$j = 1, ..., m$, where $\mathbf{D}^{-,n} := \mathbf{U}_i^n - \mathbf{U}_i^{n-1}$ and $\mathbf{D}^{+,n} := \mathbf{U}_i^{n+1} - \mathbf{U}_i^n$. Then, inside cell $[t^{n-\frac{1}{2}}, t^{n+\frac{1}{2}}]$, extrapolations of the data at both interfaces are obtained as follows

$$\begin{aligned} \bar{\mathbf{U}}_L &:= \mathbf{P}^n(t^{n-\frac{1}{2}}) = \mathbf{U}_i^n - \tfrac{1}{2}\mathbf{\Delta_i} \ , \\ \bar{\mathbf{U}}_R &:= \mathbf{P}^n(t^{n+\frac{1}{2}}) = \mathbf{U}_i^n + \tfrac{1}{2}\mathbf{\Delta_i} \ . \end{aligned} \tag{36}$$

These extrapolated values are evolved as

$$\begin{aligned} \bar{\mathbf{U}}_L^n &= \bar{\mathbf{U}}_L + \frac{\delta x}{2\delta t}(\mathbf{R}(\bar{\mathbf{U}}_L) - \mathbf{R}(\bar{\mathbf{U}}_R)) \ , \\[2mm] \bar{\mathbf{U}}_R^n &= \bar{\mathbf{U}}_R + \frac{\delta x}{2\delta t}(\mathbf{R}(\bar{\mathbf{U}}_L) - \mathbf{R}(\bar{\mathbf{U}}_R)) \ . \end{aligned} \tag{37}$$

Then, to compute the numerical flux values, $\bar{\mathbf{U}}_L^n$ and $\bar{\mathbf{U}}_R^n$ are interacted by using an approximate Riemann solver. Particularly

$$\mathbf{R}_{n+\frac{1}{2}} = \frac{1}{2}\left(\mathbf{R}(\bar{\mathbf{U}}_R^n) + \mathbf{R}(\bar{\mathbf{U}}_L^n)\right) - \frac{\mu_{n+\frac{1}{2}}}{2}\left(\bar{\mathbf{U}}_R^n - \bar{\mathbf{U}}_L^n\right) \ , \tag{38}$$

that means, $\mathbf{R}_{n+\frac{1}{2}}$ is computed by using the Rusanov flux, [41]. Here $\mu_{n+\frac{1}{2}}$ is obtained as

$$\mu_{n+\frac{1}{2}} = max\{\mu(\bar{\mathbf{U}}_L^n), \mu(\bar{\mathbf{U}}_R^n)\} \ , \tag{39}$$

where $\mu(\mathbf{U}) = max_{j=1,...,m}|\mu_j(\mathbf{U})|$, with $\mu_j(\mathbf{U})$ an eigenvalue of the Jacobian of $\mathbf{R}(\mathbf{U})$. On the extremes t_L and t_R we impose boundary conditions, by computing ghost cells through a third order extrapolation as follows

$$\begin{aligned} \mathbf{U}_i^{t_L-1} &= 3\mathbf{U}_i^1 - 3\mathbf{U}_i^2 + \mathbf{U}_i^3 \ , \\ \mathbf{U}_i^{t_R+1} &= 3\mathbf{U}_i^{\bar{M}} - 3\mathbf{U}_i^{\bar{M}-1} + \mathbf{U}_i^{\bar{M}-2} \ . \end{aligned} \tag{40}$$

9

It allows to compute fluxes at boundaries as follows

$$
\begin{aligned}
\mathbf{R}_{-(\bar{M}-1)-\frac{1}{2}} &= \frac{1}{2}\left(\mathbf{R}(\mathbf{U}_i^{t_L-1}) + \mathbf{R}(\mathbf{U}_i^{-(\bar{M}-1)})\right) - \frac{\lambda_{-(\bar{M}-1)+\frac{1}{2}}}{2}\left(\mathbf{U}_i^{-(\bar{M}-1)} - \mathbf{U}_i^{t_L-1}\right), \\
\mathbf{R}_{(\bar{M}-1)+\frac{1}{2}} &= \frac{1}{2}\left(\mathbf{R}(\mathbf{U}_i^{t_R+1}) + \mathbf{R}(\mathbf{U}_i^{(\bar{M}-1)})\right) - \frac{\lambda_{\bar{M}-1+\frac{1}{2}}}{2}\left(\mathbf{U}_i^{t_R+1} - \mathbf{U}_i^{(\bar{M}-1)}\right).
\end{aligned}
\tag{41}
$$

Therefore after N iterations, an approximation at $x > x_R$ is given by

$$
\mathbf{Q}(x,t) \approx \mathbf{R}(\mathbf{U}_N^0),
\tag{42}
$$

so, ghost cells for reconstruction procedure on the right extreme of $[x_L, x_R]$, are recovered as

$$
\mathbf{Q}_{N_{int}+j}^n = \frac{1}{\Delta x}\int_{x_{j+N_{int}-\frac{1}{2}}}^{x_{j+N_{int}+\frac{1}{2}}} \mathbf{Q}(x, t^n)dx,
\tag{43}
$$

with $j = 1, ..., M$. The same procedure applies for ghost cells on the left of the domain $[x_L, x_R]$.

Remark 2. *To evaluate the integral (43) we use a quadrature rule of three points.*

Figure 1 shows the sketch for $\bar{M} = N = 3$. In this case the solution is found in three steps. In the first iteration the boundary influences the first and last cell, then in the second iteration, the first and last cells propagate their information to the neighbours inside the computational domain and thus this process continues in the third step where the process finishes and the influences of boundaries do not reach the center cell.

Concerning the stability of numerical schemes, we know that for interior problems the Courant-Friedrich-Levita (CFL) condition imposes

$$
\frac{\Delta t}{\Delta x} = \frac{c}{\bar{\lambda}},
\tag{44}
$$

with $c \le 1$ the CFL coefficient and $\bar{\lambda} = max_i(\lambda(\mathbf{Q}_i^n))$ for $i = 1, ..., N_{int}$, where $\lambda(\mathbf{Q}) = max_{j=1,...,m}|\lambda_j(\mathbf{Q})|$ with $\lambda_j(\mathbf{Q})$ eigenvalues of the Jacobian of $\mathbf{F}(\mathbf{Q})$ with respect to $\mathbf{Q}$. Similarly for reverse problems, stability in terms of the CFL condition imposes

$$
\frac{\delta x}{\delta t} = \frac{c}{\bar{\mu}},
\tag{45}
$$

with $\bar{\mu} = max_n(\mu(\mathbf{U}_i^n))$, $n = 1, ..., 2\bar{M} - 1$ and $\mu(\mathbf{U}) = max_{j=1,...,m}|\mu_j(\mathbf{U})|$, with $\mu_j(\mathbf{U})$ eigenvalues of the Jacobian of $\mathbf{R}(\mathbf{U})$.

Proposition 4.1. *Let M be the order of accuracy of schemes for interior problems. A necessarily condition to obtain stable numerical schemes for reverse problems is that*

$$
\eta := \frac{\bar{M}}{NL} \le c^2,
$$

with $L \le M$.

10

Proof. Now, we observe that $\delta t = \frac{L\Delta t}{2\bar{M}-1}$ and $\delta x = \frac{K\Delta x}{N}$, with $L \leq M$ and K is the distance of x with respect to $[x_L, x_R]$. Therefore

$$
\begin{aligned}
\frac{c}{\bar{\mu}} &= \frac{\delta x}{\delta t} = \frac{\Delta x}{\Delta t}\frac{K(2\bar{M}-1)}{LN} \, , \\
\frac{c}{\bar{\mu}} &= \frac{\bar{\lambda}}{c}\frac{K}{L}\frac{2\bar{M}-1}{N} \, ,
\end{aligned}
\tag{46}
$$

after some manipulations

$$
\frac{c^2}{\bar{\mu}\bar{\lambda}} = (\frac{K}{L})\frac{2\bar{M}-1}{N} \, .
\tag{47}
$$

On the other hand, let us assume that all eigenvalues $\mathbf{A}(\mathbf{Q}) = \partial\mathbf{F}(\mathbf{Q})/\partial\mathbf{Q}$ are distinct from zero. Then

$$
\partial_t\mathbf{Q} + \mathbf{A}(\mathbf{Q})\partial_x\mathbf{Q} = \mathbf{0}
\tag{48}
$$

and

$$
\partial_x\mathbf{Q} + \mathbf{A}(\mathbf{Q})^{-1}\partial_t\mathbf{Q} = \mathbf{0} \, ,
\tag{49}
$$

thus we have

$$
\mathbf{A}(\mathbf{Q})^{-1} = \partial\mathbf{R}(\mathbf{U})/\partial\mathbf{U} \, .
\tag{50}
$$

Now, let us define $\underline{\lambda} = min_j|\lambda_j(\mathbf{Q})| > 0$. Then

$$
1 \leq \frac{\bar{\lambda}}{\underline{\lambda}} = \bar{\lambda}\bar{\mu} \, .
\tag{51}
$$

Therefore

$$
c^2 \geq \frac{c^2}{\bar{\mu}\bar{\lambda}} = (\frac{K}{L})\frac{2\bar{M}-1}{N} \geq \frac{\bar{M}}{NL} \, .
\tag{52}
$$

Thus the result holds. $\qquad\square$

Remark 3. *Reverse problems are hyperbolic conservation laws. Hence, any high-order numerical scheme may be applied. However, high-order schemes reduce dramatically the efficiency of global solvers. As we will see later, in proposition 5.1 and numerical results in section 6, for numerical implementations second order of accuracy for reverse problems should be enough to get the accuracy for interior problems.*

5. Some theoretical results

In this section we are going to present some theoretical results for the scalar case

$$
\left.
\begin{aligned}
\partial_t q(x,t) + \partial_x f(q(x,t)) &= 0 \, , x \in [x_L, x_R] \, , \\
q(x,0) &= h(x) \, , \\
q(x_L,t) &= g_L(t) \, , \\
q(x_R,t) &= g_R(t) \, .
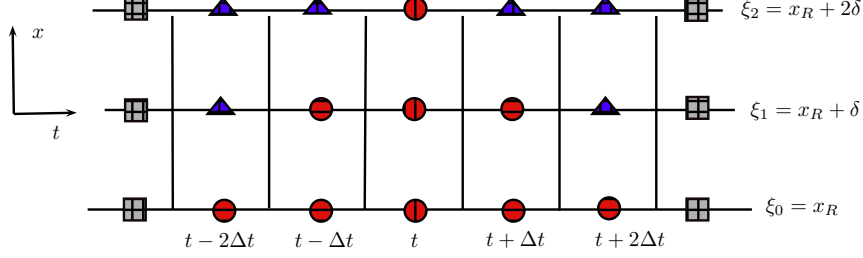\end{aligned}
\right\}
\tag{53}
$$

Figure 1: Sketch of local evolution around t for three cells $\bar{M} = N = 3$. Squares reperesent the ghost cells, circles and triangles are interior points evolved with a second order scheme. Solution is reached in $N = 3$ iterations. Triangles are interior points which at each evolution, are influencied for the information coming from boundaries. Circles are the interior cells which only depend on the interior point.

The reverse problems (15) and (16) associated to (53) are formed by the governing equation

$$\partial_x u(x,t) + \partial_t R(u(x,t)) = 0 \, , \tag{54}$$

where $R(u) = q$. Here, we are going to prove that the present methodology preserves the theoretical order of accuracy for interior problems and for the case in which the physical flux contains an inverse function, the solution of the reverse problem can be tracked from the interior problem through the characteristic curves. In what follows, we will say *reverse boundary condition* to indicate that Dirichelt boundary conditions are included through the solution of a GRP at the extremes of the computational domain and the solution of reverse problems is used to provide the ghost cells in order to fill the stencils and thus to carry out the reconstruction procedure.

To start, let us assume that problem associated to (53) has the exact solution $\bar{q}(x,t)$ and problem associated to (54) has the exact solution $\bar{u}(x,t)$. Additionally, we are going to assume that the numerical solutions for the interior problem are obtained with a numerical scheme of accuracy r in space and time. Let us assume that the scheme for solving (53) can be written as

$$q_i^{n+1} = \sum_{j=-k_L}^{k_R} b_j(q_{i+j-k_L}^n, .., q_{i+j}^n, ..., q_{i+j+k_R}^n) q_{i+j}^n \, , \tag{55}$$

where b_j is a function of the data. These functions can be non-linear. Then if we adopt the definition of accuracy as in [4], (definition 2.21). The scheme applied to the exact solution, at least for cells inside the computational domain satisfies

$$\bar{q}(x_i, t^{n+1}) - \sum_{j=-k_L}^{k_R} b_j(\bar{q}(x_{i+j-k_L}, t^n), ..., \bar{q}(x_{i+j+k_R}, t^n)) \bar{q}(x_{i+j}, t^n) = O(\Delta t^{r+1}) \, . \tag{56}$$

As the time step Δt and the mesh size Δx are related through a CFL condition, the accuracy is simultaneously the same in space and time. So, the aim here is to probe that (56) is also valid for values near to boundaries. This is proved in the following proposition.

Proposition 5.1. *Let r be the order of accuracy of the numerical scheme for the interior problem. If reverse problems are solved with a numerical scheme of order of accuracy p. Then the*

12

numerical scheme for the interior problem maintains the expected theoretical order of accuracy even at boundaries. If

$$N \geq \Delta x^{\frac{p-r}{p+1}} r , \tag{57}$$

where N is defined in proposition 4.1.

Proof. Without loss of generality, we are going to prove that (56) is attained for the first left computational cell. Then, in order to take into account the way in which the data outside the computational domain is obtained, we write (55) as

$$q_1^{n+1} = \sum_{j=-k_L}^{k_R} b_j (D(1 + j - k_L, q), .., D(0, q), ..., D(1 + j + k_R, q)) \times$$
$$D(1 + j, q) , \tag{58}$$

with

$$D(i, q) = \begin{cases} R(P_i^n(t^n)) , i < 1 , \\ q_i^n , i \geq 1 . \end{cases} \tag{59}$$

Here, $R(P_i^n(t^n))$ denotes the solution of the reverse problem at t^n, obtained as in (42) and $P_i^n(t)$ denotes the reconstruction polynomial (34). On the other hand, as the reverse problem is solved with a numerical scheme of accuracy p, in which δt and δx are related through a CFL condition, we have that the numerical and exact solutions, u_i^n and $\bar{u}(x_i, t^n)$ respectively, are related as follows

$$u_i^n = \bar{u}(x_i, t^n) + O(\delta x^{p+1}) + O(\delta t^{p+1}) = \bar{u}(x_i, t^n) + O(\delta t^{p+1}) . \tag{60}$$

Additionally, we also assume that the reconstruction polynomial is of order p in the following sense

$$P_i^n(t^n) = u_i^n + O(\delta t^{p+1}) . \tag{61}$$

Then from (60) one has

$$P_i^n(t^n) = u_i^n + O(\delta t^{p+1}) = \bar{u}(x_i, t^n) + O(\delta t^{p+1}) . \tag{62}$$

Thus, we can write

$$R(P_i^n(t^n)) = \bar{q}(x_i, t^n) + O(\delta t^{p+1}) R'(P_i^n(t^n)) + O(\delta t^{2(p+1)})$$
$$= \bar{q}(x_i, t^n) + O(\delta t^{p+1}) . \tag{63}$$

Therefore, from (59) and (63), the numerical scheme applied to the exact solution $\bar{q}(x, t)$ provides

$$\bar{q}(x_1, t^{n+1}) = \sum_{j=-k_L}^{-1} b_j (\bar{q}(x_{1+j-k_L}, t^n) + O(\delta t^{p+1}), ..., \bar{q}(x_{1+j+k_R}, t^n) + O(\delta t^{p+1})) \times$$
$$(\bar{q}(x_{1+j}, t^n) + O(\delta t^{p+1}))$$
$$+ \sum_{j=0}^{k_R} b_j (D(1 + j - k_L, \bar{q}), .., D(0, \bar{q}), ..., \bar{q}(x_{1+j+k_R}, t^n)) \bar{q}(x_{1+j}, t^n)$$
$$= \sum_{j=-k_L}^{k_R} b_j (\bar{q}(x_{1+j-k_L}, t^n), ..., \bar{q}(x_{1+j+k_R}, t^n)) \bar{q}(x_{1+j}, t^n) + O(\delta t^{p+1}) \Phi ,$$

13

with

$$\Phi = (\sum_{j=-k_L}^{k_R} b_j + \sum_{k=1+j<1} b_{j,k} O(\delta t^{p+1})) , \tag{64}$$

where $b_{j,k} = \partial b_j / \partial q_k$. For simplicity we have dropped the arguments of b_j and its derivatives $b_{j,k}$. As we are free of choosing δt, we take, δt such that, $O(\delta t^{p+1})\Phi = O(\Delta t^{r+1})$. In virtue of proposition 4.1, we note that

$$O(\delta x) = O(\tfrac{K}{N}\Delta x) = O(\Delta x^{\frac{r+1}{p+1}}) , \tag{65}$$

hence, $O(N) = O(\Delta x^{\frac{p-r}{p+1}} K)$ with K defined given in Proposition 4.1 and thus it is related with the required stencils, which for order r is exactly $K = r$. So, by using the meaning of $O(\cdot)$, the result holds.

$\square$

Remark 4. *Notice that the previous result does not consider a particular scheme. We note that for values of $\Delta x < 1$, it is required $p \leq r$ to obtain a feasible disctretization of reverse problems, that means, $N > 1$.*

Now, let us see how are related the solution of the interior problem and the solution of reverse problems. For that we need f to have an inverse function which is ensured in the scalar case by the following.

Proposition 5.2. *Let $f(q)$ be a differentiable function in $\mathbb{R}$ with $f(q)' \neq 0$, then there exists R such that $R(f(q)) = q$.*

Proof. The result follows from the inverse function Theorem, see [32] and references therein.

$\square$

Now we are going to prove that if the flux $f(q)$ contains an inverse function f^{-1}, the solution of reverse problems can be tracked from the interior problem through characteristic curves. To start let us consider the following.

Definition 5.3. *Let $w(x,t)$ be the exact solution of*

$$\left.\begin{array}{rcll} \partial_t q + \partial_x f(q(x)) & = & 0 , & x \in \mathbb{R} , \\ q(x,0) & = & h(x) . \end{array}\right\} \tag{66}$$

We say that boundary conditions in (53) are compatible if $g_L(t) = w(x_L, t)$ and $g_R(t) = w(x_R, t)$.

Now we are going to prove that the solution of the reverse problems under compatibility conditions, are contained in the characteristic curves of the interior problems. It is carried out in the following.

Proposition 5.4. *Let $w(x,t)$ be the exact solution of (66). Let $\bar{q}(x,t)$ be the exact solution of problem (53) endowed with compatible boundary conditions, where $g_L(t)$ and $g_R(t)$ are prescribed functions at boundaries and let $\bar{u}(x,t)$ be the exact solution of the reverse problems. If $R \equiv f^{-1}$. Then $\bar{q}(x,t) = w(x,t)$ in $[x_L, x_R]$ and $f^{-1}(\bar{u}(x,t)) = w(x,t)$ for all $x \leq x_L$ and $x \geq x_R$.*

Proof. Without loss of generality we are going to consider the left reverse problem, defined for $x \le x_L$ and let us denote by $\bar{u}(x, t)$ its solution. We note that on the curve $t_1(x)$ which is defined by

$$\left.\begin{aligned}\frac{dt_1(x)}{dx} &= (f^{-1}(\bar{u}(x, t_1(x))))' \, , \\ t_1(x_L) &= \delta \, ,\end{aligned}\right\}\tag{67}$$

the following holds

$$\frac{d\bar{u}(x, t_1(x))}{dx} = 0 \, .\tag{68}$$

On the other hand, from the exact solution $w(x, t)$ we define the curve $t_2(x)$ given by

$$\left.\begin{aligned}\frac{dt_2(x)}{dx} &= (f'(w(x, t_2(x)))^{-1} \, , \\ t_2(x_L) &= \delta \, ,\end{aligned}\right\}\tag{69}$$

along which the following is satisfied

$$\frac{df(w(x, t_2(x)))}{dx} = 0 \, .\tag{70}$$

On the other hand, by following these curves and from the compatibility condition

$$\bar{u}(x, t_1(x)) = \bar{u}(x_L, t_1(x_L)) = f(g_L(\delta)) = f(g_L(t_2(x_L))) = f(w(x_L, t_2(x_L))) = f(w(x, t_2(x))) \, .\tag{71}$$

From this equality we have

$$\frac{dt_1(x)}{dx} = (f^{-1}(\bar{u}(x, t_1(x))))' = (f'(w(x, t_2(x))))^{-1} = \frac{dt_2(x)}{dx} \, ,\tag{72}$$

this means that $t_1(x) - t_2(x) = t_1(x_L) - t_2(x_L) = 0$, then both curves coincide, which allows to define $t(x) := t_1(x) = t_2(x)$. Thus for $x < x_L$ we have $f^{-1}(\bar{u}(x, t)) = w(x, t)$.

On the other hand, inside the computational domain $[x_L, x_R]$ let us define the following curves $x_1(t)$ and $x_2(t)$, defined by the ODE's

$$\left.\begin{aligned}\frac{dx_1(x)}{dt} &= f'(\bar{q}(x_1(t), t) \, , \\ x_1(0) &= y \, ,\end{aligned}\right\}\tag{73}$$

and

$$\left.\begin{aligned}\frac{dx_2(x)}{dt} &= f'(w(x_2(t), t) \, , \\ x_2(0) &= y \, .\end{aligned}\right\}\tag{74}$$

These are the respective characteristic curves, along which $\bar{q}(x, t)$ and $w(x, t)$ remain constant. This yields

$$\bar{q}(x_1(t), t) = \bar{q}(x_1(0), 0) = h(y) = w(x_2(0), 0) = w(x_2(t), t) \, .\tag{75}$$

From (73), (74) and (75), we obtain $x(t) := x_1(t) = x_2(t)$. Therefore in $[x_L, x_R]$, $\bar{q}(x, t) = w(x, t)$ and thus the result holds. □

In the following section we will solve interior problems in which the present methodology for Dirichlet boundary condition is implemented.

15

6. Numerical results

In all tests, the ADER-DET method is implemented to solve interior problems and the Rusanov solver [41], is used to solve the Riemann problems. See [15] for further details.

6.1. Linear advection

Let us consider the interior problem given by the linear advection equation

$$\left.\begin{array}{rcl} \partial_t q + \partial_x(\lambda q) & = & 0 \ , x \in [0,1], t \in (0,T] \ , \\ q(x,0) & = & sin(2\pi x) \ . \end{array}\right\} \tag{76}$$

In this case we note that the left boundary is an inflow boundary where a prescribed function $g_L(t)$ is assumed to be available, whereas, the right boundary is an outflow where no boundary is required. However, in order to apply our methodology we construct the function $g_R(t)$ as suggested in section 3.3, such that $q(1,t) = g_R(t)$. If $g_L(t) = -sin(2\pi\lambda t)$, the exact solution is given by $q(x,t) = sin(2\pi(x - \lambda t))$. Therefore, reverse problems are easily obtained, so the left reverse problem is given by

$$\left.\begin{array}{rcl} \partial_x u + \partial_t(\lambda^{-1} u) & = & 0 \ , x < x_L \ , \\ u(t, x_L) & = & -\lambda \sin(2\pi\lambda t) \ , \end{array}\right\} \tag{77}$$

whereas, right reverse problem is given by

$$\left.\begin{array}{rcl} \partial_x u + \partial_t(\lambda^{-1} u) & = & 0 \ , x > x_R \ , \\ u(t, x_R) & = & u_{N_{int}}^{n-1} + \frac{t-t^n}{\Delta t}(u_{N_{int}}^n - u_{N_{int}}^{n-1}) \ . \end{array}\right\} \tag{78}$$

Extrapolations like (40) are applied to deal with boundary conditions for reverse problems. Table 1, shows the results of a systematically convergence rates assessment for the interior problem with $\lambda = 1$, CFL equals to 0.9 and $t_{out} = 4$. For reverse problems we have used $N = 20$, $\bar{M} = 10$ and $L = 0.7$, which provides $\eta = \bar{M}/NL = 0.71 < c^2 = (0.9)^2 = 0.81$, thus from the Proposition 4.1, a stable scheme is expected. We observe that the accuracy is achieved up to fifth order of accuracy. In order to compare the performance of the present procedure, we compare the corresponding CPU time against the CPU time of the inverse Lax-Wendroff procedure. Table 2, shows the CPU time comparison between the present method and inverse Lax-Wendroff procedure. We observe for second, third and fourth orders of accuracy the performance of both boundary treatments is similar. However, for fifth order the CPU of both procedures have same magnitude.

6.2. Hyperbolic system with a non-invertible Jacobian

In this section we deal with the issue of a non-invertible Jacobian matrix. To construct this test, let us consider the scalar case

$$\partial_t q(x,t) + \partial_x(a(x)q(x,t)) = 0 \ , x \in [0,1] \ , \tag{79}$$

as we note the physical flux depends on x, in such a case Riemann problem can non be solved as conventional and the accuracy can be penalized but also the stability of numerical schemes. See [60, 61] for further details on space-dependent fluxes. To overcome any difficulty arising from space-dependent fluxes, we can transform (79) into the following system

$$\partial_t \mathbf{Q} + \partial_x \mathbf{F}(\mathbf{Q}) = \mathbf{0} \ , \tag{80}$$

Theoretical order : 2

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$6.98e-02$	0.00	$3.88e-02$	0.00	$4.30e-02$	0.0240
16	1.21	$3.02e-02$	1.96	$10.00e-03$	1.72	$1.30e-02$	0.0520
32	1.25	$1.27e-02$	1.92	$2.64e-03$	1.66	$4.13e-03$	0.1320
64	1.25	$5.33e-03$	2.07	$6.28e-04$	1.68	$1.29e-03$	0.4080
128	1.29	$2.19e-03$	2.14	$1.42e-04$	1.75	$3.82e-04$	1.0880

Theoretical order : 3

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$2.01e-02$	0.00	$9.88e-03$	0.00	$1.14e-02$	0.0160
16	3.27	$2.09e-03$	3.20	$1.08e-03$	3.20	$1.24e-03$	0.0400
32	2.53	$3.61e-04$	2.82	$1.52e-04$	2.76	$1.83e-04$	0.1160
64	2.62	$5.85e-05$	2.96	$1.96e-05$	2.93	$2.41e-05$	0.3160
128	3.53	$5.06e-06$	3.11	$2.26e-06$	3.14	$2.74e-06$	1.1200

Theoretical order : 4

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$2.23e-02$	0.00	$8.64e-03$	0.00	$1.14e-02$	0.0200
16	5.41	$5.22e-04$	5.21	$2.34e-04$	5.29	$2.91e-04$	0.0520
32	3.74	$3.90e-05$	4.69	$9.03e-06$	4.53	$1.26e-05$	0.1280
64	4.11	$2.26e-06$	3.81	$6.42e-07$	3.92	$8.34e-07$	0.3920
128	3.93	$1.49e-07$	3.74	$4.80e-08$	3.77	$6.10e-08$	1.3400

Theoretical order : 5

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$9.49e-03$	0.00	$4.87e-03$	0.00	$5.38e-03$	0.0920
16	5.95	$1.53e-04$	5.83	$8.55e-05$	5.83	$9.45e-05$	0.1880
32	4.80	$5.50e-06$	5.16	$2.39e-06$	5.15	$2.67e-06$	0.4200
64	4.84	$1.93e-07$	5.33	$5.92e-08$	5.22	$7.16e-08$	1.0480
128	5.12	$5.54e-09$	5.38	$1.42e-09$	5.36	$1.74e-09$	3.1440

Table 1: Convergence rates for the linear advection at output time $t_{out} = 4$ with $C_{cfl} = 0.9$, $\lambda = 1$, $N = 20$, $\bar{M} = 10$ and $L = 0.7$. Left reverse boundary conditions are applied. Inflow boundary on the left boundary and outflow boundary on the right boundary.

Theoretical order : 2

N	CPU reverse problem	CPU inverse Lax-Wendroff
8	0.0240	0.020
16	0.0520	0.064
32	0.1320	0.240
64	0.4080	0.244
128	1.0880	0.756

Theoretical order : 3

N	CPU reverse problem	CPU inverse Lax-Wendroff
8	0.016	0.052
16	0.0400	0.056
32	0.1160	0.060
64	0.3160	0.244
128	1.1200	0.908

Theoretical order : 4

N	CPU reverse problem	CPU inverse Lax-Wendroff
8	0.0200	0.080
16	0.0520	0.020
32	0.1280	0.088
64	0.3920	0.320
128	1.3400	1.352

Theoretical order : 5

N	CPU reverse problem	CPU inverse Lax-Wendroff
8	0.0920	0.012
16	0.1880	0.040
32	0.4200	0.172
64	1.0480	0.540
128	3.1440	2.084

Table 2: Linear advection. CPU time comparisons by orders of accuracy. Reverse problems (second columns) and inverse Lax-Wendroff procedure (third columns).

where

$$\mathbf{Q} = \begin{bmatrix} q \\ a \end{bmatrix} , \mathbf{F}(\mathbf{Q}) = \begin{bmatrix} aq \\ 0 \end{bmatrix} . \tag{81}$$

The Jacobian of $\mathbf{F}(\mathbf{Q})$ is given by

$$\mathbf{A} = \begin{bmatrix} a & q \\ 0 & 0 \end{bmatrix} \tag{82}$$

and eigenvalues of $\mathbf{A}(\mathbf{Q})$ are $\lambda_1 = 0$ and $\lambda_2 = a$.

We can construct an exact solution for (79) and thus to (80). This can be done by inserting in (79), the functions

$$\begin{aligned} \hat{q}(x,t) &= sin(\tfrac{x}{t+1}) , \\ \hat{a}(x) &= exp(-50(x-0.5)^2) , \end{aligned} \tag{83}$$

which produces a new equation

$$\partial_t q(x,t) + \partial_x(a(x)q(x,t)) = s(x,t) , \tag{84}$$

with $s(x,t) = \hat{q}_t + \hat{a}'\hat{q} + \hat{a}\hat{q}_x$. Clearly (83) solves (84) and of course, also solves

$$\partial_t \mathbf{Q} + \partial_x \mathbf{F}(\mathbf{Q}) = \mathbf{S}(x,t) , \tag{85}$$

with $\mathbf{S} = [s(x,t), 0]^T$. We apply reverse problems to system (80) thus we need a function $\mathbf{R}(\mathbf{U})$ such that

$$\mathbf{R}(\mathbf{U}) = \mathbf{Q} .$$

Notice that in this case, vectors $\mathbf{U}$ have the form $\mathbf{U} = [u, 0]^T$. Therefore an inverse function can be given by

$$\mathbf{R}(\mathbf{U}) = \begin{bmatrix} u/k \\ k \end{bmatrix} , \tag{86}$$

with k a constant value. It can be verified that $\mathbf{F}(\mathbf{R}(\mathbf{U})) = \mathbf{U}$. Therefore the reverse problem on the left has the form

$$\begin{aligned} \partial_x \mathbf{U} + \partial_x \mathbf{R}(\mathbf{U}) &= \mathbf{0} , x < x_L , \\ \mathbf{U}(0,t) &= \mathbf{F}(\hat{\mathbf{Q}}(0,t)) , \end{aligned} \tag{87}$$

with

$$\hat{Q}(x,t) = \begin{bmatrix} \hat{q}(x,t) \\ \hat{a}(x) \end{bmatrix} , \tag{88}$$

whereas, the reverse problem on the right is given by

$$\begin{aligned} \partial_x \mathbf{U} + \partial_x \mathbf{R}(\mathbf{U}) &= \mathbf{0} , x > x_R , \\ \mathbf{U}(1,t) &= \mathbf{F}(\hat{\mathbf{Q}}(1,t)) . \end{aligned} \tag{89}$$
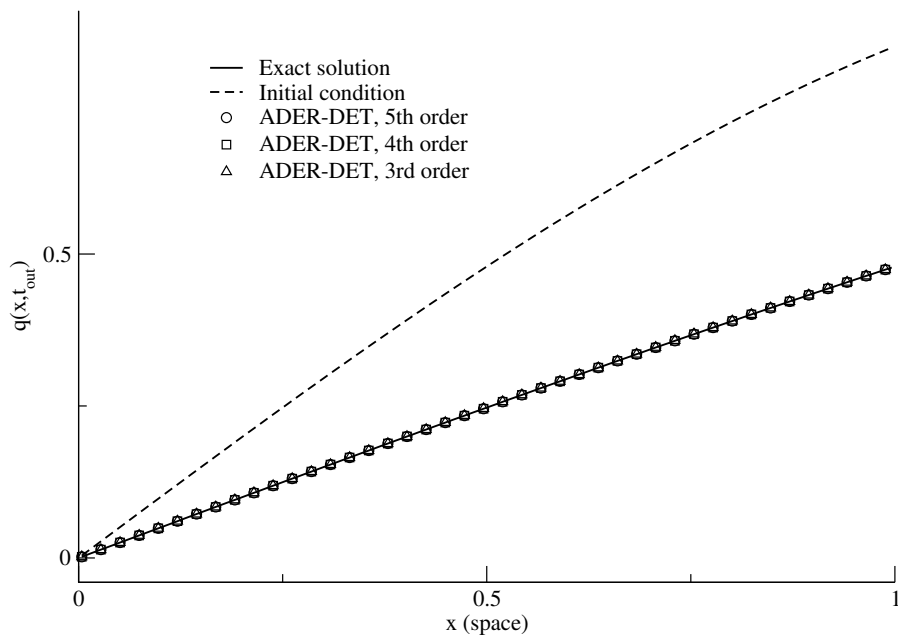
19

Figure 2: Non-invertible Jacobian matrix. Initial condition (Dashed line), exact solution (full line), third order solution (triangles), fourht order solution (squares) and fifth order solution (circles). Parameters: $N = \bar{M} = 3$, $L = 2$, $k = 1$, 128 cells, $t_{out} = 1$ and $C_{cfl} = 0.9$

Theoretical order : 2

Mesh	L_∞ - err	L_∞ - ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$2.10e-01$	0.00	$1.12e-01$	0.00	$1.31e-01$	0.0080
16	0.88	$1.13e-01$	1.64	$3.58e-02$	1.41	$4.94e-02$	0.0200
32	1.01	$5.64e-02$	1.72	$1.09e-02$	1.55	$1.68e-02$	0.0520
64	2.49	$1.00e-02$	2.65	$1.73e-03$	2.62	$2.73e-03$	0.1560
128	2.15	$2.25e-03$	2.62	$2.80e-04$	2.50	$4.83e-04$	0.5000

Theoretical order : 3

Mesh	L_∞ - err	L_∞ - ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$5.80e-01$	0.00	$1.46e-01$	0.00	$2.25e-01$	0.0040
16	3.37	$5.60e-02$	2.83	$2.06e-02$	3.00	$2.82e-02$	0.0200
32	0.90	$3.01e-02$	1.94	$5.36e-03$	1.71	$8.62e-03$	0.0680
64	2.93	$3.96e-03$	2.84	$7.50e-04$	2.84	$1.21e-03$	0.1800
128	2.84	$5.51e-04$	2.96	$9.65e-05$	2.94	$1.57e-04$	0.6520

Theoretical order : 4

Mesh	L_∞ - err	L_∞ - ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$1.08e-00$	0.00	$2.58e-01$	0.00	$4.10e-01$	0.0080
16	0.03	$1.06e-00$	0.95	$1.34e-01$	0.48	$2.94e-01$	0.0320
32	6.13	$1.51e-02$	5.26	$3.48e-03$	5.80	$5.26e-03$	0.0840
64	4.60	$6.22e-04$	4.61	$1.43e-04$	4.63	$2.13e-04$	0.2880
128	4.29	$3.18e-05$	4.72	$5.40e-06$	4.71	$8.11e-06$	1.0600

Theoretical order : 5

Mesh	L_∞ - err	L_∞ - ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$1.14e-00$	0.00	$2.10e-01$	0.00	$4.16e-01$	0.0360
16	4.63	$4.61e-02$	4.31	$1.06e-02$	4.74	$1.55e-02$	0.0960
32	4.54	$1.98e-03$	4.35	$5.17e-04$	4.35	$7.60e-04$	0.2320
64	4.61	$8.11e-05$	4.91	$1.72e-05$	4.83	$2.68e-05$	0.6840
128	5.05	$2.46e-06$	4.94	$5.58e-07$	4.94	$8.70e-07$	2.2080

Table 3: Convergence rates for space-dependent flux at output time $t_{out} = 1$ with $C_{cfl} = 0.9$ and $k = 1$ for the inverse function $\mathbf{R(U)}$. Reverse boundary conditions are applied with $N = 3$, $\bar{M} = 3$ and $L = 2$.

Table 3, shows the result of the convergence rate assessment by using reverse problems for boundary conditions. We have used the combination $N = \bar{M} = 3$ and $L = 2$, which gives $\eta = 0.5$ and so in virtue of proposition 4.1, it produces a globally stable scheme. Notice that we use the term *globally stable scheme* to say that schemes for interior problems as well as that for reverse problems are stable.

We observe that expected orders of accuracy are achieved up to fifth order of accuracy. As we note there are infinite numbers of inverse functions for flux $\mathbf{F}$, for each k $\mathbf{R}(\mathbf{U})$ is a suitable inverse function. However, we can check that by following the procedure in section 3.2, the starting guess $\mathbf{Q}^0 = [q_1, q_2]^T$ yields the converged state $\mathbf{Q}^* = [q_1, 1]^T$, hence the choice of $k = 1$ in (86) agrees with the expected result in section 3.2, due to that this is set for the simulations.

6.3. Linear system model

In this test, the interior problem is given by the linear system of conservation laws

$$
\begin{aligned}
\partial_t \mathbf{Q} + \partial_x \mathbf{F}(\mathbf{Q}) &= 0 , \quad x \in [0, 1] , \\
\mathbf{Q}(x, 0) &= \mathbf{H}_0(x) , \\
\mathbf{Q}(0, t) &= \mathbf{G}_L(x) , \\
\mathbf{Q}(1, t) &= \mathbf{G}_R(x) ,
\end{aligned}
\tag{90}
$$

with

$$
\mathbf{F}(\mathbf{Q}) = \begin{bmatrix} q_1 - q_2 \\ 2q_2 \end{bmatrix} , \quad \mathbf{H}_0(x) = \begin{bmatrix} sin(2\pi x) \\ cos(2\pi x) \end{bmatrix} .
\tag{91}
$$

The exact solution is

$$
\mathbf{Q}^e(x, t) = \begin{bmatrix} sin(2\pi(x - t)) + cos(2\pi(x - t)) - 2cos(2\pi(x - 2t)) \\ 4cos(2\pi(x - 2t)) \end{bmatrix} ,
\tag{92}
$$

so, similarly to the scalar linear case $\mathbf{G}_L(t)$ and $\mathbf{G}_R(t)$ are given by the exact solution evaluated at $x = 0$ and $x = 1$, respectively.

In this case the reverse problems are directly obtained. Without loss of generality, the left reverse problem is given by

$$
\left.
\begin{aligned}
\partial_x \mathbf{U} + \partial_t(\mathbf{R}(\mathbf{U})) &= 0 , x < x_L , \\
\mathbf{U}(t, x_L) &= \mathbf{F}(\mathbf{G}_L(t)) ,
\end{aligned}
\right\}
\tag{93}
$$

where extrapolated boundary conditions are applied on the extremes of the computational domain. Here

$$
\mathbf{U} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} , \quad \mathbf{R}(\mathbf{U}) = \begin{bmatrix} u_1 + \frac{u_2}{2} \\ \frac{u_2}{2} \end{bmatrix} .
\tag{94}
$$

Table 4 shows the error and the result of the empirical convergence rate assessment for the interior problem with the combination $N = 50$, $\bar{M} = 20$, $L = 2$, which produces $\eta = 0.2$, so a globally stable method should be obtained. Additionally, we use $t_{out} = 1$ and $CFL = 0.9$. We observe that expected order of accuracy is achieved up to fifth order of accuracy.

Theoretical order : 2

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$5.73e-01$	0.00	$2.82e-01$	0.00	$3.36e-01$	0.5360
16	1.24	$2.43e-01$	1.73	$8.48e-02$	1.55	$1.14e-01$	0.5240
32	2.65	$3.87e-02$	2.05	$2.04e-02$	2.31	$2.31e-02$	1.0120
64	1.80	$1.12e-02$	2.01	$5.06e-03$	2.05	$5.58e-03$	2.1000
128	2.16	$2.49e-03$	2.04	$1.23e-03$	2.03	$1.36e-03$	4.3240

Theoretical order : 3

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$3.46e-01$	0.00	$1.64e-01$	0.00	$2.00e-01$	0.2600
16	2.45	$6.33e-02$	2.42	$3.07e-02$	2.46	$3.63e-02$	0.3400
32	2.93	$8.28e-03$	2.88	$4.16e-03$	2.91	$4.81e-03$	0.6920
64	3.01	$1.03e-03$	3.00	$5.21e-04$	3.00	$6.00e-04$	1.4120
128	3.01	$1.28e-04$	3.01	$6.48e-05$	3.01	$7.45e-05$	3.0720

Theoretical order : 4

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$5.71e-01$	0.00	$2.97e-01$	0.00	$3.41e-01$	0.2560
16	4.46	$2.60e-02$	4.40	$1.40e-02$	4.38	$1.64e-02$	0.5400
32	4.72	$9.84e-04$	4.77	$5.14e-04$	4.81	$5.84e-04$	1.0440
64	3.83	$6.92e-05$	4.18	$2.84e-05$	4.20	$3.17e-05$	2.1600
128	3.82	$4.90e-06$	3.93	$1.87e-06$	3.91	$2.10e-06$	4.7560

Theoretical order : 5

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
8	0.00	$2.33e-01$	0.00	$1.24e-01$	0.00	$1.44e-01$	6.8160
16	5.17	$6.47e-03$	5.07	$3.69e-03$	5.08	$4.28e-03$	13.3720
32	5.07	$1.92e-04$	4.98	$1.17e-04$	5.02	$1.32e-04$	26.9999
64	4.97	$6.13e-06$	4.96	$3.75e-06$	4.97	$4.22e-06$	54.7440
128	4.97	$1.96e-07$	4.97	$1.20e-07$	4.97	$1.34e-07$	110.7640

Table 4: Convergence rates for the linear system at output time $t_{out} = 1$ with $C_{cfl} = 0.9$. Reverse boundary conditions are applied with $N = 50$, $\bar{M} = 10$ and $L = 2$.

6.4. The Euler equations

Now let us consider an interior problem in which the problem is non-linear, so we consider the Euler equations, characterized by

$$\mathbf{Q} = \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix} , \quad \mathbf{F}(\mathbf{Q}) = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(E + p) \end{bmatrix} . \tag{95}$$

Here, the pressure p is related with the conserved variables through the equation for an ideal gas with $\gamma = 1.4$, which yields

$$p = (\gamma - 1)(E - \frac{\rho u^2}{2}) . \tag{96}$$

The initial condition for this system in terms of non-conservative variables $[\rho, u, p]$, is given by

$$\begin{aligned} \rho(x, 0) &= 1 + 0.2 \sin(2\pi x) , \\ u(x, 0) &= 1, \\ p(x, 0) &= 2, \end{aligned} \tag{97}$$

additionally the system is endowed with Dirichlet boundary conditions in terms of non-conservative variables, given by

$$\mathbf{W}_L(t) = \begin{bmatrix} 1 - 0.2 \sin(2\pi t) \\ 1 \\ 2 \end{bmatrix} , \quad \mathbf{W}_R(t) = \begin{bmatrix} 1 + 0.2 \sin(2\pi(1 - t)) \\ 1 \\ 2 \end{bmatrix} . \tag{98}$$

So, the exact solution is given by

$$\begin{aligned} \rho(x, t) &= 1 + 0.2 \sin(2\pi(x - t)) , \\ u(x, t) &= 1 , \\ p(x, t) &= 2 . \end{aligned} \tag{99}$$

The reverse problems in this case are not straightforward obtained. Thus in order to provide the form of the reverse problems, we define

$$\phi(\mathbf{U}) = \frac{2u_2 + \sqrt{4u_2^2 + 8(\gamma^2 - 1)\left(\frac{u_2^2}{2} - u_1 u_3\right)}}{2(\gamma + 1)} . \tag{100}$$

Therefore, the flux for the reverse problems is given by

$$\mathbf{R}(\mathbf{U}) = \begin{bmatrix} \dfrac{u_1^2}{u_2 - \phi(\mathbf{U})} \\ u_1 \\ \dfrac{u_2}{2} + \dfrac{\phi(\mathbf{U})(3 - \gamma)}{2\gamma - 1)} \end{bmatrix} . \tag{101}$$

Table 5 shows the convergence rate assessment for the Euler equations. Here we have used $t_{out} = 1$ and $C_{CFL} = 0.9$. For simulations, reverse problems have been applied with $N = \bar{M} = 3$

24

Theoretical order : 2

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
32	0.00	$7.35e-03$	0.00	$3.81e-03$	0.00	$4.41e-03$	0.3680
64	2.02	$1.81e-03$	2.16	$8.51e-04$	2.15	$9.93e-04$	0.3880
128	2.08	$4.29e-04$	2.09	$2.00e-04$	2.08	$2.35e-04$	1.0160
256	2.04	$1.04e-04$	2.04	$4.87e-05$	2.04	$5.72e-05$	3.1960
512	2.00	$2.60e-05$	2.02	$1.20e-05$	2.02	$1.41e-05$	11.4280

Theoretical order : 3

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
32	0.00	$1.92e-03$	0.00	$8.99e-04$	0.00	$1.05e-03$	0.1440
64	3.01	$2.38e-04$	2.95	$1.16e-04$	2.97	$1.35e-04$	0.4400
128	3.01	$2.96e-05$	2.98	$1.48e-05$	2.99	$1.69e-05$	1.1320
256	3.00	$3.70e-06$	2.99	$1.86e-06$	3.00	$2.12e-06$	3.8480
512	3.00	$4.62e-07$	3.00	$2.33e-07$	3.00	$2.65e-07$	14.1720

Theoretical order : 4

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
32	0.00	$5.00e-04$	0.00	$1.70e-04$	0.00	$2.14e-04$	5.9920
64	5.79	$9.01e-06$	5.22	$4.56e-06$	5.37	$5.17e-06$	14.5119
128	3.86	$6.21e-07$	3.95	$2.95e-07$	3.93	$3.40e-07$	33.5880
256	3.90	$4.16e-08$	3.91	$1.96e-08$	3.91	$2.27e-08$	89.7599
512	3.95	$2.70e-09$	3.94	$1.27e-09$	3.94	$1.48e-09$	259.4400

Theoretical order : 5

Mesh	L_∞ - err	L_∞- ord	L_1 - err	L_1 - ord	L_2 - err	L_2 - ord	CPU
32	0.00	$3.56e-05$	0.00	$1.92e-05$	0.00	$2.16e-05$	47.8040
64	4.99	$1.12e-06$	4.94	$6.26e-07$	4.94	$7.02e-07$	103.8680
128	4.99	$3.50e-08$	4.95	$2.02e-08$	4.96	$2.26e-08$	223.6119
256	5.00	$1.10e-09$	4.97	$6.43e-010$	4.98	$7.20e-010$	502.5120
512	4.99	$3.45e-011$	4.99	$2.03e-011$	4.99	$2.27e-011$	1154.5119

Table 5: Convergence rates for the Euler equations at output time $t_{out} = 1$. with $C_{cfl} = 0.9$. Reverse boundary conditions are applied with $N = \bar{M} = 3$ and $L = 1.5$.
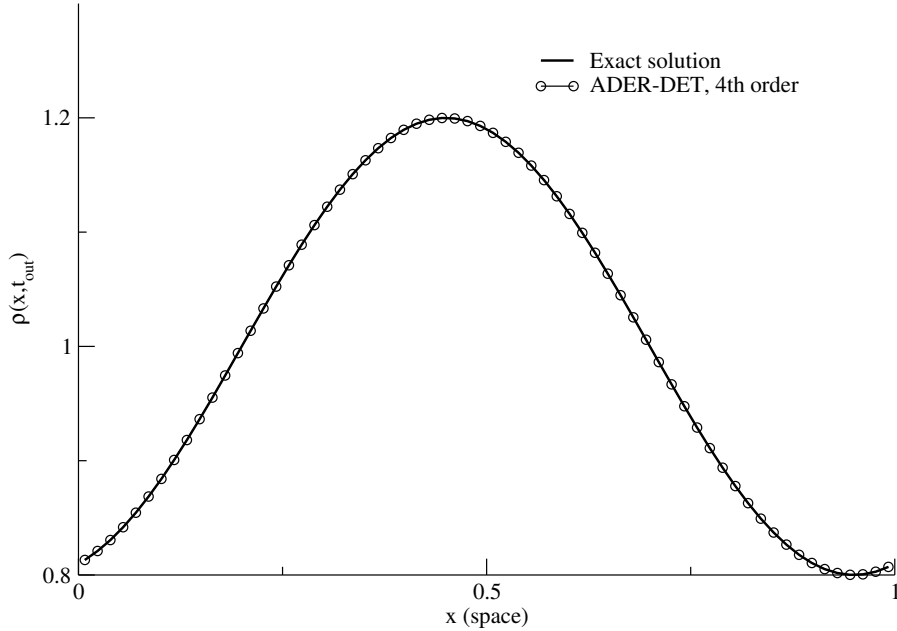
Figure 3: Density for the Euler equation. Comparison between exact and numerical solutions with 64 cells. Reverse boundary conditions have been applied. Parameters are $C_{CFL} = 0.9$ and output time $t_{out} = 0.2$.

and $L = 1.5$ and so $\eta = 0.66$, from proposition 4.1, the scheme should be globally stable. We observe that the expected theoretical orders of accuracy for interior problems are achieved. Figure 3 shows the comparison between the exact and numerical solution of fourth order of accuracy. The numerical solution has been computed with 64 cells. Table 6 shows the CPU-time comparison, between the reverse problems and the inverse Lax-Wendroff procedure. We note that the CPU time has the same order of accuracy for both procedures. The Lax-Wendroff procedure, used in this test is described in Appendix A.

6.5. The Blast wave interaction problem

Here the one-dimensional Euler equations for an ideal gas with $\gamma = 1.4$ in the domain $[0, 1]$ is solved. The initial condition, in terms of non-conservative variables is given by

$$\mathbf{W}(x,0) = \begin{cases} \mathbf{W}_L , & x < 0.1 , \\ \mathbf{W}_C , & 0.1 < x < 0.9 , \\ \mathbf{W}_R , & 0.9 < x , \end{cases} \tag{102}$$

with

$$\mathbf{W}_L = \begin{bmatrix} 1 \\ 0 \\ 1000 \end{bmatrix} , \mathbf{W}_C = \begin{bmatrix} 1 \\ 0 \\ 0.01 \end{bmatrix} , \mathbf{W}_R = \begin{bmatrix} 1 \\ 0 \\ 100 \end{bmatrix} . \tag{103}$$

26

Theoretical order : 2

Mesh	CPU reverse problems	CPU inverse Lax-Wendroff
32	0.3680	0.2720
64	0.3880	0.17199
128	1.0160	0.6599
256	3.1960	2.6239
512	11.4280	9.8200

Theoretical order : 3

Mesh	CPU reverse problems	CPU inverse Lax-Wendroff
32	0.1440	0.2440
64	0.4400	0.3079
128	1.1320	1.0999
256	3.8480	3.9199
512	14.1720	14.5480

Theoretical order : 4

Mesh	CPU reverse problems	CPU inverse Lax-Wendroff
32	5.9920	5.4359
64	14.5119	13.8479
128	33.5880	29.1920
256	89.7599	78.0840
512	259.4400	255.1160

Theoretical order : 5

Mesh	CPU reverse problems	CPU inverse Lax-Wendroff
32	47.8040	48.1220
64	103.8680	97.5440
128	223.6119	215.7440
256	502.5120	476.1519
512	1154.5119	1086.2359

Table 6: CPU for the Euler equations at output time $t_{out} = 1$. with $C_{cfl} = 0.9$. Reverse boundary conditions with $N = \bar{M} = 3$ and $L = 1.5$ (second colums) and inverse Lax-Wendroff procedure for boundaries (third columns).

See [57] for further details. The purpose of this test is the assessment of the present strategy for boundary conditions in the case of solid wall boundary conditions. In this test multiple wave-boundary interaction occurs. As it has become conventional from [57], we solve the problem up to t_{out} = 0.038 with 800 cells and reverse boundary conditions. Here we construct non-conservative prescribed functions through interpolations like (28). So two functions of time are available

$$\begin{aligned}
\tilde{\mathbf{W}}_L(t) &= \mathbf{W}_1^{n-1} + \tfrac{(t-t^n)}{\Delta t}(\mathbf{W}_1^n - \mathbf{W}_1^{n-1}) \,, \\
\tilde{\mathbf{W}}_R(t) &= \mathbf{W}_{N_{int}}^{n-1} + \tfrac{(t-t^n)}{\Delta t}(\mathbf{W}_{N_{int}}^n - \mathbf{W}_{N_{int}}^{n-1}) \,,
\end{aligned} \tag{104}$$

where $\mathbf{W}_1^n$ and $\mathbf{W}_1^{n-1}$ stand by the non-conservative variables associated to the first cell at both time steps t^{n-1} and t^n, respectively. Similarly, $\mathbf{W}_{N_{int}}^{n-1}$ and $\mathbf{W}_{N_{int}}^n$ corresponds to the non-conservative variables associated to the last cell at both times t^{n-1} and t^n. Additionally, we impose the condition of solid wall $(\tilde{\mathbf{W}}_L(t))_2 = -(\tilde{\mathbf{W}}_L(t))_2$ and $(\tilde{\mathbf{W}}_R(t))_2 = -(\tilde{\mathbf{W}}_R(t))_2$. Additionally, we have used the combination $N = \bar{M} = 3$ and $L = 1.5$ then $\eta = 0.66$, so in virtue of proposition 4.1, it ensures the scheme to be globally stable. A reference solution is obtained by using the ADER-HEOC solver of third order of accuracy with 3000 cells, see [11] for further details concerning this solver. Figure 4, shows the result for second and third orders of accuracy, we observe that reverse problems provides very good approximations with respect to the reference solution.

7. Conclusions

In this work a strategy to implement Dirichlet boundary conditions in the context of ADER schemes for hyperbolic conservation laws has been proposed. The strategy requires the acknowledgment of the information at boundary in terms of a state vector, when information at boundary is not prescribed, or partially provided, we proposed a strategy to approximate that information. The use of this information is twofold; first, to build a Riemann problems at extremes of the computational domain and so, as usual, the wave propagation at boundaries is provided through the solution of these problems. Second, to compute ghost cells, they are used to get the stencils for cells near to boundaries which is needed in the reconstruction procedure, a key ingredient to build ADER schemes. Ghost cells are computed from auxiliary problems called reverse problems.

Reverse problems are build from the governing equation of the conservation law, called here interior problem, and from information at boundaries. So in this sense the ghost cells contains a more physical meaning than other approaches like extrapolations. Notice that the present approach can be seen as a numerical version of the inverse Lax-Wendroff procedure, but Taylor expansion and the Chauchy-Kowalewski procedure are not required. In turn, reverse problems are solved numerically by using a conventional second order scheme for hyperbolic balance laws on a suitable mesh, which is built to reproduce the accuracy of the scheme for interior problems.

We have proved analytically that for the scalar case the present strategy is able to reconcile the stability and accuracy of the method for the interior problem. Furthermore, we have obtained a criterion to select the suitable mesh for the reverse problems in order to obtain a numerical schemes which is stable for both the reverse and interior problems. The numerical method to solve the interior problems was the ADER-DET. However, the present boundary treatment has been designed to be implemented in any high order scheme of the family of ADER methods.

We have dealt the issues of partial acknowledgment of information at boundary as in the case of no prescribed outflow boundaries and the case in which interior problems have no an invertible
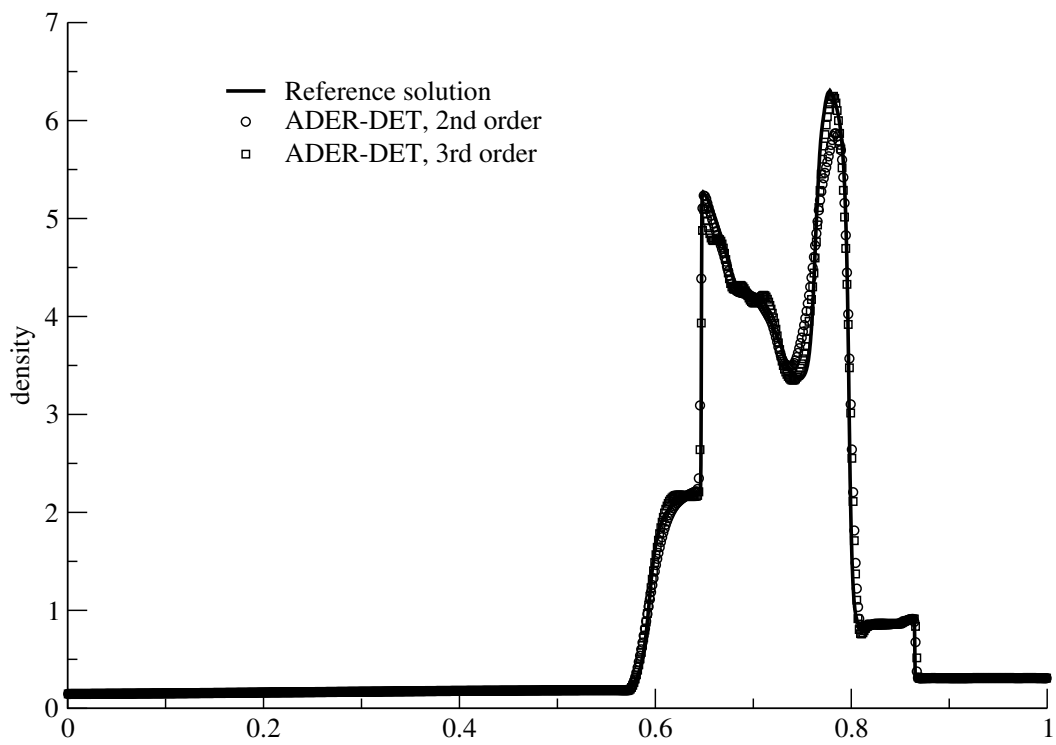
Figure 4: Blast wave interaction. Numerical solution of second and third order of accuracy. Output time $t_{out} = 0.038$ using 800 cells.

flux. We have solved; linear advection equation; linear advection system, a hyperbolic problem with a non-invertible Jacobian matrix and the Euler equations. Empirical convergence rate assessments and CPU time comparison with the inverse Lax-Wendroff approach, were carried out for some of them. We have obtained accuracy in space and time up to fifth oder of accuracy and the scheme has resulted to be comparable with the well known inverse Lax-Wendroff procedure. Thus the present boundary condition treatment in the context of high-order ADER numerical schemes is a simple and feasible strategy in terms of efficiency and accuracy. Extension to high-dimension and for complex geometries will be the issues of a future work.

Acknowledgements

References

References

[1] D. S. Balsara, C. Meyer, M. Dumbser, H. Du, and Z. Xu. Efficient implementation of ADER schemes for Euler and magnetohydrodynamical flows on structured meshes - speed comparisons with Runge-Kutta methods. *Journal of Computational Physics*, 235(0):934 – 969, 2013.

[2] D. S. Balsara, G. I. Montecinos, and E. F. Toro. Exploring various flux vector splittings for the magnetohydrodynamic system. *Journal of Computational Physics*, 311:1 – 21, 2016.

[3] D. S. Balsara, T. Rumpf, M. Dumbser, and C. D. Munz. Efficient, high accuracy ADER-WENO schemes for hydrodynamics and divergence-free magnetohydrodynamics . *Journal of Computational Physics*, 228(7):2480 – 2516, 2009.

[4] M. Ben-Artzi and J. Falcovitz. *Generalized Riemann Problems in Computational Fluid Dynamics*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2003.

[5] M. Berger, C. Helzel, and R. LeVeque. H-box methods for the approximation of hyperbolic conservation laws on irregular grids. *SIAM Journal on Numerical Analysis*, 41(3):893–918, 2003.

[6] M. J. Berger and R. J. LeVeque. Stable boundary conditions for cartesian grid calculations. *Computing Systems in Engineering*, 1(2–4):305 – 311, 1990. Computational Technology for Flight Vehicles.

[7] R. Borsche and J. Kall. ADER schemes and high order coupling on networks of hyperbolic conservation laws. *Journal of Computational Physics*, 273(0):658–670, 2014.

[8] W. Boscheri, D. S. Balsara, and M. Dumbser. Lagrangian ADER-WENO finite volume schemes on unstructured triangular meshes based on genuinely multidimensional HLL Riemann solvers. *Journal of Computational Physics*, 267:112–138, 2014.

[9] W. Boscheri, M. Dumbser, and D. S. Balsara. High-order ADER-WENO ALE schemes on unstructured triangular meshes-application of several node solvers to hydrodynamics and magnetohydrodynamics. *International Journal for Numerical Methods in Fluids*, 2014.

[10] C. E. Castro. *High–order ADER FV/DG numerical methods for hyperbolic equations*. PhD thesis, Department of Civil and Environmental Engineering, University of Trento, Italy, 2007.

[11] C. E. Castro and E. F. Toro. Solvers for the high–order Riemann problem for hyperbolic balance laws. *Journal of Computational Physics*, 227:2481–2513, 2008.

[12] C. S. Chou and C. W. Shu. High order residual distribution conservative finite difference WENO schemes for steady state problems on non-smooth meshes. *Journal of Computational Physics*, 214:698–724, 2006.

[13] M. Dumbser. *Arbitrary high order schemes for the solution of hyperbolic conservation laws in complex domains*. PhD thesis, Institut für Aero- un Gasdynamik, Universität Stuttgart, Germany, 2005.

[14] M. Dumbser. Arbitrary high order PNPM schemes on unstructured meshes for the compressible Navier-Stokes equations. *Computers & Fluids*, 39:60–76, 2010.

[15] M. Dumbser, C. Enaux, and E. F. Toro. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *Journal of Computational Physics*, 227(8):3971–4001, 2008.

[16] M. Dumbser, A. Hidalgo, and O. Zanotti. High order space-time adaptive ADER-WENO finite volume schemes for non-conservative hyperbolic systems. *Computer Methods in Applied Mechanics and Engineering*, 268(0):359 – 387, 2014.

[17] M. Dumbser and M. Käser. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. *Journal of Computational Physics*, 221(2):693–723, 2007.

[18] M. Dumbser, M. Käser, and E. F. Toro. An arbitrary high order discontinuous Galerkin method for elastic waves on unstructured meshes V: Local time stepping and *p*–adaptivity. *Geophysical Journal International*, 171:695–717, 2007.

[19] M. Dumbser and C. D. Munz. ADER Discontinuous Galerkin Schemes for Aeroacoustics. *Comptes Rendus Mécanique*, 333:683–687, 2005.

[20] M. Dumbser and C. D. Munz. Building blocks for arbitrary high order discontinuous Galerkin schemes. *Journal of Scientific Computing*, 27:215–230, 2006.

[21] M. Dumbser, O. Zanotti, A. Hidalgo, and D. S. Balsara. ADER-WENO finite volume schemes with space-time adaptive mesh refinement. *Communications in Computational Physics*, 248:257–286, 2013.

[22] M. Dumbser, O. Zanotti, R. Loubere, and S. Diot. A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws. *Journal of Computational Physics*, 278:47–75, 2014.

[23] F. Filbet and C. Yang. An inverse LaxWendroff method for boundary conditions applied to Boltzmann type models. *Journal of Computational Physics*, 245(0):43 – 61, 2013.

[24] S. K. Godunov. A Finite Difference Method for the Computation of Discontinuous Solutions of the Equations of Fluid Dynamics. *Matematicheskii Sbornik*, 47:357–393, 1959.

[25] A. Harten and S. Osher. Uniformly High-Order Accurate Nonoscillatory Schemes. I. *SIAM Journal on Numerical Analysis*, 24(2):279–309, 1987.

[26] A. Hidalgo and M. Dumbser. ADER schemes for nonlinear systems of stiff advection-diffusion-reaction equations. *Journal of Scientific Computing*, 48:173–189, 2011.

[27] A. Hidalgo, L. Tello, and E. F. Toro. Numerical and analytical study of an atherosclerosis inflammatory disease model. *Journal of Mathematical Biology*, pages 1–30, 2013.

[28] L. Huang, C.-W. Shu, and M. Zhang. Numerical boundary conditions for the fast sweeping high order WENO methods for solving the Eikonal equation. *Journal of Computational Mathematics*, 26(3):336–346, 2008.

[29] M. Käser. *Adaptive methods for the numerical simulation of transport processes*. PhD thesis, Institute of Numerical Mathematics and Scientific Computing, University of Munich, Germany, 2003.

[30] M. Käser. ADER schemes for the solution of conservation laws on adaptive triangulations. Mathematical Methods and Modelling in Hydrocarbon Exploration and Production. Springer–Verlag, Vol. 7, 2004.

[31] M. Käser and A. Iske. Adaptive ADER schemes for the solution of scalar non–linear hyperbolic problems. *Journal of Computational Physics*, 205:489–508, 2005.

[32] S.G. Krantz and H.R. Parks. *The Implicit Function Theorem: History, Theory, and Applications*. The Implicit Function Theorem: History, Theory, and Applications. Birkhäuser, 2002.

[33] R. Loubere, M. Dumbser, and S. Diot. A new family of high order unstructured MOOD and ADER finite volume schemes for multidimensional systems of hyperbolic conservation laws. *Communications in Computational Physics*, 16(3):718763, 2014.

[34] G. Montecinos, C. E. Castro, M. Dumbser, and E. F. Toro. Comparison of solvers for the generalized Riemann problem for hyperbolic systems with source terms. *Journal of Computational Physics*, 231:6472–6494, 2012.

[35] G. I. Montecinos, J. C. López-Rios, R. Lecaros, J. H. Ortega, and E. F. Toro. An ader-type scheme for a class of equations arising from the water-wave theory. *Computers & Fluids*, 132:76 – 93, 2016.

[36] G. I. Montecinos, L. O. Müller, and E. F. Toro. Hyperbolic reformulation of a 1D viscoelastic blood flow model and ADER finite volume schemes. *Preprint NI14014-NPA, Isaac Newton Institute for Mathematical Sciences*, 2014.

[37] G. I. Montecinos, L. O. Müller, and E. F. Toro. Hyperbolic reformulation of a 1D viscoelastic blood flow model and ADER finite volume schemes. *Journal of Computational Physics*, 266:101–123, 2014.

[38] G. I. Montecinos and E. F. Toro. Reformulations for general advection - diffusion - reaction equations and locally implicit ADER schemes. *Journal of Computational Physics*, 275:415–442, 2014.

[39] L. O. Müller, C. Parés, and E. F. Toro. Well-balanced high-order numerical schemes for one-dimensional blood flow in vessels with varying mechanical properties. *Journal of Computational Physics*, 242:53–85, 2013.

[40] L. O. Müller and E. F. Toro. A global multiscale mathematical model for the human circulation with emphasis on the venous system. *International Journal for Numerical Methods in Biomedical Engineering*, 30(7):681–725, 2014.

[41] V. V. Rusanov. Calculation of Interaction of NonSteady Shock Waves with Obstacles. *Union of Soviet Socialist Republics computational mathematics and mathematical physics*, 1:267–279, 1961.

[42] T. Schwartzkopff, C. D. Munz, and E. F. Toro. ADER: High–order approach for linear hyperbolic systems in 2D. *Journal of Scientific Computing*, 17:231–240, 2002.

[43] Y. Takakura and E. F. Toro. Arbitrarily Accurate Non–Oscillatory Schemes for a Non–Linear Conservation Law. *Computational Fluid Dynamics*, 11:7–18, 2002.

[44] S. Tan and C.-W. Shu. Inverse Lax-Wendroff procedure for numerical boundary conditions of conservation laws. *Journal of Computational Physics*, 229(21):8144 – 8166, 2010.

[45] S. Tan and C.-W. Shu. A high order moving boundary treatment for compressible inviscid flows. *Journal of Computational Physics*, 230(15):6023 – 6036, 2011.

[46] S. Tan, C. Wang, C.-W. Shu, and J. Ning. Efficient implementation of high order inverse Lax-Wendroff boundary treatment for conservation laws. *Journal of Computational Physics*, 231(6):2510 – 2527, 2012.

[47] V. A. Titarev and E. F. Toro. ADER: Arbitrary high order Godunov approach. *Journal of Scientific Computing*, 17:609–618, 2002.

[48] V. A. Titarev and E. F. Toro. High–order ADER schemes for scalar advection–reaction–diffusion equations. *CFD Journal*, 12(1):1–6, 2003.

[49] V. A. Titarev and E. F. Toro. ADER schemes for three–dimensional hyperbolic systems. *Journal of Computational Physics*, 204:715–736, 2005.

[50] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. Springer-Verlag, third edition, 2009. ISBN 978-3-540-25202-3.

[51] E. F. Toro and A. Hidalgo. ADER finite volume schemes for diffusion–reaction equations. *Applied Numerical Mathematics*, 59:73–100, 2009.

[52] E. F. Toro, R. C. Millington, and L. A. M. Nejad. Towards very high–order Godunov schemes. In *Godunov Methods: Theory and Applications. Edited Review, E. F. Toro (Editor)*, pages 905–937. Kluwer Academic/Plenum Publishers, 2001.

[53] E. F. Toro and G. I. Montecinos. Advection-diffusion-reaction equations: Hyperbolisation and high-order ADER discretizations. *SIAM Journal on Scientific Computing*, 36(5):A2423–A2457, 2014.

[54] E. F. Toro and V. A. Titarev. Solution of the generalised Riemann problem for advection–reaction equations. *Proceedings of the Royal Society of London A*, 458:271–281, 2002.

[55] E. F. Toro and V. A. Titarev. ADER schemes for scalar non-linear hyperbolic conservation laws with source terms in three-space dimensions. *Journal of Computational Physics*, 202(1):196–215, 2005.

[56] F. Vilar and C.-W. Shu. Development and stability analysis of the inverse Lax-Wendroff boundary treatment for central compact schemes. *ESAIM: Mathematical Modelling and Numerical Analysis*, page Published online, June 2014.

[57] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54(1):115 – 173, 1984.

[58] Tao Xiong, Mengping Zhang, Yong-Tao Zhang, and Chi-Wang Shu. Fast sweeping fifth order weno scheme for static hamilton-jacobi equations with accurate boundary treatment. *Journal of Scientific Computing*, 45(1-3):514–536, 2010.

[59] C. E. Zambra, M. Dumbser, E. F. Toro, and N. O. Moraga. A novel numerical method of high-order accuracy for flow in unsaturated porous media. *International Journal for Numerical Methods in Engineering*, 89(2):227–240, 2012.

[60] P. Zhang and R. X. Liu. Hyperbolic conservation laws with space-dependent flux: I. Characteristics theory and Riemann problem. *Journal of Computational and Applied Mathematics*, 156(1):1 – 21, 2003.

[61] P. Zhang and R. X. Liu. Hyperbolic conservation laws with space-dependent fluxes: II. General study of numerical fluxes. *Journal of Computational and Applied Mathematics*, 176(1):105 – 129, 2005.

[62] Y.-T. Zhang, H.-K. Zhao, and J. Qian. High order fast sweeping methods for static hamiltonjacobi equations. *Journal of Scientific Computing*, 29(1):25–56, 2006.

Appendix A. A simple strategy to carry out the inverse Lax-Wendroff procedure

Conventional Lax-Wendroff or Cauchy-Kowalewky procedure consists of providing time-derivatives in terms of spatial-derivatives. This is carried out by a repetitively differentiation of of the governing equations. In opposite, inverse Lax-Wendroff procedure provides space-derivatives in terms of time-derivatives. In general the Cauchy-Kowalewsky procedure can be cumbersome for complex conservations laws and symbolic software manipulators may be needed. Due to the fact that Cauchy-Kowalewsky procedure is commonly available we use it to construct a strategy for the inverse Lax-Wendroff procedure.

To set the problem in the context of applications to boundary conditions and without loss of generality, let us consider an inlet boundary condition. Then ghost cells can be approximated by

using a Taylor series expansion in space as follows

$$\mathbf{Q}(x,t) = \mathbf{Q}(x_L,t) + \sum_{k=1}^{M} \frac{(x-x_L)^k}{k!} \partial_x^{(k)} \mathbf{Q}(x_L,t) . \tag{A.1}$$

Let us assume, as usual, that time derivatives are obtained in terms of spatial derivatives through the Cauchy-Kowalewsky functionals

$$\partial_t^{(k)} \mathbf{Q}(x,t) = \mathbf{G}^k(\mathbf{Q}(x,t), ..., \partial_x^{(k)} \mathbf{Q}(x,t)) . \tag{A.2}$$

Then, given $\partial_t^{(l)} \mathbf{Q}(x_L,t) = \mathbf{G}_L^{(l)}(t)$, $l = 1, ..., M$, we look for values $\partial_x^{(j)} \mathbf{Q}(x_L,t)$ such that

$$\mathbf{G}_L^{(l)}(t) - \mathbf{G}^k(\mathbf{Q}(x_L,t), ..., \partial_x^{(k)} \mathbf{Q}(x_L,t)) = \mathbf{0} , \tag{A.3}$$

for all $k = 1, ..., M$. Thus an algebraic equation is then constructed.