Archives of Computational Methods in Engineering manuscript No.
(will be inserted by the editor)

A sequel of Inverse Lax–Wendroff high order wall boundary treatment for conservation laws

Rafael B. de Rezende Borges · Nicholas Dicati P. da Silva · Francisco A. A. Gomes · Chi-Wang Shu · Sirui Tan

Received: date / Accepted: date

Conflict of interest

The authors declare that they have no conflict of interest.

References

Rafael B. de Rezende Borges
Rio de Janeiro State University, Mathematics and Statistics Institute, Rio de Janeiro, Brazil
https://orcid.org/0000-0002-2576-8118
E-mail: rafael.borges@ime.uerj.br

Nicholas Dicati P. da Silva
Federal University of Paraná, Laboratory of Numerical Experimentation, Department of Mechanical Engineering, Curitiba, Brazil
https://orcid.org/0000-0002-5808-3932
E-mail: ndicati@gmail.com

Francisco A. A. Gomes
Federal University of Technology - Paraná, Department of Mechanical Engineering, Pato Branco, Brazil
E-mail: franciscogomes@utfpr.edu.br

Chi-Wang Shu
Brown University, Division of Applied Mathematics, Providence, United States of America
E-mail: shu@dam.brown.edu

Sirui Tan
Brown University, Formerly Division of Applied Mathematics, Providence, United States of America
https://orcid.org/0000-0002-8150-3261
E-mail: sirui_tan@alumni.brown.edu

Archives of Computational Methods in Engineering manuscript No.
(will be inserted by the editor)

A sequel of the inverse Lax–Wendroff high order wall boundary treatment for conservation laws

Authors

Received: date / Accepted: date

Abstract When solving CFD problems, the solver, or the numerical code, plays an important role. Depending on the phenomena and problem domain, designing such numerical codes can be hard work. One strategy is to start with simple problems and construct the code as building blocks. The purpose of this work is to provide a detailed review of the theory to compute analytical and exact solutions, and recent numerical methods to construct a code to solve compressible and inviscid fluid flows with high-resolution, arbitrary domains, non-linear phenomena, and on rectangular meshes. We also propose a modification to the inverse Lax-Wendroff procedure solid wall treatment and two-dimensional WENO-type extrapolation stencil selection and weights to handle more generic situations. To test our modifications, we use the finite difference method, Lax–Friedrichs splitting, WENO-Z+ scheme, and third-order strong stability preserving Runge-Kutta time discretization. Our first problem is a simple one-dimensional transient problem with periodic boundary conditions, which is useful for constructing the core solver. Then, we move to the one-dimensional Rayleigh flow, which can handle flows with heat exchange and requires more detailed boundary treatment. The next problem is the quasi-one-dimensional nozzle flow with and without shock, where the boundary treatment needs a few adjustments. The first two-dimensional problem is the Ringleb flow, and despite being smooth, it has a curved wall as the left boundary. Finally, the last problem is a two-dimensional conical flow, which presents an oblique shock and an inclined straight line wall being the cone surface. We show that the designed accuracy is being reached for smooth problems, that high-resolution is being attained for non-smooth problems, and that our modifications produce similar results while providing a more generic way to tread solid walls.

Keywords High-resolution · High-order · Rectangular meshes · Compressible · Inviscid · Boundary treatment

1 Introduction

High-resolution and high-order numerical schemes are the most recent and popular methods for solving flows containing discontinuities and other non-linear phenomena [1–7]. However, detailed review for particular cases is not easily found. Those methods were initially developed for hyperbolic conservation laws, in which the Euler equations are an example and can model flows through nozzles and around bodies [8,9,4].

The weighted essentially non-oscillatory (WENO) scheme is an example of high-order and high-resolution schemes used to approximate spatial derivatives based on smoothness indicators and substencils. When the stencil, or substencil, contains a discontinuity, oscillations arise in the approximation. The WENO idea is to avoid substencils containing discontinuities through previously established weights [10,2]. Since its first version, the WENO resolution and order were improved in several works [10–12,6]. Still, as other higher-order methods, the WENO suffers from accuracy degeneration near discontinuities [13,14]. The WENO scheme accuracy is also subject to proper boundary condition treatment, usually achieved with ghost points.

Boundary conditions can be applied in different ways depending on the problem and its domain. For example, problems with periodic solutions, such as sine function, can be computed with periodic boundary conditions.

Authors
Affiliations
e-mails

Another example is related to heat transfer problems, where Dirichlet, Neumann, and Robin boundary conditions can be used [15,16]. Unstructured meshes add more flexibility to treat arbitrary domains. However, the finite difference method demands simpler meshes [2], which can cause difficulties when solving problems with arbitrary domains and specific boundary condition treatments. Different analysis and strategies were developed regarding arbitrary domains and finite difference, such as grid overlapping and embedded methods [17–20]. In particular, the inverse Lax-Wendroff (ILW) procedure uses the given boundary condition and the partial differential equations to apply the boundary conditions in rectangular grids without requiring the physical boundary to be aligned with the meshes [5,21].

The original ILW procedure for solid walls requires a 2D WENO-type extrapolation and a Taylor expansion in the normal direction, from the boundary point, a_1, a_2, or a_3, to each ghost point, g_1, g_2, or g_3, see Fig. 1. Depending on the contour, the boundary point can lay outside the domain. This situation can happen when the ghost point is close to the boundary ends, as shown in Fig. 1 for a_2 and a_3, where the solid curve is the boundary and x_l and x_r are the boundary ends. This can be avoided by choosing another point, b, and expanding in the y-direction, for instance. Furthermore, this requires only two 2D WENO-type extrapolation and one Taylor expansion for the three ghost points g_1, g_2, and g_3.

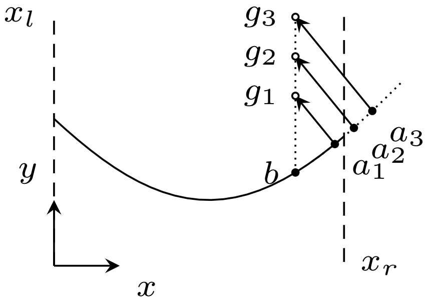

Fig. 1: Example of boundary points out of the domain

Usually, discontinuities, non-linear phenomena, and arbitrary domains are encountered in compressible fluid flows [22–24]. For simpler one-dimensional (1D) problems, such as the flow with heat addition (Rayleigh flow), the domain is straightforward and shocks can be avoided. However, for quasi-one-dimensional (Q1D) and two-dimensional (2D) flows around bodies, the domains are arbitrary and shocks are more frequent. That is, the numerical method needs to be robust and accounts for geometry variations.

We aim to present a detailed review of the theory to compute analytical and exact solutions, and to obtain a numerical code to solve compressible and inviscid fluid flows with high-resolution, arbitrary domains, and shocks while proposing modifications to the numerical methods. Starting with the numerical code to build the core solver, the review will be presented as we move from one problem to another. The first problem is a simple periodic problem, followed by 1D Rayleigh flow, Q1D nozzle flows with and without shock, 2D Ringleb flow, and 2D conical flow. To do that, we use the finite difference method with Lax–Friedrichs splitting [2], WENO-Z+ scheme [6], third-order strong stability preserving (SSP) Runge-Kutta [2], and ILW [21].

2 Mathematical models and numerical methods

In this work, the problems will be modeled by the Euler equations [25]

$$\boldsymbol{U}_t + \boldsymbol{F}(\boldsymbol{U})_\xi + \boldsymbol{G}(\boldsymbol{U})_\eta = \boldsymbol{S}(\boldsymbol{U}), \tag{1}$$

where $\boldsymbol{U}$ is the conservative variables vector, $\boldsymbol{F}$ and $\boldsymbol{G}$ are the physical flux vectors in ξ and η direction, and $\boldsymbol{S}$ is the source term vector, which depends on the problem. Depending on the coordinate system, ξ can be the x- or z-direction, and η can be the y- or r-direction. For 1D and Q1D problems, $\boldsymbol{G}(\boldsymbol{U})_\eta = 0$.

For the finite difference method, (1) is approximated as [2]

$$\begin{aligned}\frac{d}{dt}\boldsymbol{U}_{i,j} = &-\frac{1}{\Delta\xi}(\hat{\boldsymbol{F}}_{i+1/2,j} - \hat{\boldsymbol{F}}_{i-1/2,j}) \\ &-\frac{1}{\Delta\eta}(\hat{\boldsymbol{G}}_{i,j+1/2} - \hat{\boldsymbol{G}}_{i,j-1/2}) + \boldsymbol{S}(\boldsymbol{U}_{i,j}),\end{aligned} \tag{2}$$

where $\hat{\boldsymbol{F}}$ and $\hat{\boldsymbol{G}}$ are the numerical fluxes, and $\Delta\xi$ and $\Delta\eta$ the mesh sizes in each direction.

To approximate $\hat{\boldsymbol{F}}_{i+1/2,j}$, the first step is the change from the conservative to the local characteristic variables [2]

$$\boldsymbol{V} = \boldsymbol{L}(\boldsymbol{U}_{i+1/2,j})\boldsymbol{U}, \tag{3}$$

$$\boldsymbol{H} = \boldsymbol{L}(\boldsymbol{U}_{i+1/2,j})\boldsymbol{F}, \tag{4}$$

where $\boldsymbol{V}$ and $\boldsymbol{H}$ are the characteristic variable vectors and their relative flux vectors, $\boldsymbol{L}$ is the left eigenvector and $\boldsymbol{U}_{i+1/2,j} \approx (\boldsymbol{U}_{i,j} + \boldsymbol{U}_{i+1,j})/2$ [2].

The next step is the flux splitting, one of the most simple but most dissipative is the Lax–Friedrichs splitting. Still, higher-order approximations such as WENO compensate this dissipation [2]. Applied to the characteristic variables and their fluxes, the splitting yields

$$\boldsymbol{H}^{\pm}(\boldsymbol{V}) = \frac{1}{2}(\boldsymbol{H}(\boldsymbol{V}) \pm \alpha\boldsymbol{V}), \tag{5}$$

where $\alpha = \max_{\boldsymbol{U}} \max_{1\leq j\leq m} |\lambda_j(\boldsymbol{U})|$ is computed over the relevant range of $\boldsymbol{U}$ [2], which in this work is the whole domain. The WENO-Z+ is used to approximate $\hat{\boldsymbol{H}}^{\pm}_{i+1/2,j}$ through [6]

$$(\hat{H}_m)^+_{i+1/2,j} = \omega_0 f_0 + \omega_1 f_1 + \omega_2 f_2, \tag{6}$$

where m is the vector component quantity which depends on the problem, ω are the non-linear weights and f are polynomial approximations to $\hat{\boldsymbol{H}}^{+}_{i+1/2,j}$. The non-linear weights can be computed as [10,6]

$$\omega_k = \frac{\alpha_k}{\sum_{j=0}^{2}\alpha_j}, \quad k = 0, \ldots, 2, \tag{7}$$

and α_k are the WENO-Z+ weights [6],

$$\alpha_k = d_k\left[1 + \left(\frac{\tau+\epsilon}{\beta_k+\epsilon}\right)^p + \lambda\left(\frac{\beta_k+\epsilon}{\tau+\epsilon}\right)\right], \; k = 0, \ldots, 2, \tag{8}$$

where d_k are the ideal weights, τ is the global smoothness indicator, β_k are the local smoothness indicators, ϵ is the sensitivity parameter, p is the power parameter and λ is a parameter for fine-tuning the size of the increment of the weight of less smooth stencils [6]. In this work, $p = 2$, $\epsilon = 1E - 40$, and $\lambda = \Delta_\xi^{2/3}$. d_k, β_k, and f_k can be computed as follows [6]:

$$d_0 = \frac{1}{10}, \qquad d_1 = \frac{6}{10}, \qquad d_2 = \frac{3}{10}, \tag{9}$$

$$\begin{aligned}
\beta_0 =& \frac{1}{4}\left[(H_m)^+_{i-2,j} - 4(H_m)^+_{i-1,j} + 3(H_m)^+_{i,j}\right]^2 \\
&+ \frac{13}{12}\left[(H_m)^+_{i-2,j} - 2(H_m)^+_{i-1,j} + (H_m)^+_{i,j}\right]^2, \\
\beta_1 =& \frac{1}{4}\left[-(H_m)^+_{i-1,j} + (H_m)^+_{i+1,j}\right]^2 \\
&+ \frac{13}{12}\left[(H_m)^+_{i-1,j} - 2(H_m)^+_{i,j} + (H_m)^+_{i+1,j}\right]^2, \\
\beta_2 =& \frac{1}{4}\left[-3(H_m)^+_{i,j} + 4(H_m)^+_{i+1,j} - (H_m)^+_{i+2,j}\right]^2 \\
&+ \frac{13}{12}\left[(H_m)^+_{i,j} - 2(H_m)^+_{i+1,j} + (H_m)^+_{i+2,j}\right]^2,
\end{aligned} \tag{10}$$

$$\begin{aligned}
f_0 =& \frac{1}{6}\left[2(H_m)^+_{i-2,j} - 7(H_m)^+_{i-1,j} + 11(H_m)^+_{i,j}\right], \\
f_1 =& \frac{1}{6}\left[-(H_m)^+_{i-1,j} + 5(H_m)^+_{i,j} + 2(H_m)^+_{i+1,j}\right], \\
f_2 =& \frac{1}{6}\left[2(H_m)^+_{i,j} + 5(H_m)^+_{i+1,j} - (H_m)^+_{i+2,j}\right].
\end{aligned} \tag{11}$$

Notice that $(\hat{H}_m)^+_{i+1/2,j}$ is approximated by the stencil [12,6]

$$\begin{aligned}
S = \{H_m)^+_{i-2,j}, &(H_m)^+_{i-1,j}, (H_m)^+_{i,j}, \\
&(H_m)^+_{i+1,j}, (H_m)^+_{i+2,j}\}.
\end{aligned} \tag{12}$$

Similarly, $(\hat{H}_m)^-_{i+1/2,j}$ can be approximated with [12,6]

$$\begin{aligned}
S = \{(H_m)^-_{i+3,j}, (H_m)^-_{i+2,j}, &(H_m)^-_{i+1,j}, \\
&(H_m)^-_{i,j}, (H_m)^-_{i-1,j}\}.
\end{aligned} \tag{13}$$

Then, [2]

$$(\hat{H}_m)_{i+1/2,j} = (\hat{H}_m)^+_{i+1/2,j} + (\hat{H}_m)^-_{i+1/2,j}, \tag{14}$$

and the conservative variables numerical flux is [2]

$$\hat{\boldsymbol{F}}_{i+1/2,j} = \hat{\boldsymbol{H}}_{i+1/2,j}\boldsymbol{R}(\boldsymbol{U}_{i+1/2,}), \tag{15}$$

where $\boldsymbol{R}$ is the right eigenvector.

If necessary, the same procedure can be used for the η direction. The time integration can be done with the third-order SSP Runge-Kutta [2]

$$\begin{aligned}
{U_m}^{(1)} &= {U_m}^n + \Delta t L({U_m}^n), \\
{U_m}^{(2)} &= \frac{3}{4}{U_m}^n + \frac{1}{4}{U_m}^{(1)} + \frac{1}{4}\Delta t L({U_m}^{(1)}), \\
{U_m}^{n+1} &= \frac{1}{3}{U_m}^n + \frac{2}{3}{U_m}^{(2)} + \frac{2}{3}\Delta t L({U_m}^{(2)}),
\end{aligned} \tag{16}$$

where Δt is the time step, n and $n+1$ represent the actual and the next time step, and $L(\cdot)$ is the spatial approximation. Δt can be computed as

$$\Delta t = \min\left[\min(\Delta\xi, \Delta\eta)\mathrm{CFL}/\alpha, \min(\Delta\xi, \Delta\eta)^{5/3}\right], \tag{17}$$

where CFL $= 0.5$ and α is computed in the whole $\boldsymbol{U}$ range.

Although most of the problems in this work are steady, the time integration is still needed to the time marching technique. To compute (1), we still need to compute the boundary conditions, which will be presented in the next sections for each problem.

2.1 Simple periodic problem

For the simple periodic problem $\xi = x$, $\boldsymbol{G}(\boldsymbol{U})_\eta = \boldsymbol{0}$, $\boldsymbol{S}(\boldsymbol{U}) = \boldsymbol{0}$, [25]

$$\boldsymbol{U} = \begin{bmatrix}\rho \\ \rho u \\ E\end{bmatrix}, \quad \boldsymbol{F}(\boldsymbol{U})_x = \begin{bmatrix}\rho u \\ \rho u^2 + p \\ u(E+p)\end{bmatrix}, \tag{18}$$

where ρ, u, E, and p are the density, velocity in x-direction, total energy, and pressure. Those properties are related through the equation of state, which can be used to write [25]

$$E = \frac{p}{\gamma - 1} + \frac{\rho u^2}{2}, \tag{19}$$

where $\gamma = 1.4$ is the specific heat ratio for the air as an ideal gas.

The domain is discretized as follows

$$x_i = x_l + \mathit{\Delta} x(i-1/2), \quad 1 \le i \le N_x, \\ \mathit{\Delta} x = (x_r - x_l)/N_x, \tag{20}$$

where $x_l = 0\ m$ and $x_r = 1\ m$ are the left and right boundary positions, and N_x is the point quantity in the x-direction.

The initial conditions for this problem are

$$\rho(x,0) = 1 + \sin(2\pi x)/5, \quad u(x,0) = 1, \\ p(x,0) = 1. \tag{21}$$

It can be easily shown that (1), for this initial condition, is similar to

$$\rho_t + \rho_x = 0, \tag{22}$$

and that the analytical solution is [3]

$$\tilde{\rho}(x,t) = 1 + \sin[(x-t)2\pi]/5, \tag{23}$$

with $t = 1$ as the final time.

Since the solution is periodic, the periodic boundary conditions can be applied together with ghost points

$$\boldsymbol{U}_{-2} = \boldsymbol{U}_{N_x-2}, \quad \boldsymbol{U}_{-1} = \boldsymbol{U}_{N_x-1}, \quad \boldsymbol{U}_0 = \boldsymbol{U}_{N_x}, \\ \boldsymbol{U}_{N_x+1} = \boldsymbol{U}_1, \quad \boldsymbol{U}_{N_x+2} = \boldsymbol{U}_2, \quad \boldsymbol{U}_{N_x+3} = \boldsymbol{U}_3. \tag{24}$$

Although simple, this problem is a good starting point to build compressible and inviscid fluid flow codes, since all numerical tools presented before are used. Once the code and solution were verified, one can move to the next problem.

2.2 One-dimensional Rayleigh flow

For the 1D Rayleigh flow, $\xi = x$, $\boldsymbol{U}$, $\boldsymbol{F}(\boldsymbol{U})_x$, and $\boldsymbol{G}(\boldsymbol{U})_\eta$ are the same as for the simple periodic problem, and [8]

$$\boldsymbol{S}(\boldsymbol{U}) = \begin{bmatrix} 0 \\ 0 \\ \dfrac{\mathit{\Delta} q \rho u}{\mathit{\Delta} x} \end{bmatrix}, \tag{25}$$

where $\mathit{\Delta} q \equiv q/N_x$ is the heat added in each point and q is the total heat per unit mass added to the flow.

The domain is discretized with (20) and the flow parameters are presented in Tab. 1, where M is the Mach number and T is the temperature.

Table 1: 1D Rayleigh flow parameters

$x_l\ (m)$	0	$T_l\ (K)$	400
$x_r\ (m)$	0.2	$p_l\ (Pa)$	$2E+5$
M_l	1.3	$q\ (J/kg)$	$-5E+4$

The Mach number can be computed as [8,9]

$$M = \frac{|\boldsymbol{V}|}{a} = \frac{\sqrt{u^2+v^2}}{a}, \tag{26}$$

where $\boldsymbol{V}$ is the velocity vector, v is the η-direction velocity component ($v = 0$ for 1D problems), and $a = \sqrt{\gamma p/\rho}$ is the speed of sound.

Pressure, density and temperature are related through the following equivalent form of the equation of state [8]

$$p = \rho R T, \tag{27}$$

where $R = 286.9\ J/(kgK)$ is the gas constant for the air as an ideal gas.

Since this is a steady problem, the initial condition is an initial estimate. We use a more general initial estimate in the first few meshes to check the convergence, then we switch to the analytical or exact solution. In this problem, the initial estimate is

$$\rho_i = \rho_l, \quad u_i = u_l, \quad p_i = p_l, \quad 1 \le i \le N_x. \tag{28}$$

For the 1D Rayleigh flow, the heat added will change the total temperature of the flow [8]

$$q = c_p(T_{02} - T_{01}), \tag{29}$$

where the subscript 0 means total (or stagnation) property, subscripts 2 and 1 two arbitrary points in the flow, and $c_p = \gamma R/(\gamma - 1)$ is the specific heat at constant pressure. The total temperature can be obtained with [8]

$$\frac{T_0}{T} = 1 + \frac{\gamma - 1}{2} M^2. \tag{30}$$

The Mach number at any point can be computed through [8]

$$\frac{T_{02}}{T_{01}} = \left(\frac{1+\gamma M_1^2}{1+\gamma M_2^2}\right)^2 \left(\frac{M_2}{M_1}\right)^2 \left(\frac{1+\frac{\gamma-1}{2}M_2^2}{1+\frac{\gamma-1}{2}M_1^2}\right), \tag{31}$$

which can be solved by an algebraic software, e.g., Maxima, or by a method for finding roots.

Now, the velocity can be computed with (26), and density and pressure with [8]

$$\frac{\rho_2}{\rho_1} = \left(\frac{1+\gamma M_2^2}{1+\gamma M_1^2}\right)\left(\frac{M_1}{M_2}\right)^2, \tag{32}$$

$$\frac{p_2}{p_1} = \frac{1+\gamma M_1^2}{1+\gamma M_2^2}. \tag{33}$$

For the boundary conditions, the eigenvalues should be checked first [25]

$$\lambda_1 = u - a, \quad \lambda_2 = u, \quad \lambda_3 = u + a. \tag{34}$$

Since the flow is supersonic everywhere, at the inflow all boundary conditions shall be imposed and none at the outflow. The ILW procedure uses a Taylor expansion around the boundary to compute the properties at the ghost points [21]

$$(U_m)_i = \sum_{k=0}^{4} \frac{(x_i - x_b)^k}{k!} U_m^{*(k)}, \tag{35}$$

where x_b is the boundary position and $U_m^{*(k)}$ is a $(5-k)$th order approximation of the spatial derivatives of U_m at the boundary.

For the left boundary, $U_m^{*(0)}$ is computed with [21]

$$\boldsymbol{U}^{*(0)} = \boldsymbol{U}_l. \tag{36}$$

For $U_m^{*(1)}$, we use the Euler equations for steady flow, [21]

$$\begin{aligned} &(U_2)_x = S_1(\boldsymbol{U}^{*(0)}), \\ &\left[\frac{U_2^2}{U_1} + (\gamma - 1)\left(U_3 - \frac{U_2^2}{2U_1}\right)\right]_x = S_2(\boldsymbol{U}^{*(0)}), \\ &\left\{\frac{U_2}{U_1}\left[U_3 + (\gamma - 1)\left(U_3 - \frac{U_2^2}{2U_1}\right)\right]\right\}_x = S_3(\boldsymbol{U}^{*(0)}). \end{aligned} \tag{37}$$

Notice that the derivatives of (37) need to be expanded to form a system with three equations and three unknowns, $\boldsymbol{U}^{*(1)}$.

As suggested in [21], the other $U_m^{*(k)}$ are computed with

$$\boldsymbol{L}(\boldsymbol{U}_l)\boldsymbol{U}^{*(k)} = \boldsymbol{V}^{*(k)}, \quad k = 2, \ldots, 4, \tag{38}$$

where $\boldsymbol{V}^{*(k)}$ are $(5-k)$th order approximations of the spatial derivatives of $\boldsymbol{V}$ at the boundary.

To obtain $\boldsymbol{V}^{*(k)}$, the following 1D WENO-type extrapolation is used [21]

$$V_m^{*(k)} = \sum_{r=0}^{4} \omega_r \left.\frac{d^k p_r(x)}{d_x^k}\right|_{x_b}, \tag{39}$$

where $p_r(x)$ is a polynomial approximation on different substencils and ω_r are the non-linear weights [21],

$$\omega_r = \frac{\alpha_r}{\sum_{s=0}^{4} \alpha_s}, \tag{40}$$

with

$$\alpha_r = \frac{d_r}{(1E - 6 + \beta_r)^3}, \tag{41}$$

where $d_0 = \Delta x^4$, $d_1 = \Delta x^3$, $d_2 = \Delta x^2$, $d_3 = \Delta x$, $d_4 = 1 - \sum_{r=0}^{3} d_r$, and β_r are the smoothness indicators [21]. As in [26], we use

$$\begin{aligned} &\beta_0 = \Delta x^2, \\ &\beta_r = \frac{1}{\sum_{k=0}^{r} f_k^2} \sum_{l=1}^{r} \int_{x_b - \Delta x/2}^{x_b + \Delta x/2} \left(\frac{d^l}{dx^l} p_r(x)\right)^2 dx, \end{aligned} \tag{42}$$

where $r = 1, \ldots, 4$ and f_k are the values used to build the polynomial p_r. We use $\beta_r = 0$ if $\sum_{k=0}^{r} f_k^2 = 0$ [26].

For the right boundary, all components shall be extrapolated

$$\boldsymbol{L}(\boldsymbol{U}_{N_x})\boldsymbol{U}^{*(k)} = \boldsymbol{V}^{*(k)}, \quad k = 0, \ldots, 4. \tag{43}$$

2.3 Quasi-one-dimensional nozzle flows

For the Q1D nozzle flows, $\xi = x$, $\boldsymbol{U}$, $\boldsymbol{F}(\boldsymbol{U})_x$, and $\boldsymbol{G}(\boldsymbol{U})_\eta$ are the same as for the simple periodic problem, and [10,25]

$$\boldsymbol{S}(\boldsymbol{U}) = -\frac{A_x}{A} \begin{bmatrix} \rho u \\ \rho u^2 \\ u(E + p) \end{bmatrix}, \tag{44}$$

where $A = A(x)$ is the nozzle area and A_x its derivative.

Designing nozzle profiles is a challenging step of nozzle simulation. Care must be taken to avoid the appearance of shocks and oscillations. In 2D problems, a proper way of designing a nozzle is through the characteristic method, as discussed in [8]. However, this is a discrete method and can present inconsistency if not used with finite difference calculations in the throat region [8]. Other possibility is to use experimental nozzle profiles, such as the classical $45° - 15°$ nozzle of [27,28]. We remark that the functions and its derivatives, which describe the nozzle contour, shall be smooth regarding the numerical scheme accuracy order. Otherwise, oscillations and other instabilities can arise.

For verification purpose, we use the idealized smooth nozzle profile of (45). The domain is discretized with (20) and the flow parameters are presented in Tab. 2.

$$r(x) = 0.01 \cos\left(\frac{3\pi x}{2x_r}\right) - 0.01 \tag{45}$$

Table 2: Q1D nozzle flows parameters

x_l (m)	0	T_0 (K)	800
x_r (m)	0.25	p_0 (Pa)	$1E+6$
p_r (Pa)	$8E+5$		

For the isentropic flow (smooth), the initial estimate for ρ, u, and p are computed with a linear distribution for M. From x_l to x_r, M vary linearly from 0 to 2. Once M_i for $i = 1, \dots, N_x$ are obtained, u can be computed with (26), and ρ and p with isentropic relations [8]

$$\frac{\rho_0}{\rho} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{1/(\gamma-1)}, \tag{46}$$

$$\frac{p_0}{p} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\gamma/(\gamma-1)}. \tag{47}$$

For the adiabatic flow (with shock), the initial conditions for ρ, u, and p are also computed with a linear distribution for M. From x_l to x_{th} M vary linearly from 0 to 1, and from x_{th} to x_r M vary linearly from 1 to 0. x_{th} is the the throat position, that is the lowest value of $r(x)$ position in $[x_l, x_r]$. Once M_i for $i = 1, \dots, N_x$ are obtained, u can be computed with (26), and ρ and p with (46) and (47). We remark that total pressure and density changes through the shock as explained next.

For the Q1D nozzle flows, M can be computed exactly through [8]

$$\left(\frac{A}{A^*}\right)^2 = \frac{1}{M^2}\left[\frac{2}{\gamma + 1}\left(1 + \frac{\gamma - 1}{2} M^2\right)\right]^{\frac{\gamma+1}{\gamma-1}}, \tag{48}$$

where A^* is the critical area. Notice that this equation can be solved by root finding methods. Furthermore, the standard Newton-Raphson and bisection methods present precision loss as $A \to A^*$. This can be avoided by using higher-precision computations, e.g., quadruple precision when the numerical solution is computed with double-precision.

For the isentropic flow, $A^* = A_{th}$ is the throat area for the entire flow. However, A^* changes through the shock because of its non-isentropic nature [8]. In the shock case, the Mach number at right boundary is [8]

$$\begin{aligned} M_r^2 = &- \frac{1}{\gamma - 1} + \left[\frac{1}{(\gamma - 1)^2}\right. \\ &\left. + \left(\frac{2}{\gamma - 1}\right)\left(\frac{2}{\gamma + 1}\right)^{\frac{\gamma+1}{\gamma-1}}\left(\frac{p_0 A_{th}}{p_r A_r}\right)^2\right]^{1/2}. \end{aligned} \tag{49}$$

With M_r, the total pressure downstream the shock can be obtained with (47) [8]. Through (47) and normal shock relations, the M immediately upstream and downstream the shock can be computed [8]

$$\frac{p_2}{p_1} = 1 + \frac{2\gamma}{\gamma + 1}(M_1^2 - 1), \tag{50}$$

$$M_2^2 = \frac{1 + [(\gamma - 1)/2] M_1^2}{\gamma M_1^2 - (\gamma - 1)/2}, \tag{51}$$

where 1 and 2 means immediately upstream and downstream the shock.

The critical area downstream the shock is computed with M_2 in (48). In summary, M is computed through (48) with A_{th} for the isentropic flow and with A_{th} upstream and A^* downstream the shock for the adiabatic flow. Again, u is computed with (26), and ρ and p with (46) and (47). $\boldsymbol{U}_l$ and $\boldsymbol{U}_r$ are computed with flow parameters and exact solution.

Since both inflow are subsonic and regarding (34), two boundary conditions shall be imposed. We use the exact solution, $\boldsymbol{U}^{*(0)} = \boldsymbol{U}_l$, to compute

$$(V_m)^{*(0)} = \boldsymbol{l}_m(\boldsymbol{U}_l)\boldsymbol{U}^{*(0)}, \quad m = 2, 3. \tag{52}$$

Then we update $\boldsymbol{U}^{*(0)}$ with

$$\boldsymbol{U}^{*(0)} = \boldsymbol{V}^{*(0)}\boldsymbol{R}(\boldsymbol{U}_l). \tag{53}$$

To obtain $\boldsymbol{U}^{*(1)}$, the first two equations of (37) are needed and the third is replaced with

$$l_{1,1}(\boldsymbol{U}_l)U_1^{(1)} + l_{1,2}(\boldsymbol{U}_l)U_2^{(1)} + l_{1,3}(\boldsymbol{U}_l)U_3^{(1)} = V_1^{*(1)}, \tag{54}$$

and the other $\boldsymbol{U}^{*(k)}$ are obtained with (38).

The isentropic outflow is supersonic and (43) can be used. The normal shock wave causes a non-isentropic velocity decrease, in a such way that the flow becomes subsonic downstream the shock. Hence, for the adiabatic flow right boundary we use the exact solution, $\boldsymbol{U}^{*(0)} = \boldsymbol{U}_r$, to compute

$$(V_1)^{*(0)} = \boldsymbol{l}_1(\boldsymbol{U}_r)\boldsymbol{U}^{*(0)}. \tag{55}$$

Then we update $\boldsymbol{U}^{*(0)}$ with

$$\boldsymbol{U}^{*(0)} = \boldsymbol{V}^{*(0)}\boldsymbol{R}(\boldsymbol{U}_r). \tag{56}$$

$\boldsymbol{U}^{*(1)}$ are obtained through

$$\begin{aligned} &U_2^{*(1)} = S_1(\boldsymbol{U}^{*(0)}), \\ &l_{2,1}(\boldsymbol{U}_r)U_1^{*(1)} + l_{2,2}(\boldsymbol{U}_r)U_2^{*(1)} + l_{2,3}(\boldsymbol{U}_r)U_3^{*(1)} = V_2^{*(1)}, \\ &l_{3,1}(\boldsymbol{U}_r)U_1^{*(1)} + l_{3,2}(\boldsymbol{U}_r)U_2^{*(1)} + l_{3,3}(\boldsymbol{U}_r)U_3^{*(1)} = V_3^{*(1)}, \end{aligned} \tag{57}$$

and the other $\boldsymbol{U}^{*(k)}$ are obtained through

$$\boldsymbol{L}(\boldsymbol{U}_r)\boldsymbol{U}^{*(k)} = \boldsymbol{V}^{*(k)}, \quad k = 2, \dots, 4. \tag{58}$$

Notice that $\boldsymbol{V}^{*(k)}$ are obtained with 1D WENO-type extrapolation.

2.4 Two-dimensional Ringleb flow

For the 2D Ringleb flow, $\xi = x$, $\eta = y$, $\boldsymbol{S}(\boldsymbol{U}) = \boldsymbol{0}$, [25]

$$\boldsymbol{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix}, \quad \boldsymbol{F}(\boldsymbol{U})_x = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ u(E+p) \end{bmatrix}, \quad \boldsymbol{G}(\boldsymbol{U})_y = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ v(E+p) \end{bmatrix}, \tag{59}$$

and the total energy is rewritten as [25]

$$E = \frac{p}{\gamma - 1} + \frac{\rho(u^2+v^2)}{2}. \tag{60}$$

The Ringleb flow is subsonic-supersonic and has exact solution by the hodograph method [29,30]. It is common in unstructured mesh verification and usually its domain is set between two streamlines, which are treated as solid walls [31,32]. Here, for simplicity and to test our modifications, we will define a simpler domain composed by three straight lines and one streamline at the left boundary, as shown in Fig. 2. The flow parameters are presented in Tab. 3, where l is a characteristic length, V is the absolute velocity, ψ represents the streamlines, and a_0 is the speed of sound computed with the total properties.

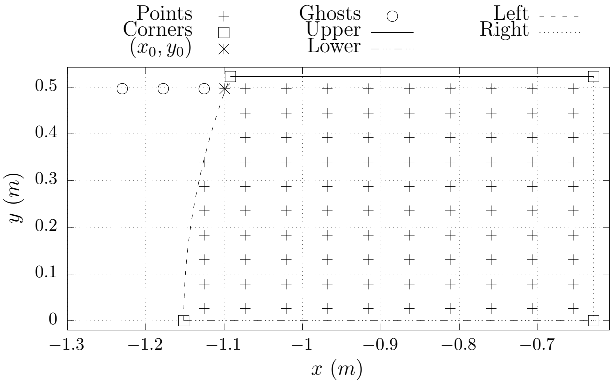

Fig. 2: Domain and mesh with 87 points for the 2D Ringleb flow

Table 3: 2D Ringleb flow parameters

l (m)	1	T_0 (K)	800
$\psi_{\min}$	$a_0 l$	p_0 (Pa)	$1E+6$
$\psi_{\max}$	$1.3 a_0 l$	ρ_0 (kg/m^3)	$p_0/(RT_0)$
$V_{\min}$ (m/s)	$0.7 a_0$		

The Ringleb flow solution is a particular case in which [30]

$$\psi = \frac{a_0^2 l}{V} \sin(\theta), \tag{61}$$

where θ is the velocity angle.

The flow properties and the coordinates (x and y) are obtained through (61) and the hodograph method, which can be summarized as [29–31]

$$\frac{x}{l} = \frac{1}{2}\frac{\rho_0}{\rho}\left(\frac{a_0^2}{V^2} - 2\frac{\psi^2}{l^2 a_0^2}\right) + \frac{L}{2}, \tag{62}$$

$$\frac{y}{l} = \pm\frac{\rho_0}{\rho}\frac{a_0}{V}\frac{\psi}{a_0 l}\sqrt{1 - \frac{V^2}{a_0^2}\left(\frac{\psi}{a_0 l}\right)^2}, \tag{63}$$

$$L = \frac{1}{\Gamma} + \frac{1}{3\Gamma^3} + \frac{1}{5\Gamma^5} - \frac{1}{2}\ln\left(\frac{1+\Gamma}{1-\Gamma}\right), \tag{64}$$

$$\Gamma = \sqrt{1 - 0.2\frac{V^2}{a_0^2}} = \left(\frac{\rho}{\rho_0}\right)^{1/5}, \tag{65}$$

$$\frac{y^2}{l^2} + \left(\frac{x}{l} - \frac{L}{2}\right)^2 = \left(\frac{a_0}{2V^2\Gamma^5}\right)^2 \tag{66}$$

$$p = p_0 \Gamma^7, \tag{67}$$

$$(u, v) = -V(\cos(\theta), \sin(\theta)). \tag{68}$$

The left lower domain corner is obtained with $\psi = \psi_{\max}$ and $y_{ll} = 0$, the right lower with $\psi = \psi_{\min}$ and $y_{rl} = 0$, the right upper is set as $(x_{ru}, y_{ru}) = (x_{rl}, y_{rl} + x_{rl} - x_{ll})$, and the left upper is obtained with $\psi = \psi_{\max}$ and $y_{lu} = y_{ru}$. The domain is discretized as follows

$$x_i = x_{ll} + \Delta x(i - 1/2), \quad y_j = y_{ll} + \Delta y(j - 1/2), \quad \Delta y = \Delta x = \frac{x_{rl} - x_{ll}}{N_x}, \tag{69}$$

with $1 \le i \le N_x$ and $1 \le j \le N_{y_i}$. We remark that the point quantity in the y-direction changes with respect to x-coordinate.

As initial estimate, we use

$$(U_1)_{i,j} = \rho_0, \quad (U_2)_{i,j} = 0, \quad (U_3)_{i,j} = 0, \quad (U_4)_{i,j} = \frac{p_0}{\gamma - 1}. \tag{70}$$

For simplicity, we use the exact solution at all boundaries except at the left where we use the sequel of ILW high order wall boundary treatment. To compute the ghost points with higher order, we need derivatives of the flow properties, in which we take implicit derivatives from (61) to (63) and (66).

For the upper boundary, we need to impose three conditions. With the exact solution and its first derivative, we compute $\boldsymbol{U}^{*(0)}$ and $\boldsymbol{U}^{*(1)}$. Then, for $m = 1, 2, 3$ and $k = 0, 1$

$$(V_m)^{*(k)} = \boldsymbol{l}_m(\boldsymbol{U}_u)\boldsymbol{U}^{*(k)}, \tag{71}$$

where the subscript u means at the upper boundary.

With a 1D WENO-type extrapolation on the y-direction, we compute the other $\boldsymbol{V}^{*(k)}$ and then

$$\boldsymbol{U}^{*(k)} = \boldsymbol{V}^{*(k)}\boldsymbol{R}(\boldsymbol{U}_u) \quad k = 0, \ldots, 4. \tag{72}$$

That is, we update $\boldsymbol{U}^{*(0)}$ and $\boldsymbol{U}^{*(1)}$ and compute the the other derivatives to insure proper boundary application. The values at the ghost points are computed with the Taylor expansion (35) in the y-direction.

The right boundary is similar to the upper and the lower is an outflow, one can use exact solution or simply

$$\begin{aligned} &\boldsymbol{U}_{i,-2} = \boldsymbol{U}_{i,3},\ \boldsymbol{U}_{i,-1} = \boldsymbol{U}_{i,2},\ \boldsymbol{U}_{i,0} = \boldsymbol{U}_{i,1}, \\ &(U_2)_{i,-2} = -(U_2)_{i,-2},\ (U_2)_{i,-1} = -(U_2)_{i,-1}, \\ &(U_2)_{i,0} = -(U_2)_{i,0}. \end{aligned} \tag{73}$$

The left boundary is a streamline and can be treated as a solid wall. The original ILW procedure uses a Taylor expansion in the normal direction around the point (x_0, y_0) [5]. However, this can demand points outside the domain, as depicted in Fig. 1. To avoid that, we propose to choose (x_0, y_0) in a way that the x- or y-coordinate remains fixed. For instance, (x_0, y_0) in Fig. 2 was chosen in a way that the y-coordinate is the same for the three left boundary ghost points. This new form of choosing (x_0, y_0) requires slight modifications to the original algorithm, as showed next.

With the exception of problems with analytical or exact solution, at a solid wall there is a lack of information to impose boundary conditions. To compute an extrapolation in x-direction, one shall assess the eigenvalues to determine how many conditions shall be imposed. For this case, [25]

$$\lambda_1 = u - a, \quad \lambda_2 = u, \quad \lambda_3 = u, \quad \lambda_4 = u + a. \tag{74}$$

If one has $a > u > 0$, three boundary conditions shall be imposed, and at the solid wall one only has the normal velocity, which is zero. Other situations may require different boundary condition quantity, i.e., for $u > a > 0$ none boundary condition shall be imposed, and a Taylor expansion in the x-direction together with an extrapolation would be enough. Here, we present a more general procedure that covers the lack of information in a solid wall.

Our goal is to choose (x_0, y_0) to perform a Taylor expansion in the x- or y-direction. At the 2D Ringleg flow left boundary, we need to prescribe one boundary condition ($u < 0$ and $|u| < a$) and to perform a Taylor expansion in the x-direction. We start the discussion with the first derivative for the primitive variables, $\boldsymbol{W} = (\rho, u, v, p)^T$,

$$\begin{aligned} &W_2^{*(0)}W_1^{*(1)} + W_3^{*(0)}\rho_y + W_1^{*(0)}(W_2^{*(1)} + v_y) = 0, \\ &(V_m)^{*(1)} = \boldsymbol{l}_m(\boldsymbol{W}_n)\boldsymbol{W}^{*(1)}, \quad m = 1, 2, 3, \end{aligned} \tag{75}$$

which yields a linear system with four equations and six unknowns ($\boldsymbol{W}^{*(1)}$ plus ρ_y and v_y) and n means the nearest point to (x_0, y_0).

Even in the case were only one boundary condition should be imposed, we need additional information for y-direction derivatives. For a solid wall, we have information in the normal direction, $\hat{x}$, and we can use directional derivatives to provide the additional information

$$\boldsymbol{W}_y = \frac{\boldsymbol{W}_{\hat{x}} - \cos(\theta)\boldsymbol{W}_x}{\sin(\theta)}. \tag{76}$$

As in [21], we also use rotation but we only compute the rotated values and its first derivatives. Then, we rotate back and use the result to compute the Taylor expansion in the x-direction.

After (x_0, y_0) is chosen, we compute the angle, θ, between the normal pointing outwards the boundary and the x-direction, and form a local coordinate system [21]

$$\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \boldsymbol{T} \begin{pmatrix} x \\ y \end{pmatrix}, \tag{77}$$

The Euler equations are rewritten as [21]

$$\hat{\boldsymbol{U}}_t + \boldsymbol{F}(\hat{\boldsymbol{U}})_{\hat{x}} + \boldsymbol{G}(\hat{\boldsymbol{U}})_{\hat{y}} = \boldsymbol{0}, \tag{78}$$

$$\hat{\boldsymbol{U}} = \begin{bmatrix} \rho \\ \rho\hat{u} \\ \rho\hat{v} \\ E \end{bmatrix}, \quad \begin{bmatrix} \hat{u} \\ \hat{v} \end{bmatrix} = \boldsymbol{T} \begin{bmatrix} u \\ v \end{bmatrix}. \tag{79}$$

We use primitive variables because it gives simpler equations and we have the velocity and pressure first derivative in the normal direction. The conservative variables could also be used as commented in [5,21]. The primitive variables rotated vector is $\hat{\boldsymbol{W}} = (\rho, \hat{u}, \hat{v}, p)^T$.

In the normal direction the velocity is zero, $\hat{u} = 0$. That is, one boundary condition shall be imposed [21]

$$\begin{aligned} &\hat{W}_2^{*(0)} = 0, \\ &(\hat{V}_m)^{*(0)} = \boldsymbol{l}_m(\hat{\boldsymbol{W}}_n)\hat{\boldsymbol{W}}^{*(0)}, \quad m = 2, 3, 4, \end{aligned} \tag{80}$$

which yields a linear system with four equations and four unknowns.

For the first derivative, $\hat{W}_4^{*(1)} = p_{\hat{x}} = \rho\hat{v}^2\kappa$, with κ being the curvature, as stated in [21]. However, x and y for the 2D Ringleb flow are given by (62) and (63). For simplicity, in this case we compute the pressure derivative in the normal direction through the exact solution and directional derivative

$$\begin{aligned} &\hat{W}_4^{*(1)} = p_x\cos(\theta) + p_y\sin(\theta) = p_{\hat{x}}, \\ &(\hat{V}_m)^{*(1)} = \boldsymbol{l}_m(\hat{\boldsymbol{W}}_n)\hat{\boldsymbol{W}}^{*(1)}, \quad m = 2, \ldots, 4. \end{aligned} \tag{81}$$

With $\hat{\boldsymbol{W}}^{*(0)}$ and $\hat{\boldsymbol{W}}^{*(1)}$, we rotate back to obtain $\boldsymbol{W}^{*(0)}$ and its derivative in the normal direction. Replacing (76) in (75), yields

$$\begin{aligned}&\left[W_2^{*(0)} - \frac{W_3^{*(0)}\cos(\theta)}{\sin(\theta)}\right] W_1^{*(1)} + W_1^{*(0)} W_2^{*(1)} \\ &- W_1^{*(0)}\frac{\cos(\theta)}{\sin(\theta)} W_3^{*(1)} = -\frac{W_3^{*(0)}\rho_{\hat{x}} + W_1^{*(0)} v_{\hat{x}}}{\sin(\theta)}, \qquad (82)\\ &(V_m)^{*(1)} = \boldsymbol{l}_m(\boldsymbol{W}_n)\boldsymbol{W}^{*(1)}, \quad m = 2,\dots,4.\end{aligned}$$

For the other derivatives,

$$\boldsymbol{W}^{*(k)} = \boldsymbol{R}(\boldsymbol{W}_n)\boldsymbol{V}^{*(k)}, \quad k = 2,\dots,4. \qquad (83)$$

Now, $\boldsymbol{W}^{*(k)}$ are available at the boundary and we use a Taylor expansion in the x-direction to compute the primitive variables vectors at the three ghost points of Fig. 2. Then, we transform those vectors from primitive to conservative.

We remark that in the original ILW procedure one needs to find three distinct (x_0, y_0) points for the three ghost points of Fig. 2, and this would require three 2D WENO-type interpolations (see also Fig. 1). In our modification, only two 2D WENO-type are needed.

The $\boldsymbol{V}^{*(k)}$ and $\hat{\boldsymbol{V}}^{*(k)}$ needed for the left boundary systems of equations are obtained through 2D WENO-type interpolation, in which we construct least squares polynomials approximations as in [21],

$$p_r(x,y) = \sum_{0\le m+l\le r} a_{lm} z^l r^m, \qquad (84)$$

satisfying $p_r(x_{i,j}, y_{i,j}) \approx (V_m)_{i,j}$ with i, j meaning the stencil points.

In the original 2D WENO-type extrapolation, the stencil points are chosen from 1D substencils near (x_0, y_0) [5], which may require complicated algorithms depending on the boundaries and the availability of interior points. Here, we start constructing the stencils with the nearest $(r+1)^2$ points. Then, we approximate the rank of the linear system matrix. If the rank is deficient, we add the next nearest point to (x_0, y_0) in the stencil and repeat the approximation. When the matrix becomes full rank, we stop adding points to the stencil. One should note that for *fixed boundaries*, the stencils for each (x_0, y_0) can be computed with the mesh generation and will remains fixed until the simulation is over.

With the stencils, we compute [5, 21]

$$(V_m)^{*(k)} = \sum_{r=0}^{4} \omega_r \left.\frac{\partial^k}{\partial \hat{x}^k} p_r(x,y)\right|_{x_0,y_0}, \qquad (85)$$

$$\omega_r = \frac{\alpha_r}{\sum_{s=0}^{4}\alpha_s}, \qquad (86)$$

$$\alpha_r = \frac{d_r}{(1E-6+\beta_r)^3}, \qquad (87)$$

where $d_0 = 2\Delta x^4$, $d_1 = 2\Delta x^3$, $d_2 = 2\Delta x^2$, $d_3 = 2\Delta x$, and $d_4 = 1 - \sum_{r=0}^{3} d_r$ [21]. The smoothness indicators are obtained with [21, 26]

$$\begin{aligned}\beta_0 &= 2\Delta x^2,\\ \beta_r &= \frac{1}{\sum_{k=0}^{r} p_k^2}\sum_{1\le|\alpha|\le r}\int_K |K|^{|\alpha|-1}\left[D^\alpha p_r(x,y)\right]^2 dxdy,\end{aligned} \qquad (88)$$

where $r = 1,\dots,4$, α is a multi-index and $K = [x_0 - \Delta x/2, x_0 + \Delta x/2] \times [y_0 - \Delta x/2, y_0 + \Delta x/2]$ as shown in [21]. As an example, for $r = 1$, $\alpha = (0,1)$ and $\alpha = (1,0)$.

The original 2D WENO-type extrapolation weights were improved in [26] for q_r interpolation. Here, we changed the factor before the integral in (88) to handle least squares and we take $\beta_r = 0$ if $\sum_{k=0}^{r} p_k^2 = 0$.

2.5 Two-dimensional conical flow

For the 2D conical flow, $\xi = z$, $\eta = r$, $\boldsymbol{U}$ is the same as for the 2D Ringleb flow, $\boldsymbol{F}(\boldsymbol{U})_z = \boldsymbol{F}(\boldsymbol{U})_x$, $\boldsymbol{G}(\boldsymbol{U})_r = \boldsymbol{G}(\boldsymbol{U})_y$, and [25]

$$\boldsymbol{S}(\boldsymbol{U}) = \begin{bmatrix} -\rho v/r \\ -\rho uv/r \\ -\rho v^2/r \\ -v(E+p)/r \end{bmatrix}. \qquad (89)$$

We set $\Delta r = \Delta z$ and discretize the domain with (20) in both directions, which results the mesh presented in Fig. 3.

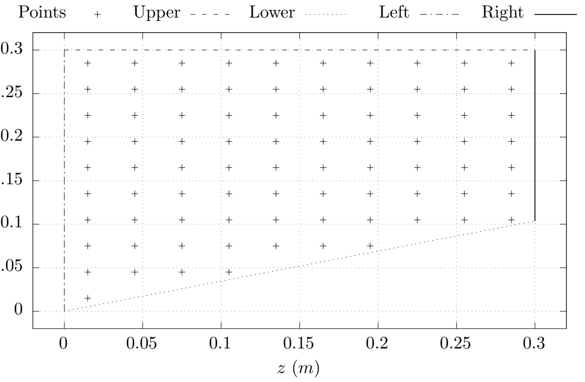

Fig. 3: Domain and mesh for the 2D conical flow

The flow parameters are presented in Tab. 4, where ∞ means free-stream properties and θ_s the oblique shock inclination.

Table 4: 2D conical flow parameters

z_l (m)	0	T_∞ (K)	800
z_r (m)	0.3	p_∞ (Pa)	$5E+5$
r_u (m)	0.3	M_∞	2
θ_s $(°)$	37		

The lower boundary is the cone surface and has an inclination of θ_c, which causes the mesh point quantity to change. The initial conditions are the free-stream properties repeated in each mesh point.

The conical flow has a 1D exact solution in spherical coordinates. We will follow the inverse approach as presented in [8]. That is, for a given free-stream Mach number and a shock inclination, we will find a correspondent cone surface inclination. With oblique shock relations [8],

$$M_{n1} = M_1 \sin(\phi_s), \tag{90}$$

$$M_{n2}^2 = \frac{M_{n1}^2 + [2/(\gamma-1)]}{[2\gamma/(\gamma-1)]M_{n1}^2 - 1}, \tag{91}$$

$$\tan(\delta) = 2\cot(\phi_s)\left[\frac{M_1^2 \sin^2(\phi_s) - 1}{M_1^2[\gamma + \cos(2\phi_s)] + 2}\right], \tag{92}$$

$$M_2 = \frac{M_{n2}}{\sin(\phi_s - \delta)} \tag{93}$$

$$\frac{\rho_2}{\rho_1} = \frac{(\gamma+1)M_{n1}^2}{(\gamma-1)M_{n1}^2 + 2}, \tag{94}$$

$$\frac{p_2}{p_1} = 1 + \frac{2\gamma}{\gamma+1}(M_{n1}^2 - 1), \tag{95}$$

we compute the Mach number immediately behind the shock, M_2, and the flow deflection angle, δ.

From M_2, we compute the dimensionless velocity as [8]

$$V' = \frac{V}{V_{\max}} = \left[\frac{2}{(\gamma-1)M^2} + 1\right]^{-1/2}, \tag{96}$$

and the dimensionless velocities in r- and θ-directions with [8]

$$v_r' = V' \cos(\phi_s - \delta), \tag{97}$$

$$v_\theta' = -V' \sin(\phi_s - \delta). \tag{98}$$

Using v_r' as a boundary value, we solve [8]

$$\begin{aligned}&\frac{\gamma-1}{2}\left[1 - v_r'^2 - \left(\frac{dv_r'}{d\phi}\right)^2\right]\left[2v_r' + \frac{dv_r'}{d\phi}\cot(\phi) + \frac{d^2v_r'}{d\phi^2}\right] \\ &- \frac{dv_r'}{d\phi}\left[v_r'\frac{dv_r'}{d\phi} + \frac{dv_r'}{d\phi}\frac{d^2v_r'}{d\phi^2}\right] = 0.\end{aligned} \tag{99}$$

When $v_\theta' \approx 0$, the respective θ will represent θ_c and the computation reached its surface.

Once θ_c is reached, we compute the flow properties with (46), (47), and [8]

$$\frac{V_{\max}^2}{2} = \frac{a^2}{\gamma-1} + \frac{V^2}{2} = \frac{\gamma p}{\rho(\gamma-1)} + \frac{V^2}{2}. \tag{100}$$

Since $V_{\max}$ is a constant for the flow [8], it can be determined with free-stream properties and used to compute V behind the shock.

Notice that (99) need to be modified before solving with standard Runge-Kutta methods. Writing (99) as a system of equations [33]

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} v_r \\ \dfrac{dv_r'}{d\theta} \end{bmatrix}, \tag{101}$$

a standard Runge-Kutta method can be used to solve for [33]

$$\begin{aligned} y' &= f(\theta, y) = \begin{bmatrix} \dfrac{dv_r'}{d\theta} \\ \dfrac{d^2v_r'}{d\theta^2} \end{bmatrix}, \\ y' &= \begin{bmatrix} y_2 \\ \dfrac{y_2^2 y_1 - \frac{\gamma-1}{2}(1 - y_1^2 - y_2^2)[2y_1 + y_2\cot(\theta)]}{\frac{\gamma-1}{2}(1 - y_1^2 - y_2^2) - y_2^2} \end{bmatrix}. \end{aligned} \tag{102}$$

For this problem, the lower boundary is a solid wall that requires (z_0, r_0) outside the domain with the original ILW procedure. Therefore, we can use the same procedure as for the 2D Ringleb flow. At this boundary we need to prescribe three boundary conditions ($a > v > 0$) and to perform a Taylor expansion in the y-direction. For the first derivative, we have

$$\begin{aligned} &W_2^{*(0)}\rho_x + W_3^{*(0)}W_1^{*(1)} + \\ &W_1^{*(0)}(u_x + W_3^{*(1)}) = -\frac{W_1^{*(0)}W_3^{*(0)}}{r_0}, \\ &W_2^{*(0)}u_x + W_3^{*(0)}W_2^{*(1)} + \frac{1}{W_1^{*(0)}}p_x = 0, \\ &W_2^{*(0)}v_x + W_3^{*(0)}W_3^{*(1)} + \frac{1}{W_1^{*(0)}}W_4^{*(1)} = 0, \\ &(V_4)^{*(1)} = \boldsymbol{l}_4(\boldsymbol{W}_n)\boldsymbol{W}^{*(1)}. \end{aligned} \tag{103}$$

We need more information for the boundary values and its first derivatives, and we again provide the additional information based on the normal direction and directional derivative

$$\boldsymbol{W}_x = \frac{\boldsymbol{W}_{\hat{x}} - \sin(\theta)\boldsymbol{W}_y}{\cos(\theta)}. \tag{104}$$

To compute the rotated values at the boundary we use (80). For the rotated first derivatives, we replace the first equation of (81) with $\hat{W}_4^{*(1)} = \hat{p}_x = \rho\hat{v}^2\kappa = 0$ because $\kappa = 0$ for straight lines.

With $\hat{\boldsymbol{W}}^{*(0)}$ and $\hat{\boldsymbol{W}}^{*(1)}$, we rotate back to obtain $\boldsymbol{W}^{*(0)}$ and its derivative in the normal direction. Replacing (104) in (103), yields

$$\begin{aligned}
&\left[W_3^{*(0)} - W_2^{*(0)}\tan(\theta)\right] W_1^{*(1)} - W_1^{*(0)}\tan(\theta) W_2^{*(1)} + \\
&W_1^{*(0)} W_3^{*(1)} = -\frac{W_2^{*(0)}\rho_{\hat{x}} + W_1^{*(0)} u_{\hat{x}}}{\cos(\theta)} - \frac{W_1^{*(0)} W_3^{*(0)}}{r_0}, \\
&\left[W_3^{*(0)} - W_2^{*(0)}\tan(\theta)\right] W_2^{*(1)} \\
&-\frac{\tan(\theta)}{W_1^{*(0)}} W_4^{*(1)} = -\frac{W_2^{*(0)} u_{\hat{x}} + \frac{p_{\hat{x}}}{W_1^{*(0)}}}{\cos(\theta)}, \\
&\left[W_3^{*(0)} - W_2^{*(0)}\tan(\theta)\right] W_3^{*(1)} \\
&+\frac{1}{W_1^{*(0)}} W_4^{*(1)} = -\frac{W_2^{*(0)} v_{\hat{x}}}{\cos(\theta)}, \\
&(V_4)^{*(1)} = \boldsymbol{l}_4(\boldsymbol{W}_n)\boldsymbol{W}^{*(1)}.
\end{aligned} \tag{105}$$

The other $\boldsymbol{W}^{*(k)}$ derivatives are computed with (83).

Because of the domain, there is no flow across the upper boundary and we can simply use

$$\boldsymbol{U}_{Nr+1,i} = \boldsymbol{U}_\infty, \quad \boldsymbol{U}_{Nr+2,i} = \boldsymbol{U}_\infty, \quad \boldsymbol{U}_{Nr+3,i} = \boldsymbol{U}_\infty. \tag{106}$$

For the left boundary, all boundary conditions shall be imposed because the flow is supersonic ($u > a > 0$),

$$\boldsymbol{U}^{*(0)} = \boldsymbol{U}_\infty. \tag{107}$$

For $\boldsymbol{U}^{*(1)}$ we use the following system of equations

$$\begin{aligned}
&U_2^{*(1)} = 0, \\
&\left[\frac{U_2^2}{U_1} + (\gamma - 1)\left(U_4 - \frac{U_2^2 + U_3^2}{2U_1}\right)\right]_x = 0, \\
&\left(\frac{U_2 U_3}{U_1}\right)_x = 0, \\
&\left\{\frac{U_2}{U_1}\left[U_4 + (\gamma - 1)\left(U_4 - \frac{U_2^2 + U_3^2}{2U_1}\right)\right]\right\}_z = 0.
\end{aligned} \tag{108}$$

For the other derivatives,

$$\boldsymbol{L}(\boldsymbol{U}_\infty)\boldsymbol{U}^{*(k)} = \boldsymbol{V}^{*(k)}, \quad k = 2, \ldots, 4, \tag{109}$$

where $\boldsymbol{V}^{*(k)}$ are obtained with the 2D WENO-type extrapolation.

Finally, at the right boundary the flow is supersonic ($u > a > 0$) and we perform a 1D WENO-type extrapolation to compute $\boldsymbol{V}^{*(k)}$, and

$$\boldsymbol{U}^{*(k)} = \boldsymbol{R}(\boldsymbol{U}_{N_z,j})\boldsymbol{V}^{*(k)}, \quad k = 0, \ldots, 4. \tag{110}$$

3 Results

3.1 Simple periodic problem

The simple periodic problem is useful for verifying the core solver accuracy. The boundary treatment is straightforward, which ease code debugging. We are mainly interested in the accuracy results shown in Tab. 5, where we can see that the designed accuracy is reached. Note that we focus on the density because the other conservative variables behaves in a similar manner.

3.2 One-dimensional Rayleigh flow

In the 1D Rayleigh flow we added a more robust way to treat boundary conditions while the interior scheme remained the same, with the exception of the source term. This allow us to concentrate on the boundary treatment. Notice that this problem is steady and there are many ways of setting the initial conditions. We use the left boundary values as initial estimate, which are more generic than the analytical solution, in the first few meshes. The latter is used in the remaining meshes because it gives faster results. We remark that one still needs the numerical procedure, since the analytical solution is *not* a solution for the numerical scheme.

The accuracy analysis is shown in Tab. 6 for the Mach number since it involves all the conservative variables, where we can see that the designed accuracy is being reached.

3.3 Quasi-one-dimensional nozzle flow

The Q1D flows are subsonic-supersonic flows in an idealized convergent-divergent nozzle. The subsonic inflow is isentropically compressed in the convergent region of the nozzle. At the throat, the flow becomes sonic and it is isentropically expanded downstream until the nozzle exit (isentropic flow) or until the shock (adiabatic flow). Downstream the shock, the flow becomes subsonic. Since we already verified most of the boundary treatment, it is easy now to compute the Q1D flows.

The accuracy analysis is shown in Tab. 7 for the Mach number, where we can see that the designed accuracy is being reached for the isentropic flow. The accuracy behavior for the adiabatic flow is due to the shock and it is the expected behavior in flows with discontinuities [13,14].

The Mach number profiles for isentropic and adiabatic flow are shown in Figs. 4 and 5, where we can see good agreement between the numerical and exact solutions and that the high-resolution is being attained,

Table 5: L^1 and L^∞ norms and its orders for density and simple periodic problem

Points	Δx	L^1	Order	L^∞	Order
20	5.00E-02	2.08E-04	-	3.75E-04	-
40	2.50E-02	6.37E-06	5.03	1.14E-05	5.04
80	1.25E-02	1.88E-07	5.08	3.81E-07	4.91
160	6.25E-03	5.72E-09	5.04	1.20E-08	4.99
320	3.13E-03	1.78E-10	5.00	3.68E-10	5.03

Table 6: L^1 and L^∞ norms and its orders for Mach number and 1D Rayleigh flow

Points	Δx	L^1	Order	L^∞	Order
20	1.00E-02	5.88E-07	-	2.05E-06	-
40	5.00E-03	2.94E-08	4.32	9.95E-08	4.36
80	2.50E-03	1.24E-09	4.57	3.98E-09	4.64
160	1.25E-03	4.55E-11	4.77	1.42E-10	4.81
320	6.25E-04	1.53E-12	4.89	4.73E-12	4.90

Table 7: L^1 and L^∞ norms and its orders for Mach number and Q1D nozzle flow

		Isentropic flow				Adiabatic flow			
Points	Δx	L^1	Order	L^∞	Order	L^1	Order	L^∞	Order
20	1.25E-02	2.70E-04	-	8.00E-04	-	4.36E-02	-	1.75E-01	-
40	6.25E-03	1.20E-05	4.49	1.76E-05	5.51	2.72E-02	0.68	2.58E-01	-0.57
80	3.13E-03	3.66E-07	5.03	3.50E-07	5.65	1.57E-02	0.79	3.44E-01	-0.41
160	1.56E-03	1.04E-08	5.13	8.08E-09	5.44	6.62E-03	1.24	1.88E-01	0.87
320	7.81E-04	4.30E-10	4.83	2.55E-10	5.05	3.33E-03	0.99	2.15E-01	-0.19

even with the shock near the boundary in the 20 points mesh.

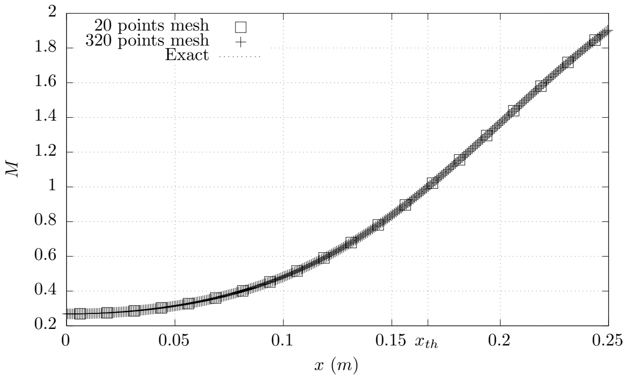

Fig. 4: Mach number profile for Q1D isentropic nozzle flow

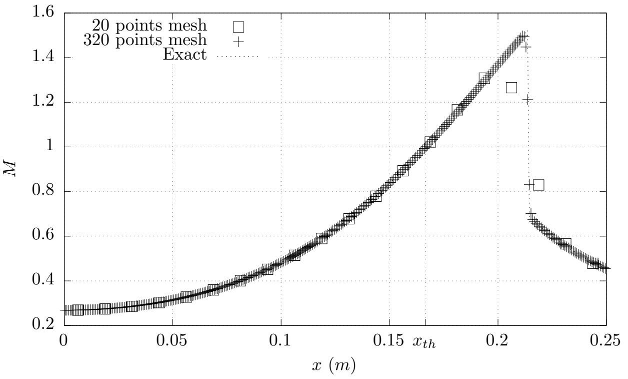

Fig. 5: Mach number profile for Q1D adiabatic nozzle flow

3.4 Two-dimensional Ringleb flow

The 2D Ringleb flow is a subsonic-supersonic flow that depends on streamlines. In this particular case, our left boundary is a streamline, which was modeled by a curved wall. We presented the procedure to solve flows with arbitrary domains that are not aligned with the mesh points. The 1D x-direction space discretization is also carried in the y-direction with a few adjustments to the whole solver, which saves time and effort. Also, the general idea of 1D ILW is extended to 2D along with the modified solid wall treatment and 2D WENO-type extrapolation.

The accuracy analysis is shown in Tab. 8 for the Mach number, where we can see that the designed accuracy is being reached.

The Mach number color map is presented in Fig. 6, where we see that the supersonic region is located at the right most region of the flow. By changing $\psi_{\max}$ and $\psi_{\min}$, one can change the region size as needed. For the mixed subsonic-supersonic lower boundary presented

Table 8: L^1 and L^∞ norms and its orders for Mach number and 2D Ringleb flow

N_x	Points	$\Delta x = \Delta y$	L^1	Order	L^∞	Order
10	97	5.23E-02	1.32E-06	-	4.25E-06	-
20	385	2.61E-02	5.76E-08	4.52	3.00E-07	3.83
40	1541	1.31E-02	1.66E-09	5.12	2.45E-08	3.61
80	6160	6.54E-03	5.35E-11	4.95	1.09E-09	4.49
160	24635	3.27E-03	1.27E-12	5.40	2.49E-11	5.45

here, it is easier to apply the boundary conditions regarding (73). However, since other flow problems could present mixed subsonic-supersonic boundaries this is a good case to test the procedure.

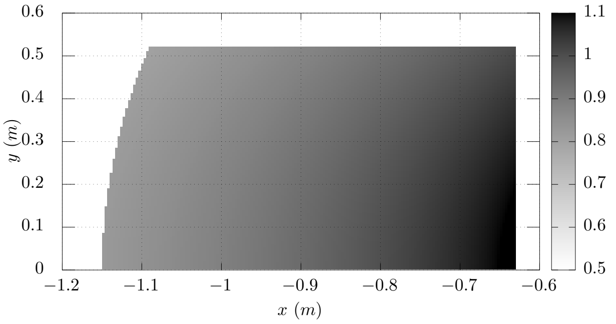

Fig. 6: Mach number color map for the 2D Ringleb flow and 24635 points mesh

3.5 Two-dimensional conical flow

In the 2D conical flow we did not add any additional procedure to handle the oblique shock wave since the WENO-Z+ and WENO-type extrapolation are robust and designed for that. The shock is attached to the tip of the cone with an inclination of 37 ° and, differently from the Q1D case, the flow downstream the shock is supersonic. Still, there is a non-isentropic velocity decrease through the shock.

The accuracy analysis is shown in Tab. 9 for the Mach number, where we can see that the designed accuracy is not being reached, which is the same situation as in the Q1D adiabatic flow. However, high-resolution is still attained.

The Mach number color map is presented in Fig. 7, where we see the oblique shock, which is less severe when compared to the normal shock and the flow is supersonic everywhere.

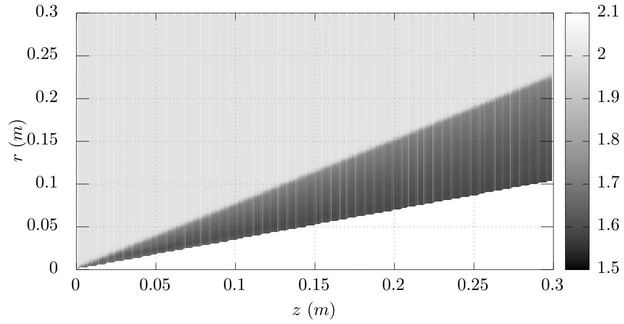

Fig. 7: Mach number color map for the 2D conical flow and 21016 points mesh

4 Concluding remarks

Because of non-linear phenomena, compressible fluid flows requires robust solvers, and CFD problems usually have arbitrary domains. In contrast to unstructured mesh solvers, the ILW allows the mesh to be structured and simple, which does not require modifications on the numerical schemes. In this work, we reviewed the most recent and popular methods for solving compressible fluid flows with arbitrary domains, presented detailed information regarding the numerical code construction and verification, and proposed modifications in the ILW solid wall treatment and 2D WENO-type stencil selection and weights.

In all smooth problems, we showed that designed accuracy was reached. In problems with shocks, the designed order is not being reached for the whole domain, which is the expected behavior of high-order numerical schemes such as WENO. However, high-resolution was still attained. When compared to smooth problems, no modifications were needed to handle the shock wave. The ILW and WENO performance are good even with dissipative numerical splitting and shocks, as one can notice from coarse mesh in the Q1D adiabatic flow. Our modifications showed similar behavior to the original ILW procedure while providing a more generic boundary treatment. With the methods, procedures, and the detailed information presented here, one can build numerical codes to solve compressible and inviscid fluid

Table 9: L^1 and L^∞ norms and its orders for Mach number and 2D conical flow

N_z	Points	$\Delta z = \Delta r$	L^1	Order	L^∞	Order
10	72	3.00E-02	7.52E-02	-	4.33E-01	-
20	310	1.50E-02	4.05E-02	8.91E-01	5.33E-01	-3.02E-01
40	1282	7.50E-03	2.21E-02	8.73E-01	5.23E-01	2.84E-02
80	5214	3.75E-03	1.02E-02	1.11E+00	5.34E-01	-3.09E-02
160	21016	1.87E-03	5.06E-03	1.02E+00	5.41E-01	-1.84E-02

flows with high-resolution, arbitrary domains, and with further adjustments, viscous flows can also be computed.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. P. Roe, *A Brief Introduction to High-Resolution Schemes*, pp. 9–28. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997.
2. C.-W. Shu, “Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws,” in *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations* (A. Quarteroni, ed.), pp. 325–432, Berlin, Heidelberg: Springer Berlin Heidelberg, 1998.
3. R. J. Leveque, *Finite Volume Methods for Hyperbolic Problems.* New York: Cambridge texts in applied mathematics, 1 ed., 2002.
4. C. Hirsch, *Numerical computation of internal and external flows.* Elsevier, 2 ed., 2007.
5. S. Tan and C.-W. Shu, “Inverse lax-wendroff procedure for numerical boundary conditions of conservation laws,” *Journal of Computational Physics*, vol. 229, pp. 8144–8166, 2010.
6. F. Acker, R. B. R. Borges, and B. Costa, “An improved weno-z scheme,” *Journal of Computational Physics*, vol. 313, pp. 726–753, 2016.
7. F. Fambri, “Discontinuous galerkin methods for compressible and incompressible flows on space–time adaptive meshes: Toward a novel family of efficient numerical methods for fluid dynamics,” *Archives of Computational Methods in Engineering*, vol. 27, pp. 199–283, Jan 2020.
8. J. D. Anderson, *Modern Compressible Flow.* New York: McGraw-Hill, 3 ed., 2003.
9. J. E. John and T. G. Keith, *Gas Dynamics.* New Jersey: Prentice Hall, 3 ed., 2006.
10. G.-S. Jiang and C.-W. Shu, “Efficient implementation of weighted ENO schemes,” *Journal of Computational Physics*, vol. 126, no. 1, pp. 202 – 228, 1996.
11. A. K. Henrick, T. D. Aslam, and J. M. Powers, “Mapped weighted essentially non-oscillatory schemes: Achieving optimal order near critical points,” *Journal of Computational Physics*, vol. 207, no. 2, pp. 542 – 567, 2005.
12. R. Borges, M. Carmona, B. Costa, and W. S. Don, “An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws,” *Journal of Computational Physics*, vol. 227, pp. 3191–3211, 2008.
13. S. Gottlieb, D. Gottlieb, and C.-W. Shu, “Recovering high-order accuracy in weno computations of steady-state hyperbolic systems,” *Journal of Scientific Computing*, vol. 28, pp. 307–318, Sep 2006.
14. R. Zhang, M. Zhang, and C.-W. Shu, “On the order of accuracy and numerical performance of two classes of finite volume weno schemes,” *Communications in Computational Physics*, vol. 9, no. 3, p. 807–827, 2011.
15. Ásdís Helgadóttir, Y. T. Ng, C. Min, and F. Gibou, “Imposing mixed dirichlet–neumann–robin boundary conditions in a level-set framework,” *Computers & Fluids*, vol. 121, pp. 68 – 80, 2015.
16. X. Wu, J.-y. Shi, H. Lei, Y.-p. Li, and L. Okine, “Analytical solutions of transient heat conduction in multilayered slabs and application to thermal analysis of landfills,” *Journal of Central South University*, vol. 26, pp. 3175–3187, Nov 2019.
17. G. Chesshire and W. Henshaw, “Composite overlapping meshes for the solution of partial differential equations,” *Journal of Computational Physics*, vol. 90, no. 1, pp. 1 – 64, 1990.
18. K. Sebastian and C.-W. Shu, “Multidomain weno finite difference method with interpolation at subdomain interfaces,” *Journal of Scientific Computing*, vol. 19, pp. 405–438, Dec 2003.
19. H.-O. Kreiss and N. A. Petersson, “A second order accurate embedded boundary method for the wave equation with dirichlet data,” *SIAM Journal on Scientific Computing*, vol. 27, no. 4, pp. 1141–1167, 2006.
20. S. Nilsson, N. A. Petersson, B. Sjögreen, and H.-O. Kreiss, “Stable difference approximations for the elastic wave equation in second order formulation,” *SIAM Journal on Numerical Analysis*, vol. 45, no. 5, pp. 1902–1936, 2007.
21. 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*, vol. 231, pp. 2510–2527, 2012.
22. J. OLEJNICZAK, M. J. WRIGHT, and G. V. CANDLER, “Numerical study of inviscid shock interactions on double-wedge geometries,” *Journal of Fluid Mechanics*, vol. 352, pp. 1–25, 1997.
23. F. Rispoli, R. Saavedra, A. Corsini, and T. E. Tezduyar, “Computation of inviscid compressible flows with the v-sgs stabilization and y$\zeta\beta$ shock-capturing,” *International Journal for Numerical Methods in Fluids*, vol. 54, no. 6–8, pp. 695–706, 2007.
24. M. Visbal, “Viscous and inviscid interactions of an oblique shock with a flexible panel,” *Journal of Fluids and Structures*, vol. 48, pp. 27–45, 2014.
25. E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction.* Springer Berlin Heidelberg, 3 ed., 2009.
26. F. Filbet and C. Yang, “An inverse lax–wendroff method for boundary conditions applied to boltzmann type models,” *Journal of Computational Physics*, vol. 245, pp. 43 – 61, 2013.

27. L. H. Back, P. F. Massier, and H. L. Gier, "Comparison of measured and predicted flows through conical supersonic nozzles, with emphasis on the transonic region," *AIAA Journal*, vol. 3, no. 9, pp. 1606–1614, 1965.
28. R. C. O. SERRA, "Determination of internal gas flows by a transient numerical technique," *AIAA Journal*, vol. 10, no. 5, pp. 603–611, 1972.
29. A. H. Shapiro, *The Dynamics and Thermodynamics of Compressible Fluid Flow*, vol. 1. John Wiley & Sons, 1953.
30. A. H. Shapiro, *The Dynamics and Thermodynamics of Compressible Fluid Flow*, vol. 2. The Ronald Press Company, 1954.
31. K. Kitamura and E. Shima, "Simple and parameter-free second slope limiter for unstructured grid aerodynamic simulations," *AIAA Journal*, vol. 50, no. 6, pp. 1415–1426, 2012.
32. C. Breviglieri and J. ao Luiz F. Azevedo, "Further development and application of high-order spectral volume methods for compressible flows," *Journal of Aerospace Technology and Management*, vol. 9, pp. 301 – 327, 09 2017.
33. J. L. Sims, "Tables for supersonic flow around right circular cones at zero angle of attack," *Tech. Rep. SP-3004, NASA*, 1964.