Relatório - Trabalho final

Vinícius Pessoa Mapelli

1 Introduction

Shock Tube problem proposed by [3] is a classical problem that involves a system of hyperbolic equations, such as 1D Euler equations. This is an interesting problem to test numerical schemes, once it involves phenomena such as time evolution of a shock wave, a contact discontinuity (interface between two fluid regions of distinct densities, but equal pressure) and a expansion fan. Also, existence of analytical solution to Sod Shock Tube problem makes it a great tool to compare and evaluate numerical solutions. In this context, shock tube problem is a representative of the numerical difficulties seen in solution of the Euler equations of gas dynamics [2].

Among several strategies to solve this system of partial differential equation (PDE), this work focus on using Finite Element Method (FEM). In short, this method consists in rewriting the original PDE by multiplying it by a weight function and integrate over the space domain, resulting in the weak form of the governing equation. Afterwards, solution is approximated to be a linear combination of shape functions, which are similar to weight functions (Galerkin approach). The number of functions depends on how many elements results from space domain discretization. In that way, this numerical method resumes in solving a linear system of equations, whose solution is a vector composed by the values of the coefficients related to each of the approximation function assumed.

2 Problem formulation

Sod Shock Tube problem is governed by the Euler equations of gas dynamics, which express conservation of mass, momentum and energy in a compressible, inviscid and non-conducting fluid. Euler equations can be expressed as:

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) &= 0 \\ \frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v} + p\mathbf{I}) &= \rho \mathbf{b} \\ \frac{\partial \rho E}{\partial t} + \nabla \cdot ((\rho E + p)\mathbf{v}) &= \mathbf{v} \cdot \rho \mathbf{b} \end{aligned} \tag{1}$$

In Shock Tube problem, where body forces $\mathbf{b}$ are not taken in account and $\mathbf{v}$ has non-zero component value only in x direction (v), we can rewrite the system described in Eq. (1) in the following vector form:

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0, \tag{2}$$

where

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho v \\ \rho E \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ v(\rho E + p) \end{pmatrix} \tag{3}$$

Pressure can be calculated under the assumption of a perfect gas:

$$p = (\gamma - 1)\rho \left(E - \frac{v^2}{2} \right), \tag{4}$$

where the heat capacity ratio γ is assumed 1.4 in this work, which is a reasonable estimate for air properties.

In theses conditions, components of inviscid flux vector $\mathbf{F}$ are homogeneous functions of degree 1, and it can be shown that it obeys the relation:

$$\mathbf{F}(\mathbf{U}) = \mathbf{A}(\mathbf{U})\mathbf{U}, \tag{5}$$

where $\mathbf{A}(\mathbf{U}) = \partial \mathbf{F} / \partial \mathbf{U}$ is given by:

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 \\ -\frac{1}{2}(3-\gamma)v^2 & (3-\gamma)v & \gamma - 1 \\ (\gamma-1)v^3 - \gamma v E & \gamma E - \frac{3}{2}(\gamma-1)v^2 & \gamma v \end{pmatrix} \tag{6}$$

Given this, Sod Shock Tube problem consists in solving numerical system presented in Eq. (2) on the spatial domain $[0, 1]$, in which initial data is:

$0 \leq x \leq 1/2$	$1/2 < x \leq 1$
$\rho = 1.0$	$\rho = 0.125$
$\rho v = 0.0$	$\rho v = 0.0$
$\rho E = 2.5$	$\rho E = 0.25$

The initial regions which posses distinct densities and pressure values are maintained by a diaphragm, which is ruptured at $t = 0$. Thus, the objective of this problem is to compute the evolution in time of the conservation variables along the tube [2].

It is important mentioning that reflection of shock or expansion waves at the boundaries of the spatial domain will not be taken in account in this work, so numerical scheme can be used only up to time instant when these waves are about to reach one of the ends of tube. In that way, spatial boundaries are considered to have fixed values, given by initial data.

3 Numerical methodology

3.1 One-step Taylor-Galerkin method

System of equations to be numerically solved is presented in its strong in Eq. (2). To solve this via Finite Element Method, we have to find its weak formulation, and here we are going to present how this is done using the so called One-Step Taylor-Galerkin method.

The first step is to use a weight function $\mathbf{W}(x)$ and integrate over the whole spatial domain $[0, L]$ (in shock tube problem, we have $L = 1$):

$$\int_0^L \mathbf{W} \cdot [\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x] \, dx = 0 \tag{7}$$

Time integration can be performed by applying a Taylor expansion of $\mathbf{U}(x, t + \Delta t)$ around $\mathbf{U}(x, t)$ up to second order in such a way that:

$$\frac{\mathbf{U}(x,t+\Delta t)-\mathbf{U}(x,t)}{\Delta t}=\mathbf{U}_t+\frac{\Delta t}{2}\mathbf{U}_{tt}, \tag{8}$$

Now, substituting Eq. (8) to Eq. (7), one can obtain:

$$\int_0^L \mathbf{W}\cdot\frac{\mathbf{U}^{n+1}-\mathbf{U}^n}{\Delta t}\,dx=-\int_0^L \mathbf{W}\cdot\mathbf{F}_x^n\,dx + \frac{\Delta t}{2}\int_0^L \mathbf{W}\cdot\mathbf{U}_{tt}^n\,dx \tag{9}$$

By using original system equation ($\mathbf{U}_t=-\mathbf{F}_x$), it is possible to determine:

$$\mathbf{U}_{tt}=-\mathbf{F}_{xt}=-\mathbf{F}_{tx}=-(\mathbf{A}\mathbf{U}_t)_x=(\mathbf{A}\mathbf{F}_x)_x=(\mathbf{A}^2\mathbf{U}_x)_x \tag{10}$$

Substuting Eq. (10) in Eq. (9), it yields:

$$\int_0^L \mathbf{W}\cdot\frac{\mathbf{U}^{n+1}-\mathbf{U}^n}{\Delta t}\,dx=-\int_0^L \mathbf{W}\cdot\mathbf{F}_x^n\,dx + \frac{\Delta t}{2}\int_0^L \mathbf{W}\cdot((\mathbf{A}^n)^2\mathbf{U}_x)_x\,dx \tag{11}$$

Applying integration by parts in both terms of the left-hand side of Eq. (11), one can obtain:

$$\begin{aligned}\int_0^L \mathbf{W}\cdot\mathbf{F}_x^n\,dx&=-\int_0^L \mathbf{W}_x\cdot\mathbf{F}^n\,dx+[\mathbf{W}\cdot\mathbf{F}^n]\,\big|_0^L\\ \int_0^L \mathbf{W}\cdot((\mathbf{A}^n)^2\mathbf{U}^n)_x\,dx&=-\int_0^L \mathbf{W}_x\cdot\mathbf{A}^n\mathbf{F}^n\,dx\\ &\quad+\left[\mathbf{W}\cdot(\mathbf{A}^n)^2\mathbf{U}^n)\right]\,\big|_0^L\end{aligned} \tag{12}$$

Again, from relations in Eq. (10), we know that $\mathbf{A}^2\mathbf{U}=\mathbf{A}\mathbf{F}=\mathbf{A}(-\mathbf{U}_t)=-\mathbf{F}_t$. Thus, replacing relations from Eq. (12) in Eq. (11), one can obtain One-step Taylor Galerkin method [2]:

$$\begin{aligned}\int_0^L \mathbf{W}\cdot\frac{\mathbf{U}^{n+1}-\mathbf{U}^n}{\Delta t}\,dx&=\int_0^L \mathbf{W}_x\cdot\mathbf{F}^n\,dx\\ -\frac{\Delta t}{2}\int_0^L \mathbf{W}_x\cdot\mathbf{A}^n\mathbf{F}^n\,dx&-[\mathbf{W}\cdot(\mathbf{F}^n+(\Delta t/2)\mathbf{F}_t^n)]\,\big|_0^L\end{aligned} \tag{13}$$

3.2 Linear elements

In order to solve Sod problem using linear elements, one may assume that shape and test functions are the equal to each other (Galerkin approach), and thus, we can write weight function and approximated solution, respectively, as:

$$\mathbf{W}_j=\sum_{j=1}^{j=N}\mathbf{\Phi}_j(x)\boldsymbol{\beta}_j,\quad \mathbf{U}_i=\sum_{i=1}^{i=N}\mathbf{\Phi}_i(x)\mathbf{U}(\mathbf{x_i}) \tag{14}$$

where N is the number of nodes in discretized domain $[0,L]$, and

$$\mathbf{\Phi}_i(x)=\begin{pmatrix}\phi_i & 0 & 0\\ 0 & \phi_i & 0\\ 0 & 0 & \phi_i\end{pmatrix}, \tag{15}$$

where

$$\phi_i=\begin{cases}0, & x<x_{i-1}\\ \frac{x-x_{i-1}}{x_i-x_{i-1}}, & x_{i-1}\leqslant x\leqslant x_{i+1}\\ 0, & x>x_{i+1},\end{cases} \tag{16}$$

Also, in this work, inviscid flux $\mathbf{F}$ is interpolated using its test functions as described in Eq. (17). This approach is known as group representation:

$$\mathbf{F}=\sum_{i=1}^{i=N}\mathbf{\Phi}_i(x)\mathbf{F}(\mathbf{U}_i) \tag{17}$$

By substituting Eqs. (14) and (17) in weak problem formulation in Eq. (13), since we are interested in a solution valid to any shape function $\mathbf{W}$, it can be shown this procedure yields the following linear system:

$$\mathbf{M}\boldsymbol{\Delta}\mathbf{U}=\Delta t\mathbf{C}\mathbf{F}^n-\frac{\Delta t^2}{2}\mathbf{K}(\mathbf{A}^2)^n\mathbf{U}^\mathbf{n}-\Delta t\mathbf{b}^\mathbf{n} \tag{18}$$

where $\boldsymbol{\Delta}\mathbf{U}=\mathbf{U}^{n+1}-\mathbf{U}^n$, and matrices $\mathbf{M},\mathbf{C}$ and $\mathbf{K}$ (or consistent mass, convective and diffusive matrices) are defined by:

$$\mathbf{M}=\begin{pmatrix}\int_0^L\mathbf{\Phi}_1\mathbf{\Phi}_1dx & \int_0^L\mathbf{\Phi}_1\mathbf{\Phi}_2dx & \dots & \int_0^L\mathbf{\Phi}_1\mathbf{\Phi}_Ndx\\ \int_0^L\mathbf{\Phi}_2\mathbf{\Phi}_1dx & \int_0^L\mathbf{\Phi}_2\mathbf{\Phi}_2dx & \dots & \int_0^L\mathbf{\Phi}_2\mathbf{\Phi}_Ndx\\ \vdots & \vdots & \ddots & \vdots\\ \int_0^L\mathbf{\Phi}_N\mathbf{\Phi}_1dx & \int_0^L\mathbf{\Phi}_N\mathbf{\Phi}_2dx & \dots & \int_0^L\mathbf{\Phi}_N\mathbf{\Phi}_Ndx\end{pmatrix} \tag{19}$$

$$\mathbf{C}=\begin{pmatrix}\int_0^L\mathbf{\Phi}_1'\mathbf{\Phi}_1dx & \int_0^L\mathbf{\Phi}_1'\mathbf{\Phi}_2dx & \dots & \int_0^L\mathbf{\Phi}_1'\mathbf{\Phi}_Ndx\\ \int_0^L\mathbf{\Phi}_2'\mathbf{\Phi}_1dx & \int_0^L\mathbf{\Phi}_2'\mathbf{\Phi}_2dx & \dots & \int_0^L\mathbf{\Phi}_2'\mathbf{\Phi}_Ndx\\ \vdots & \vdots & \ddots & \vdots\\ \int_0^L\mathbf{\Phi}_N'\mathbf{\Phi}_1dx & \int_0^L\mathbf{\Phi}_N'\mathbf{\Phi}_2dx & \dots & \int_0^L\mathbf{\Phi}_N'\mathbf{\Phi}_Ndx\end{pmatrix} \tag{20}$$

$$\mathbf{K}=\begin{pmatrix}\int_0^L\mathbf{\Phi}_1'\mathbf{\Phi}_1'dx & \int_0^L\mathbf{\Phi}_1'\mathbf{\Phi}_2'dx & \dots & \int_0^L\mathbf{\Phi}_1'\mathbf{\Phi}_N'dx\\ \int_0^L\mathbf{\Phi}_2'\mathbf{\Phi}_1'dx & \int_0^L\mathbf{\Phi}_2'\mathbf{\Phi}_2'dx & \dots & \int_0^L\mathbf{\Phi}_2'\mathbf{\Phi}_N'dx\\ \vdots & \vdots & \ddots & \vdots\\ \int_0^L\mathbf{\Phi}_N'\mathbf{\Phi}_1'dx & \int_0^L\mathbf{\Phi}_N'\mathbf{\Phi}_2'dx & \dots & \int_0^L\mathbf{\Phi}_N'\mathbf{\Phi}_N'dx\end{pmatrix} \tag{21}$$

Matrices shown in Eqs. (19), (20) and (21) are in global coordinates, and due to characteristics of $\mathbf{\Phi}$, it has zero values in many positions. To show how to compute these global matrices, it is easier to explain taking element matrices as starting point. But one should keep in mind that the origin from this approach comes from these matrices shown above.

3.2.1 Linear element matrices

For a linear element, in local coordinates x_e, since there are only two nodes, only two shape functions are needed:

$$\mathbf{\Phi}_1(x)=\begin{pmatrix}\phi_1 & 0 & 0\\ 0 & \phi_1 & 0\\ 0 & 0 & \phi_1\end{pmatrix},\quad \mathbf{\Phi}_2(x)=\begin{pmatrix}\phi_2 & 0 & 0\\ 0 & \phi_2 & 0\\ 0 & 0 & \phi_2\end{pmatrix}, \tag{22}$$

where

$$\phi_1(x)=\frac{x_e}{L_e},\quad \phi_2(x)=\frac{1-x_e}{L_e} \tag{23}$$

in which the element length L_e, in this case, is equal to the grid spacing ($L_e=h=L/(N-1)$).

By performing integrations over the element domain, given definitions of $\mathbf{\Phi}$, one can obtain the element matrices:

$$\begin{aligned}\mathbf{M}^e &= \begin{pmatrix} \int_0^{L_e} \mathbf{\Phi}_1\mathbf{\Phi}_1 dx & \int_0^{L_e} \mathbf{\Phi}_1\mathbf{\Phi}_2 dx \\ \int_0^{L_e} \mathbf{\Phi}_2\mathbf{\Phi}_1 dx & \int_0^{L_e} \mathbf{\Phi}_2\mathbf{\Phi}_2 dx \end{pmatrix} \\ &= \frac{L_e}{6}\begin{pmatrix} 2\mathbf{I} & \mathbf{I} \\ \mathbf{I} & 2\mathbf{I} \end{pmatrix}\end{aligned} \tag{24}$$

where $\mathbf{I}$ is a 3x3 identity matrix.

In the same way, convective matrix in linear element is given by:

$$\begin{aligned}\mathbf{C}^e &= \begin{pmatrix} \int_0^{L_e} \mathbf{\Phi}_1^{'}\mathbf{\Phi}_1 dx & \int_0^{L_e} \mathbf{\Phi}_1^{'}\mathbf{\Phi}_2 dx \\ \int_0^{L_e} \mathbf{\Phi}_2^{'}\mathbf{\Phi}_1 dx & \int_0^{L_e} \mathbf{\Phi}_2^{'}\mathbf{\Phi}_2 dx \end{pmatrix} \\ &= \frac{1}{2}\begin{pmatrix} -\mathbf{I} & -\mathbf{I} \\ \mathbf{I} & \mathbf{I} \end{pmatrix}.\end{aligned} \tag{25}$$

And finally, diffusive element matrix can be calculated as:

$$\begin{aligned}\mathbf{K}^e &= \begin{pmatrix} \int_0^{L_e} \mathbf{\Phi}_1^{'}\mathbf{\Phi}_1^{'} dx & \int_0^{L_e} \mathbf{\Phi}_1^{'}\mathbf{\Phi}_2^{'} dx \\ \int_0^{L_e} \mathbf{\Phi}_2^{'}\mathbf{\Phi}_1^{'} dx & \int_0^{L_e} \mathbf{\Phi}_2^{'}\mathbf{\Phi}_2^{'} dx \end{pmatrix} \\ &= \frac{1}{L_e}\begin{pmatrix} \mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & \mathbf{I} \end{pmatrix}.\end{aligned} \tag{26}$$

3.2.2 Building global matrices

Matrices shown in previous section corresponds to each of elements. Considering linear shape functions, in a spatial discretization with N nodes, there are $N-1$ elements, which means we have $N-1$ sets of each of these matrices corresponding to each element: $[\mathbf{M^1}, \mathbf{M^2}, \ldots, \mathbf{M^{N-1}}]$, $[\mathbf{C^1}, \ldots, \mathbf{C^{N-1}}]$, $[\mathbf{K^1}, \ldots, \mathbf{K^{N-1}}]$. Here, in this case, it is considered all elements are linear (same node size) with same length, thus all matrices $\mathbf{M}^e$ (or $\mathbf{C}^e$ or $\mathbf{K}^e$) have same values among themselves. Element index is useful to show how to build global matrices:

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}_{11}^1 & \mathbf{M}_{12}^1 & & & \\ \mathbf{M}_{21}^1 & \mathbf{M}_{22}^1 + \mathbf{M}_{11}^2 & \mathbf{M}_{12}^2 & & \\ & \mathbf{M}_{21}^2 & \mathbf{M}_{22}^2 + \mathbf{M}_{11}^3 & & \\ & & & \ddots & \\ & & & \mathbf{M}_{22}^{N-2} + \mathbf{M}_{11}^{N-1} & \mathbf{M}_{12}^{N-1} \\ & & & \mathbf{M}_{12}^{N-1} & \mathbf{M}_{22}^{N-1} \end{pmatrix} \tag{27}$$

in which $\mathbf{M}_{ij}^e$ corresponds to the 3x3 matrix in position (i, j) of the element matrix presented in Eq. (24), related to the e-th element. Same procedure applies to build global matrices $\mathbf{K}$ and $\mathbf{C}$. In fact, this is same procedure used to build global matrices when nodes have only 1 degree of freedom (DOF) instead of 3, as in the of Sod Shock tube. The main difference is that single values becomes blocks of 3×3 matrices, and element matrices which were expected to have 2×2 dimension in linear case, now posses 6×6 values. Thus, global matrices final shapes are $3N \times 3N$, and one can see that it has 0 values in many places, i.e, a high sparsity characteristic.

3.3 Quadratic elements

Extending solution to quadratic elements follows same initial procedures highlighted in previous section to linear elements. To avoid repetition, we are just going to pinpoint the main differences in element matrices and in the building procedure.

The first difference to note is that quadratic elements involves three nodes $\mathbf{U}_1^e$, $\mathbf{U}_2^e$ e $\mathbf{U}_3^e$, in order to be possible to have quadratic shape functions:

$$\mathbf{\Phi}_1(x) = \phi_1(x)\mathbf{I},\ \mathbf{\Phi}_2(x) = \phi_1(x)\mathbf{I},\ \mathbf{\Phi}_3(x) = \phi_3(x)\mathbf{I}, \tag{28}$$

where

$$\begin{aligned}\phi_1 &= \frac{2}{L_e}x_e^2 - \frac{x_e}{L_e}, \quad \text{for} -L_e/2 \leqslant x \leqslant L_e/2 \\ \phi_2 &= \frac{-4}{L_e}x_e^2 + 1, \quad \text{for} -L_e/2 \leqslant x \leqslant L_e/2 \\ \phi_3 &= \frac{2}{L_e}x_e^2 + \frac{x_e}{L_e}, \quad \text{for} -L_e/2 \leqslant x \leqslant L_e/2\end{aligned} \tag{29}$$

in this case, element length L_e is equal to twice the spatial grid spacing ($L_e = 2h$). Following same procedure shown in previous section, it is possible to find quadratic element matrices:

$$\mathbf{M}^e = \frac{L_e}{30}\begin{pmatrix} 4\mathbf{I} & 2\mathbf{I} & -1\mathbf{I} \\ 2\mathbf{I} & 16\mathbf{I} & 2\mathbf{I} \\ -1\mathbf{I} & 2\mathbf{I} & 4\mathbf{I} \end{pmatrix} \tag{30}$$

$$\mathbf{C}^e = \frac{1}{6}\begin{pmatrix} -3\mathbf{I} & -4\mathbf{I} & 1\mathbf{I} \\ 4\mathbf{I} & 0 & -4\mathbf{I} \\ -1\mathbf{I} & 4\mathbf{I} & 3\mathbf{I} \end{pmatrix} \tag{31}$$

$$\mathbf{K}^e = \frac{1}{3L_e}\begin{pmatrix} 7\mathbf{I} & -8\mathbf{I} & 1\mathbf{I} \\ -8\mathbf{I} & 16\mathbf{I} & -8\mathbf{I} \\ 1\mathbf{I} & -8\mathbf{I} & 7\mathbf{I} \end{pmatrix} \tag{32}$$

3.3.1 Building global matrices

Another major difference in using quadratic elements is in the process of building global matrix. Since quadratic elements involves 3 nodes, number of elements for a spatial discretization with N nodes is $(N-1)/2$. This fact imposes a condition in discretization to use quadratic elements: the number of nodes N has to be an odd number equal or greater than 3. That being said, set of element matrices are $[\mathbf{M^1}, \mathbf{M^2}, \ldots, \mathbf{M^{(N-1)/2}}]$, $[\mathbf{C^1}, \ldots, \mathbf{C^{(N-1)/2}}]$, $[\mathbf{K^1}, \ldots, \mathbf{K^{(N-1)/2}}]$, and building process of $\mathbf{M}$ can be described as:

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}_{11}^1 & \mathbf{M}_{12}^1 & \mathbf{M}_{13}^1 & & & \\ \mathbf{M}_{21}^1 & \mathbf{M}_{22}^1 & \mathbf{M}_{23}^1 & & & \\ \mathbf{M}_{31}^1 & \mathbf{M}_{32}^1 & \mathbf{M}_{33}^1 + \mathbf{M}_{11}^2 & \mathbf{M}_{12}^2 & \mathbf{M}_{13}^2 & \\ & & \mathbf{M}_{21}^2 & \mathbf{M}_{22}^2 & \mathbf{M}_{23}^2 & \\ & & \mathbf{M}_{31}^2 & \mathbf{M}_{22}^2 & \mathbf{M}_{33}^2 + \mathbf{M}_{11}^3 & \\ & & & & & \ddots \end{pmatrix} \tag{33}$$

Building process of global $\mathbf{C}$ and $\mathbf{K}$ matrices using quadratic elements are analogous to the procedure as in Eq. (33).

3.4 Matrices A and F

Although matrices $\mathbf{A}$ and $\mathbf{F}$ and their relation have already been defined in Eqs. (6), (3) and (5), their appearance in linear system to be solved in Eq. (18) may cause some confusion at a first glance. In this equation, terms $\mathbf{F}^n = \mathbf{A}^n\mathbf{U}^n$ and $(\mathbf{A}^2)^n\mathbf{U}^n$ are expected to have dimension of $3Nx1$. Dimensions of matrix $(\mathbf{A}^2)^n$, as defined in Eq. (6), has dimension of 3x3, and actually, the term $(\mathbf{A}^2)^n\mathbf{U^n}$ means that we are applying $\mathbf{A}$ to each value of vector function $\mathbf{U_i}$ evaluated at nodes coordinates $x = x_i$. In that way, we can also work with concept of a global matrix $\mathbf{A}$ such that:

$$\mathbf{A}^n\mathbf{U}^n = \begin{pmatrix} \mathbf{A_1}^n & & & \\ & \mathbf{A_2}^n & & \\ & & \ddots & \\ & & & \mathbf{A_N}^n \end{pmatrix} \begin{pmatrix} \mathbf{U_1}^n \\ \mathbf{U_2}^n \\ \vdots \\ \mathbf{U_N}^n \end{pmatrix} \tag{34}$$

where $\mathbf{A}_i^n$ is calculated as described in Eq. (6), evaluated by using the vector function $\mathbf{U}_i^n$.

3.5 Boundary condition

Finally, we focus our attention on the boundary vector $\mathbf{b}^n = [\mathbf{W} \cdot (\mathbf{F}^n + (\Delta t/2)\mathbf{F}_t^n)] \big|_0^L$ originated from last term in left-hand side of one-step Taylor Galerkin (Eq. (13)).

To start off, since we are not dealing with reflections at boundaries, time variation of flux at these nodes are equal to zero ($\mathbf{F}_t = 0$). That leaves us to handle only the term $[\mathbf{W} \cdot \mathbf{F}^n]|_0^L$. Analyzing characteristics of shape functions, one can observe only ϕ_1 and ϕ_N have non-zero values at, respectively, $x = 0$ and $x = L$. Actually, exactly at the boundaries, these shape functions are $\phi = 1$, and boundary term resumes to evaluations of flux $\mathbf{F}$ as follows:

$$\mathbf{b}^n = \begin{pmatrix} -\mathbf{F_1}^n \\ 0 \\ \vdots \\ 0 \\ \mathbf{F_N}^n \end{pmatrix} \tag{35}$$

where prescribed flux values at the boundary nodes can be computed from initial condition:

$$\mathbf{F_1} = \begin{pmatrix} 0 \\ 1.0 \\ 0 \end{pmatrix}, \quad \mathbf{F_N} = \begin{pmatrix} 0 \\ 0.1 \\ 0 \end{pmatrix} \tag{36}$$

3.6 Computational implementation

The Sod shock tube problem was solved following methodology described previously, using a mesh of 101 nodes, which corresponds to 100 elements when linear shape functions are used, and 50 elements for quadratic elements. Time integration has been performed until $t = 0.2$ using a fixed time step $\Delta t = 1.5 \times 10^3$, which corresponds, in our code, to a Courant number:

$$C = (v_{max} + c)\frac{\Delta t}{h} \approx 0.33$$

Numerical code was implemented computationally in Python™ language, using software Spyder™ as Development Environment. Global matrices $\mathbf{M}$, $\mathbf{K}$ e $\mathbf{C}$ were built as described in Sections (3.2.2) and (3.3.1) using sparse storage implemented in SciPy sparse package, as well as global matrix $\mathbf{A}$ (as discussed in Section (3.4)). In this way, memory needed to matrices storage is reduced and code performance regarding computation speed is increased.

Since global matrices are independent of time and grid spacing is fixed, one can calculate these matrices just once in program, and afterwards, perform time integration in a main loop as described in Eq. (18). This is the basic idea behind the main code. All the rest follows the procedures described before, by using object-oriented abstraction with nodes, elements, boundary conditions and problem classes.

4 Numerical Results

Numerical results obtained for density (ρ), velocity(v), pressure(p) and total energy density (ρE) are presented in Fig. 1, using linear elements, and in Fig. 2, for quadratic elements. Numerical results are represented as crosses, while solid lines represent analytic results, generated through a Sod Shock Tube Problem Solver available in MathWorks forum [1]. It is worth mentioning that compatibility function to find pressure at contact discontinuity has to be replaced as it is suggested in comments.

In Fig. 3, one can see density evolution in time (y axis) inside spatial domain (x axis). Density values are represented by colors scaled as represented by colormap in the right-hand side of the figure. This figure makes it possible to visualize five regions stabilized in this flow: from left-hand to right-hand side of domain, one can observe a first region where density is still unaffected, and its value is still $\rho_L = 1$. After that, a second region where density value decays due to expansion waves. Then, we have two regions where density value stay almost constant, and there is a point between these regions where there is a density discontinuity, which is called contact discontinuity (pressure and velocity are continuous). Last region (in the far right-hand side of tube) is also unaffected and remains with $\rho_R = 0.1$. A discontinuity is also observed with previous region, but this time, this is caused by a shock wave, which also causes discontinuities in velocity and pressure.

5 Final Remarks

From results presented in this work, one can conclude that methodology presented to produce a numerical solution was able to reproduce analytic solution with good agreement. Discontinuities locations were well predicted by numerical model, as well as amplitudes of flow variables studied. Numerical difficulties were already expected and are visible near discontinuities regions, where oscillation of flow quantities are observed.

Referências

[1] . Sod shock tube problem solver. https://www.mathworks.com/matlabcentral/fileexchange/46311-sod-shock-tube-problem-solver, 2014. Accessed: 2019-06-05.

[2] J Donea and A Huerta. *Finite Element Methods for Flow Problems*. John Wiley and Sons, 2003.

[3] Gary A Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of computational physics*, 27(1):1–31, 1978.

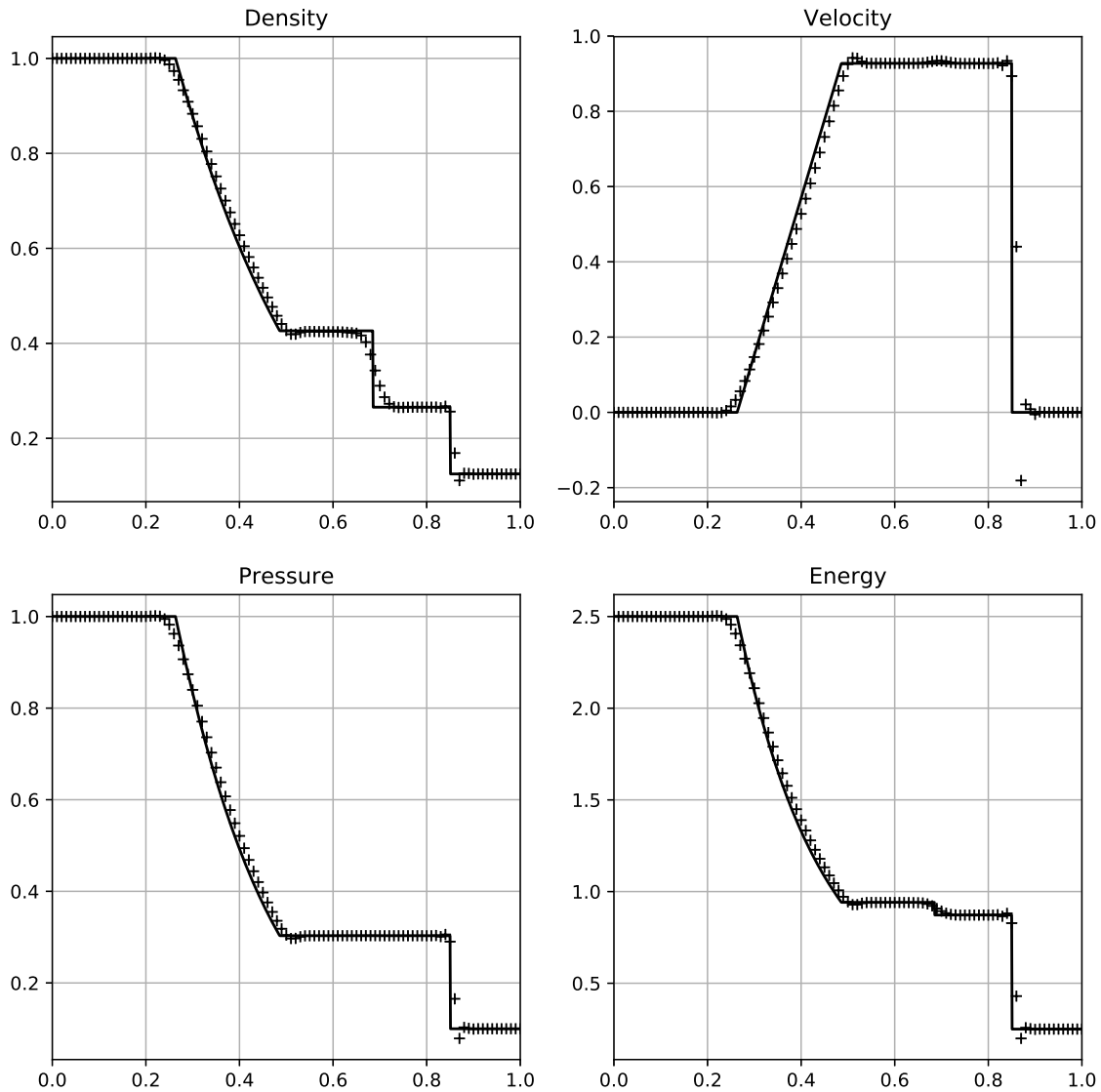

Figura 1: Numerical results obtained using linear elements compared to analytic solution (solid lines).

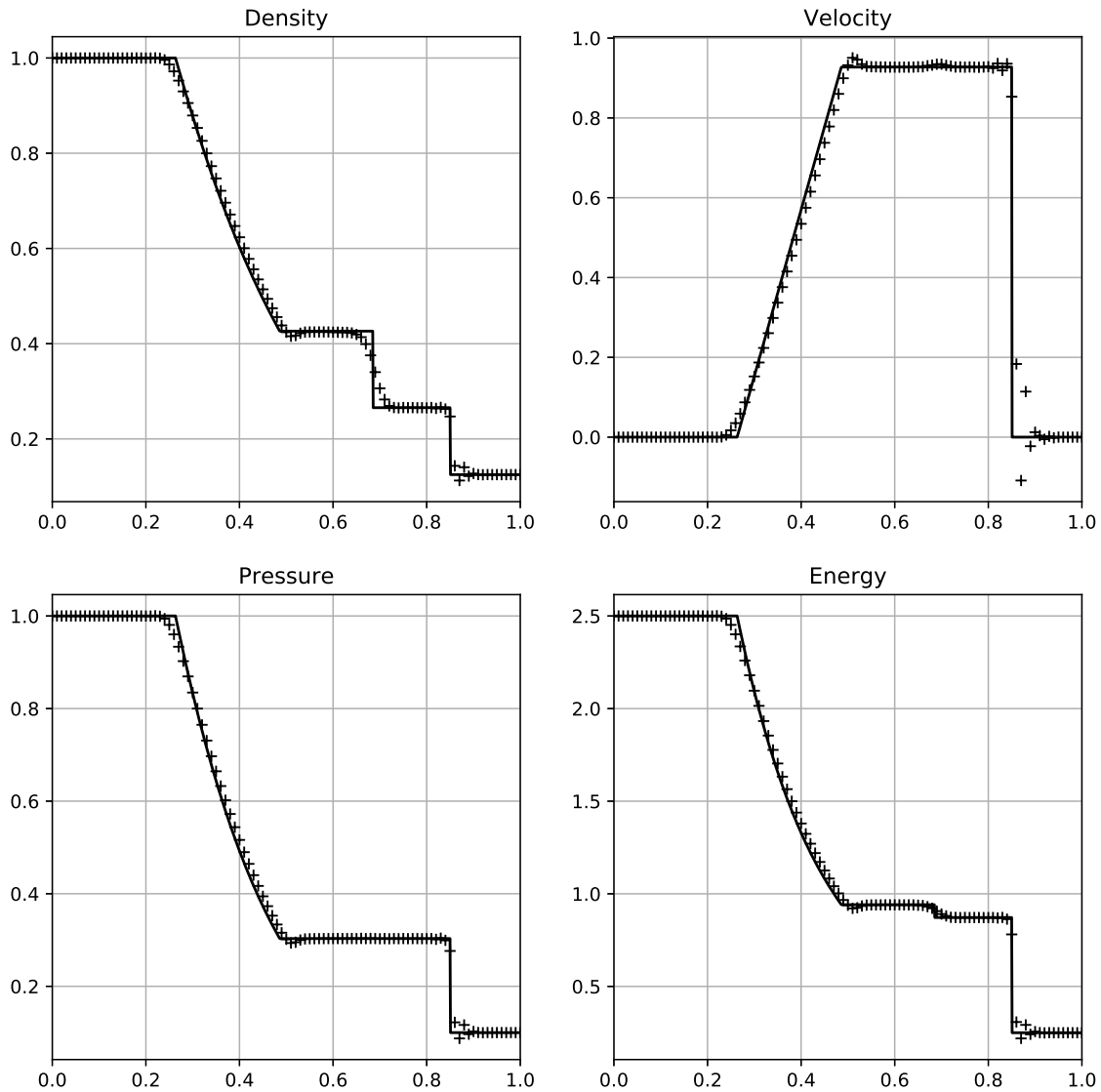

Figura 2: Numerical results obtained using quadratic elements compared to analytic solution (solid lines).

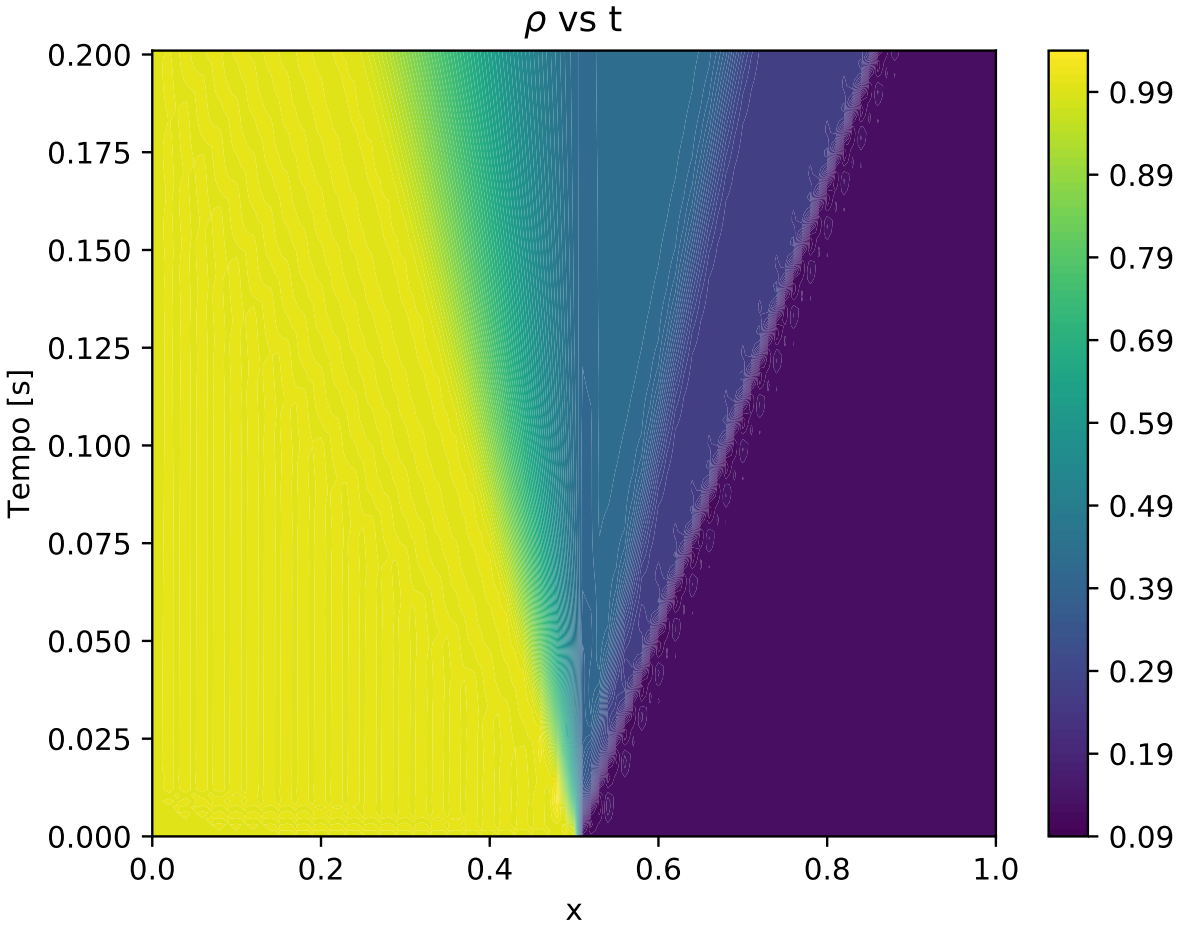

Figura 3: Evolution in time of density inside shock tube between $t = 0$ and $t = 0.2[s]$.