Plasma Fluid Models

Roshan Kumar Subramanian

May 8, 2022

Abstract

This mini project is to develop the MATLAB code for the Euler Equation, and the Ideal MHD equations. The equations are implemented in the conservative form. And, then the equations are implemented using finite volume method of MUSCL scheme. MUSCL scheme is a Total variation diminishing (TVD) scheme. Roe flux is used for the Euler equation model, and the Roe, and the HLLE fluxes are implemented for the Ideal MHD equation. Also, the open source production code called 'Gkeyll' is also used to compare the solution of Euler equation, with the Vlasov-Maxwell for the SOD shock problem. Based on the simulations, we can see the rarefaction, and the shock propagation across the contact discontinuities. The plots on density, bulk velocity, pressure, and energy are plotted for these different set of equations. And the difference in solution can be seen, and discussed in this report.

Keywords— SOD Shock, Euler Equation, Ideal MHD equation, Gkeyll simulation code, Finite volume method, Conservative form, Minmod flux limiter

1 Introduction

The project is to model the SOD shock using three different models namely using Euler equation, Ideal MHD equation, and using the Gkeyll simulation software with Vlasov-Maxwell Equation. The project is implemented from scratch in MATLAB. The equations are implemented in the conservation form, calculating the flux using the finite volume method. The MUSCL scheme is used to calculate the flux. The MUSCL scheme is the TVD scheme, and the TVD scheme for a one-dimensional (1D) function is defined as follows,

$$TVD = \sum_{i=1}^{N} |\phi_{i+1} - \phi_i| \tag{1}$$

where 'N' is the number of grids. Since, MUSCL scheme is TVD, it is a monotonicity preserving, that is, there will be no new maxima, and minima in the solution. This can be visualized in all the plots described in this report. So, that the solutions are not tending to develop unrealistic overshoots, and undershoots at steep gradients withholding the original information. The basic mathematical description of plasma requires to solve for the distribution function, f with 6 spatial dimensions, and the time domain. These equations like Euler, Ideal MHD, and Vlasov Equation simplifies the complexity by taking the velocity moments. This significantly reduces the computational cost, and also helps to simulate the complex phenomenons in the plasma dynamics. This report shows the project simulation the rarefaction, and shock waves in the SOD shock tube model using these equations. The initial conditions of the SOD-Shock problem are given as follows,

Region1:

$$\begin{aligned} \rho &= 1.0 \\ p &= 1.0 \\ u &= 0.0 \end{aligned} \tag{2}$$

Region2:

$$\begin{aligned} \rho &= 0.125 \\ p &= 0.1 \\ u &= 0.0 \end{aligned} \tag{3}$$

where ρ is the density, p is the pressure, and u is the bulk or total velocity.

2 Analytical Model

2.1 Euler Equation

The Euler equations are implemented in conservative form as shown below,

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} = 0 \tag{4}$$

where Q is the conserved variable, and F is the flux. The system of equations are represented in conserved form with the matrices as below, The conserved variables, Q are defined as below,

$$\begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix} \tag{5}$$

The Flux, F is defined as below,

$$\begin{bmatrix} \rho u \\ \rho u^2 + P \\ u(E + P) \end{bmatrix} \tag{6}$$

where ρ is the density, p is the pressure, and u is the bulk or total velocity, E is the energy, and P is the pressure.

2.2 Gkeyll software using Vlasov-Maxwell Equation

The Gkeyll is an open source software to model plasma dynamics using gyrokinetic equations, Vlasov-Maxwell equations, and multi-fluid equations. This software is being developed by the plasma researchers at Massachusetts Institute of Technology (MIT), and Virginia Tech (VT). The Gkeyll simulation is done with Vlasov-Maxwell equation for SOD-shock problem to compare it with the Euler equations. The Vlasov-Maxwell equation can be described with the following equations,

The Vlasov Equation with no collision can be described as,

$$\frac{\partial f}{\partial t} + \vec{v}_i \frac{\partial f}{\partial \vec{x}} + \frac{q}{m}(\vec{E} + \vec{v} * \vec{B})\frac{\partial f}{\partial \vec{v}} = 0 \tag{7}$$

The Maxwell Equations are given by,

$$\begin{aligned} \frac{1}{c^2}\frac{\partial \vec{E}}{\partial t} + \mu_0 \vec{j} &= \nabla * \vec{B} \\ \frac{\partial \vec{B}}{\partial t} + \nabla * \vec{E} &= 0 \end{aligned} \tag{8}$$

Additional constraints that are required are as below,

$$\begin{aligned} \nabla.\vec{E} &= \frac{\rho c}{\epsilon_0} \\ \nabla.\vec{B} &= 0 \end{aligned} \tag{9}$$

The equations are implemented with the serendipity basis function making it fewer interior nodes, and making it feasible to run complex, and high computational cost problems. The Knudsen number is also used to control the boundary conditions. The Knudsen number is the non-dimensional number which helps to characterize the boundary conditions of the fluid flow.

$$Kn = \frac{\lambda}{L_{char}} = \frac{mean\ free\ path}{characteristic\ length} \tag{10}$$

2.3 Ideal Magnetohydrodynamics Equation

The Ideal magnetohydrodynamics equation (Ideal-MHD) is the subset of the complex Magnetohydrodynamic equation. There are different variants like two-fluid MHD, resistive-MHD, and Hall-MHD holding the hall terms. In this project, the simplified Ideal MHD equations are used to simulate the SOD shock model. The Ideal-MHD is good enough to simulate even some rigorous plasma model, so the SOD shock model can be simulated appropriately. The Ideal-MHD equations are purely hyperbolic. The equation is shown below in the conservative form.

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} = 0 \tag{11}$$

The conserved variables, Q are defined as below,

$$\begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ B_x \\ B_y \\ B_z \\ E \end{bmatrix} \tag{12}$$

The Flux, F is defined as below,

$$\begin{bmatrix} \rho u \\ \rho u^2 + \frac{B_x^2}{\mu_0} + P + \frac{B^2}{2\mu_0} \\ \rho uv - \frac{B_x B_y}{\mu_0} \\ \rho uw - \frac{B_x B_z}{\mu_0} \\ 0 \\ uB_y - B_x v \\ uB_z - B_x w \\ (E + P + \frac{B^2}{2\mu_0})u - \frac{\vec{B}\vec{u_t}}{\mu 0} B_x \end{bmatrix} \tag{13}$$

where u, v, w are the velocities in x, y, and z directions. And, ρ is the density, and B_x, B_y, and B_z are the magnetic fields in x, y, and z directions respectively. The E, and P are the energy, and pressure respectively.

The total velocity u_t^2 is defined as $u^2 + v^2 + w^2$, and the total magnetic field, B^2 is defined as $B_x^2 + B_y^2 + B_z^2$. The energy is defined as the following equation,

$$E = \frac{P}{\gamma - 1} + \frac{1}{2}\rho u_t^2 + \frac{B^2}{2\mu_0} \tag{14}$$

And, the pressure, P is given by,

$$P = \frac{2}{3}\left(E - \frac{1}{2}\rho u_t^2 - \frac{B^2}{2\mu_0}\right) \tag{15}$$

3 Numerical Method

3.1 MUSCL scheme

The MUSCL scheme is one of the Finite volume methods that can take the Godunov scheme, and reconstruct the states. This is one of the monotonicity preserving schemes.

The left, and right conserved variables are calculated using the following formulation as mentioned in figure-1,

$$
\begin{aligned}
Q^L_{i+\frac{1}{2}} &= Q_i + \frac{\epsilon}{4}[(1-\kappa)(Q_i - Q_{i-1}) + (1+\kappa)(Q_{i+1} - Q_i)) \\
Q^R_{i+\frac{1}{2}} &= Q_{i+1} - \frac{\epsilon}{4}[(1=\kappa)(Q_{i+1} - Q_i) + (1-\kappa)(Q_{i+2} - Q_{i+1}))
\end{aligned}
\tag{16}
$$

where ϵ is '0' for the 'first-order accuracy' or '1' for the 'second-order accuracy'.

And, κ can be described with the following values, -1 as fully upwind, 0 as upwind-bias, 1/3 as third-order upwind bias, 1/2 as Leonad's QUICK, and 1 as central difference.

Here, we have implemented the model as second-order, and fully upwind scheme.

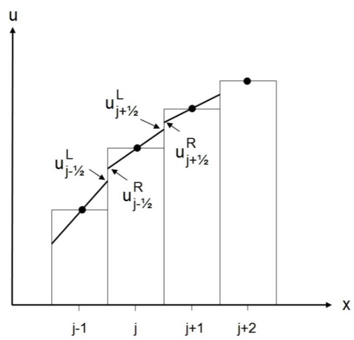

Figure 1: Piecewise Linear Extrapolation

In addition to the above formulation, we can also add the flux limiters θ along with MUSCL scheme for limiting the slope. There are various flux limiters are available, and we have used the 'minmod' limiter.

3.2 Minmod limiter

The Minmod limiter is defined as below,

$$\theta(r) = max(0, min(1, r)) \tag{17}$$

Although the minmod limiter is more diffusive than any other flux limiters, it still preserves the TVD property. And, hence a good flux limiter that can be used for the Riemann problem. This is implemented in the code.

3.3 Roe flux

The Roe flux is implemented using the reference paper by Cargo et al [1]. The Roe flux is given by the following formulation,

$$F_{i-\frac{1}{2}} = \frac{F_L + F_R}{2} - \frac{R(\lambda * ws)}{2} \tag{18}$$

where R is the right eigen vector, λ is the eigen values or wave speeds, and ws is the characteristic variables or wave strength.

The Right eigen vectors of the corresponding eigen values are given as follows,

$$R_{\bar{u}} = \frac{1}{\bar{a}^2} \begin{bmatrix} 1 \\ \bar{u} \\ \bar{v} \\ \bar{w} \\ 0 \\ 0 \\ \frac{\bar{V}^2}{2} + [\frac{\gamma-2}{\gamma-1}]X \end{bmatrix} \tag{19}$$

$$R_{\bar{u}\mp\bar{c}_a} = \begin{bmatrix} 0 \\ 0 \\ \mp\underline{\rho}\beta_z \\ \mp\underline{\rho}\beta_y \\ -S\sqrt{\underline{\rho}}\beta_z \\ S\sqrt{\underline{\rho}}\beta_y \\ \mp\underline{\rho}(\underline{\mathrm{v}}\beta_z - \bar{w}\beta_y) \end{bmatrix} \tag{20}$$

$$R_{\bar{u}\mp\bar{c}_s} = \frac{1}{\underline{\rho}\bar{a}^2} \begin{bmatrix} \underline{\rho}\alpha_s \\ \underline{\rho}\alpha_s(\underline{\mathrm{u}} \mp \underline{c_s}) \\ \underline{\rho}\alpha_s(\underline{\mathrm{v}} \mp \alpha_f \underline{c_f}\beta_y S) \\ \underline{\rho}\alpha_s(\underline{\mathrm{w}} \mp \alpha_f \underline{c_f}\beta_z S) \\ -\sqrt{\underline{\rho}}\alpha_f \bar{a}\beta_y \\ -\sqrt{\underline{\rho}}\alpha_f \bar{a}\beta_z \\ \sqrt{\underline{\rho}}\alpha_s \left[\bar{H}^* - \frac{\underline{B}^2}{\underline{\rho}} \mp \bar{u}\bar{c}_s\right] \mp \underline{\rho}\alpha_f \bar{c}_f S(\bar{v}\beta_y + \bar{w}\beta_z) - \sqrt{\underline{\rho}}\alpha_f \bar{a}|\underline{\mathrm{B}}| \end{bmatrix} \tag{21}$$

$$R_{\bar{u}\mp\bar{c}_f} = \frac{1}{\underline{\rho}\bar{a}^2} \begin{bmatrix} \underline{\rho}\alpha_f \\ \underline{\rho}\alpha_f(\underline{\mathrm{u}} \mp \underline{c_f}) \\ \underline{\rho}\alpha_f(\underline{\mathrm{v}} \mp \alpha_s \underline{c_s}\beta_y S) \\ \underline{\rho}\alpha_f(\underline{\mathrm{w}} \mp \alpha_s \underline{c_f}\beta_z S) \\ \sqrt{\underline{\rho}}\alpha_s \bar{a}\beta_y \\ \sqrt{\underline{\rho}}\alpha_s \bar{a}\beta_z \\ \sqrt{\underline{\rho}}\alpha_f \left[\bar{H}^* - \frac{\underline{B}^2}{\underline{\rho}} \mp \bar{u}\bar{c}_f\right] \mp \underline{\rho}\alpha_s \bar{c}_s S(\bar{v}\beta_y + \bar{w}\beta_z) - \sqrt{\underline{\rho}}\alpha_s \bar{a}|\underline{\mathrm{B}}| \end{bmatrix} \tag{22}$$

where $\bar{u}$, $\bar{c}_a$, $\bar{c}_s$, $\bar{c}_f$, are the bulk or total velocity, alfven velocity, slow, and fast magnetosonic speeds respectively. The corresponding parameters can be referenced in the article by Cargo at al. [1].

3.4 HLL flux

The HLL flux is one other method of describing the flux in this numerical calculations. The HLL flux is described as below with three different conditions,

$$\begin{aligned} &\lambda_L < 0 : F_{i-\frac{1}{2}} = F_L \\ &\lambda_R < 0 : F_{i-\frac{1}{2}} = F_R \\ &\lambda_L < 0 \quad \lambda_R > 0 : F_{i-\frac{1}{2}} = \frac{\lambda_R F_L - \lambda_L F_R + \lambda_L \lambda_R (Q_R - Q_L)}{\lambda_R - \lambda_L} \end{aligned} \tag{23}$$

where λ_L, and λ_R are the left, and right eigen values respectively.

3.5 Lax-Friedrichs flux

The Lax-Friedrichs flux is the simple, and special case of HLL. The flux is given as below,

$$F_{i-\frac{1}{2}} = \frac{F_L + F_R}{2} - max(\lambda)(\frac{Q_R - Q_L}{2}) \tag{24}$$

where λ is the eigen value.

4 Results and Analysis

4.1 SOD Shock - Euler Equations

The density plot is shown in the figure 2 below which clearly shows the rarefaction phase or expansion fan propagating in the Region1, and the discontinuity at approximately at position 0.7. And, then we can also see the shock wave propagating, and currently seen close to position 0.9 at the end of the simulation in region 2.

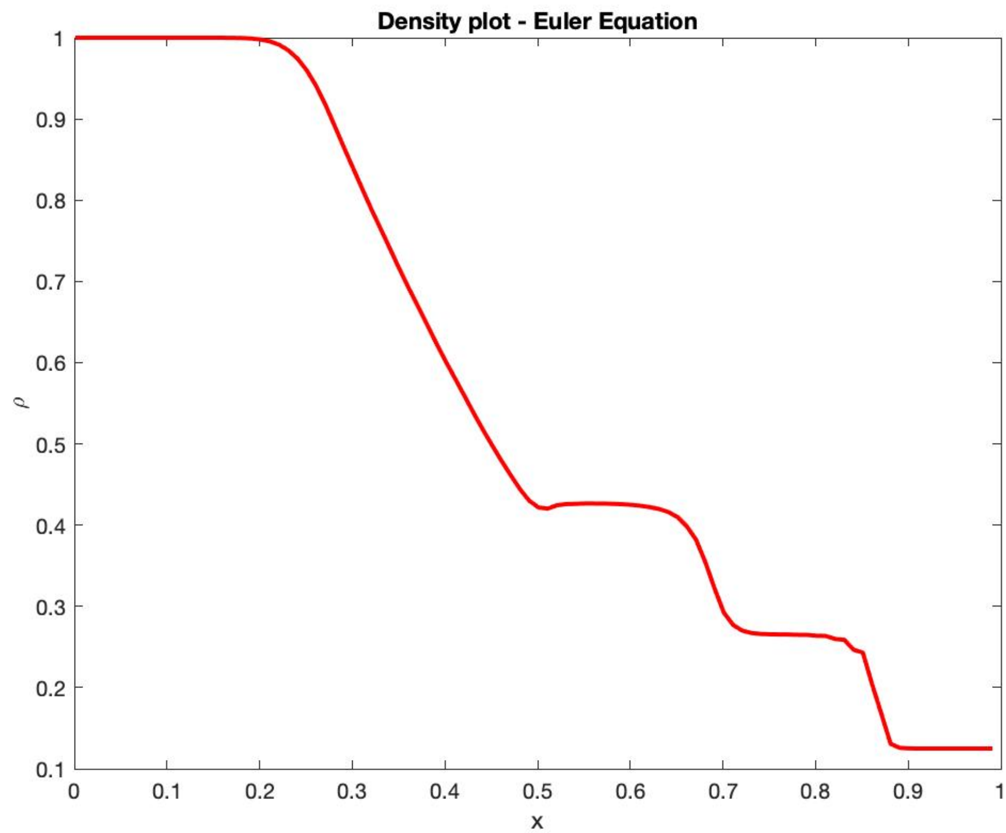

Figure 2: Density Plot for SOD shock using Euler Equations

In figure 3, we can visualize the bulk velocity of the Euler SOD shock problem. The velocity is increasing between the positions of rarefaction, and shock waves as shown in figure 2. The velocity gradually increases in region 1, and stays constant, and at the shock the velocity changes drastically.

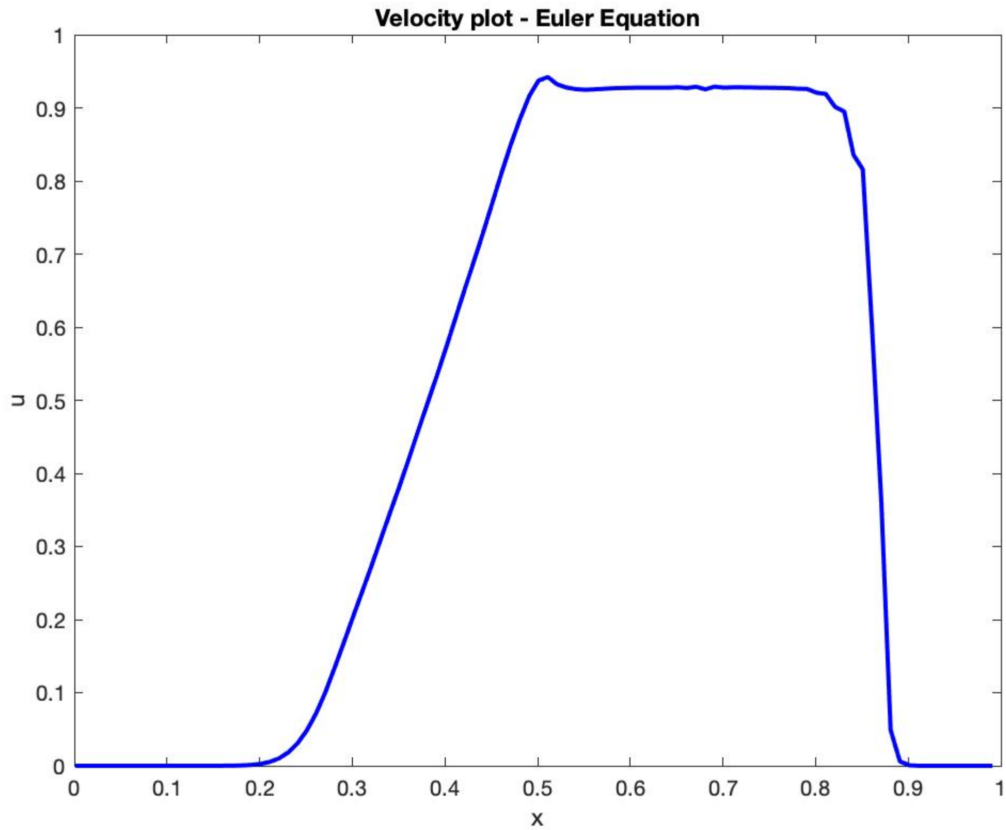

Figure 3: Bulk velocity Plot for SOD shock using Euler Equations

The energy plot in figure 4 also signifies the phenomenon of drastic change in energy at the shock. The energy gradually decreases along the expansion fan, and then increases, and changes drastically at the shock location at the end of the simulation.

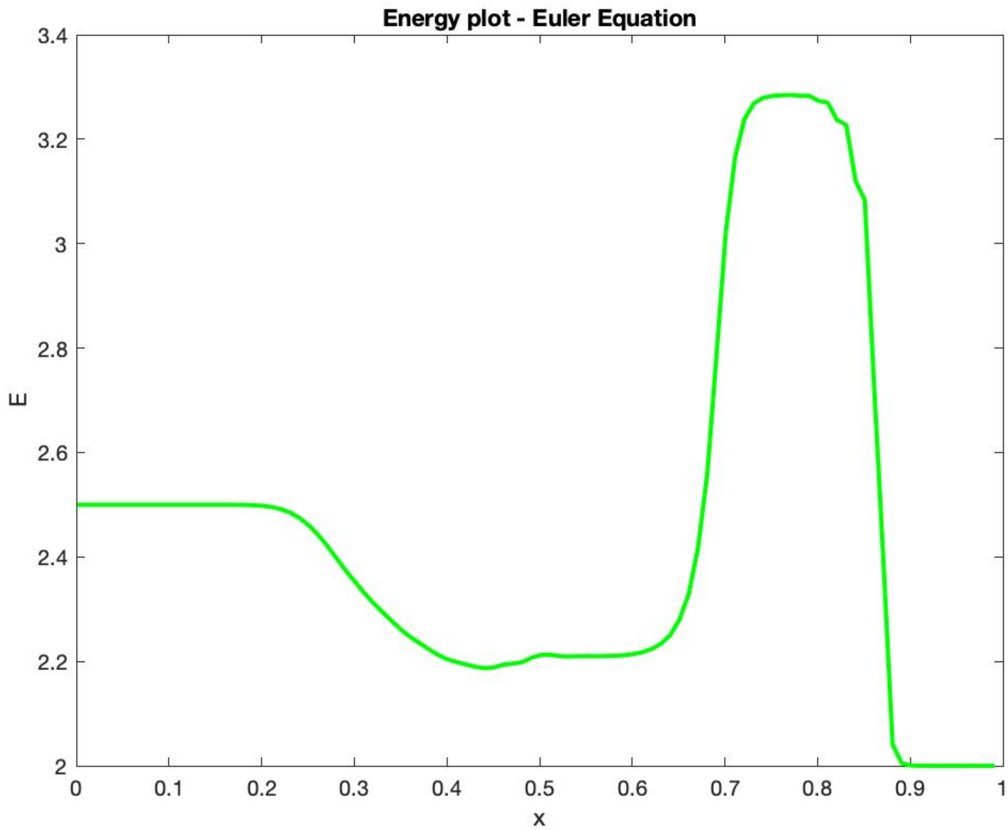

Figure 4: Energy Plot for SOD shock using Euler Equations

The figure 5 describes the pressure plot which clearly shows the pressure values at the rarefaction, and shock locations in region 1, and region 2 respectively, similar to the density plot.

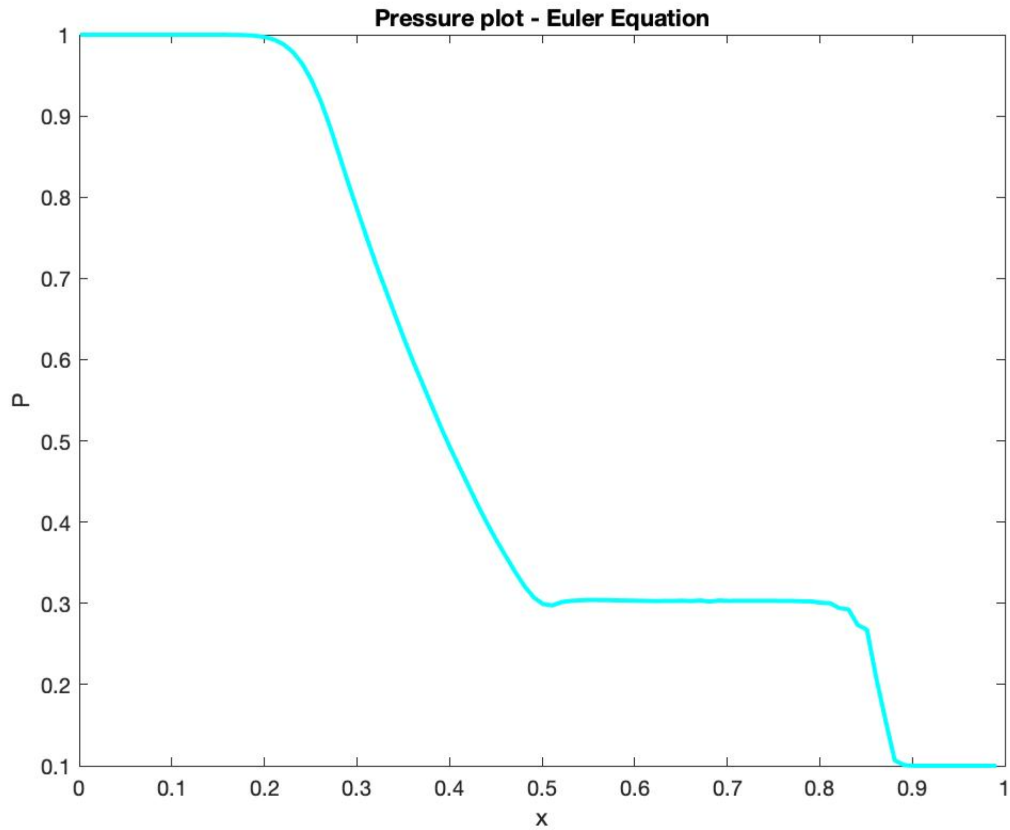

Figure 5: Pressure Plot for SOD shock using Euler Equations

The figures 6-9 describes the density, velocity, energy, and pressure plots with CFL number =1.1. The plots clearly show the presence of dispersions or oscillations at the shock locations where the gradient changes sharply.

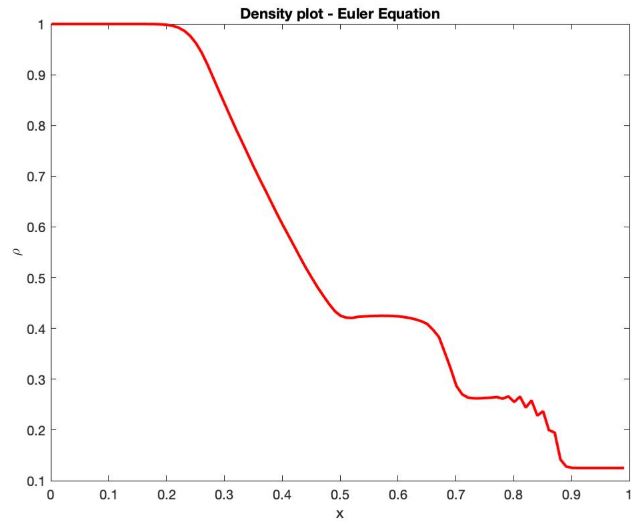

Figure 6: Density Plot for SOD shock using Euler Equations with CFL=1.1

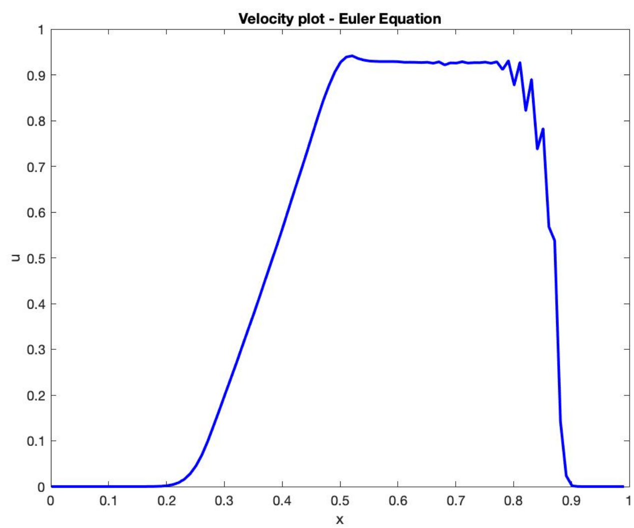

Figure 7: Velocity Plot for SOD shock using Euler Equations with CFL=1.1

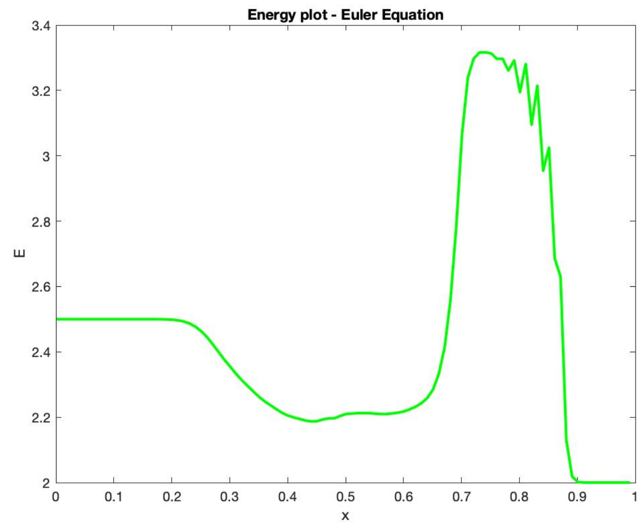

Figure 8: Energy Plot for SOD shock using Euler Equations with CFL=1.1

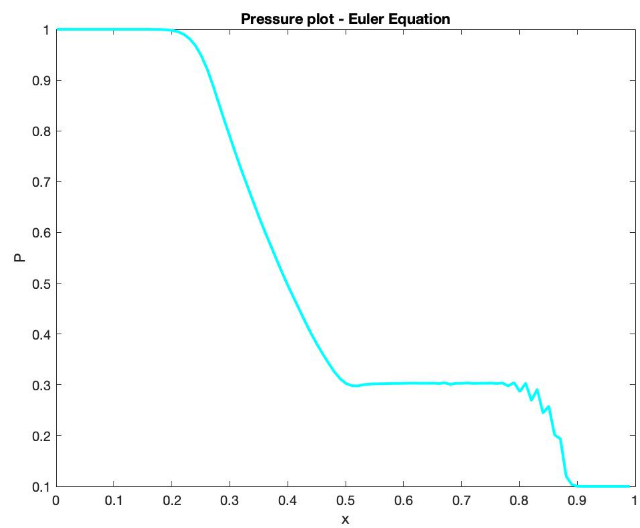

Figure 9: Pressure Plot for SOD shock using Euler Equations with CFL=1.1

4.2 SOD Shock - Gkeyll

As we can see from the figure-10, and figure-6, the discontinuity has moved a little further in the euler equations than the Vlasov-Maxwell Equations. But, apart from that, both the models show the similar rarefaction wave in region 1, and shock propagation at the region 2. The Knudsen number (Kn) used here is 0.0001, and hence it describes as the no-slip boundary condition.

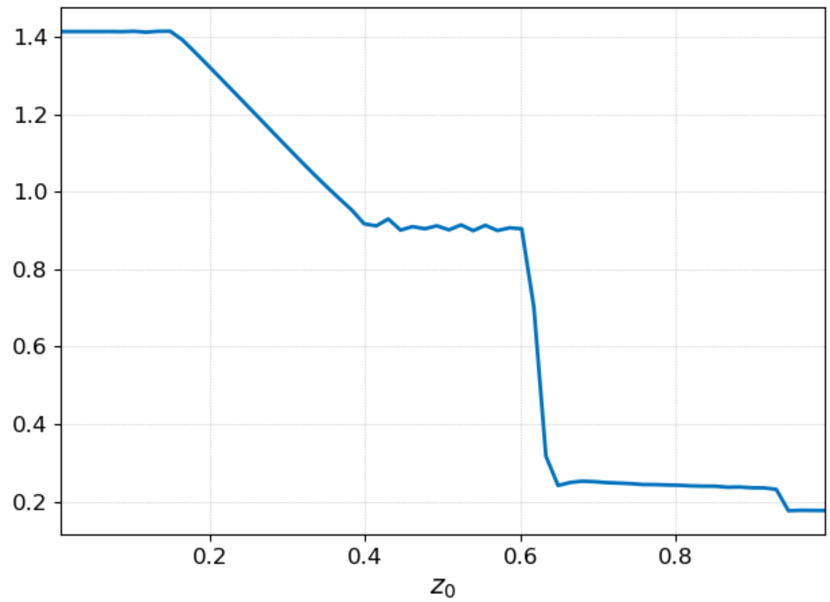

Figure 10: Density Plot for SOD shock using Gkeyll Vlasov-Maxwell Equations

4.3 SOD Shock - Ideal-MHD

The Ideal MHD equations used to simulate SOD shock problem had issues with 'NaN' for running till the end of time. So, I have plotted the first time step of density, velocity, Energy, and pressure as shown in figures 11-14 below. The figure 11 shows the density plot in which we can clearly see the shock wave appearing, and will propagate in region 2. We can also see the discontinuity, and probable rarefaction wave in region 1. This can also be visualized in velocity plot where the velocity increases closer to the discontinuity, indicating the shock wave propagation. The energy, and pressure plots also show the variations at this locations. The flux used in these plots are 'HLL flux'. The Roe flux is also implemented but the issue of 'NaNs' was present in the Roe flux as well. These implementations can be seen in the MATLAB code.

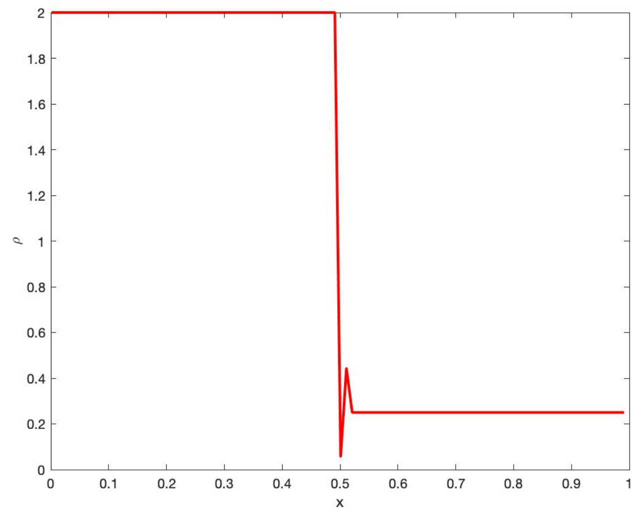

Figure 11: Density Plot for SOD shock using Ideal MHD Equations

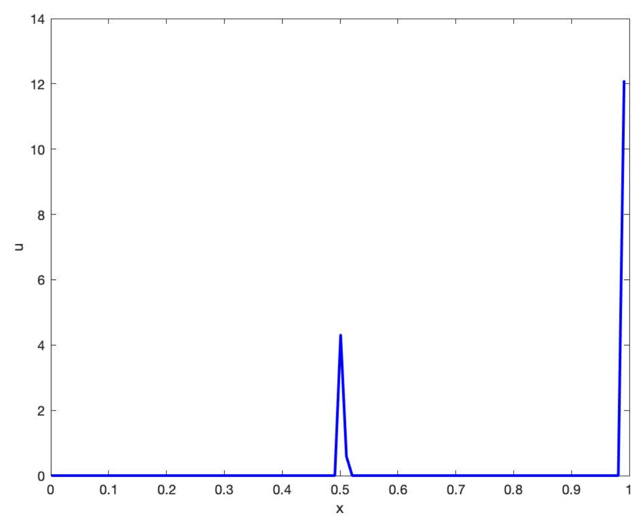

Figure 12: Velocity Plot for SOD shock using Ideal MHD Equations

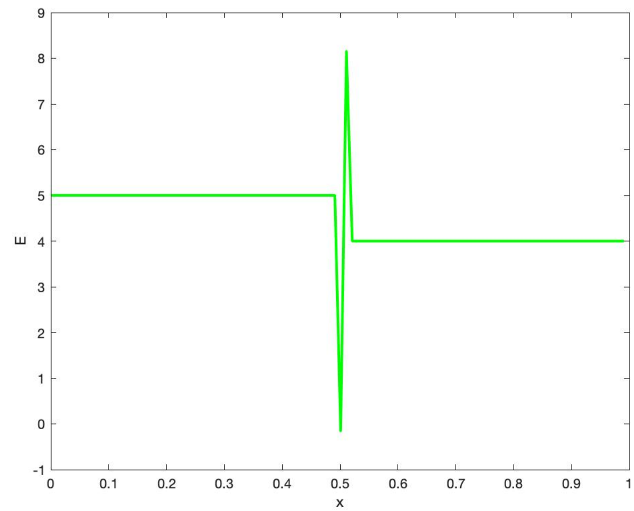

Figure 13: Energy Plot for SOD shock using Ideal MHD Equations

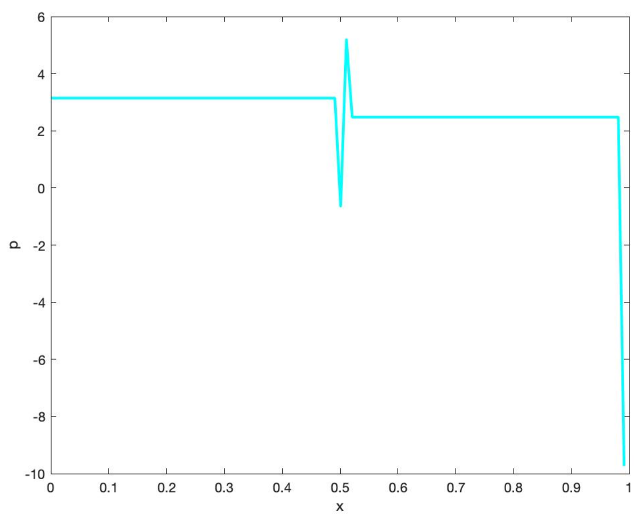

Figure 14: Pressure Plot for SOD shock using Ideal MHD Equations

In order to make sure the Ideal MHD equation shows the rarefaction, and shock waves appearing in the solution, I implemented with the simple flux called Lax-Friedrichs flux. The solution is shown in figure-15. Although this had the issue of 'NaNs', I was able to visualize the rarefaction, and shock waves appearing in the solution as shown in figure-

15. The solution matches closely to the actual solution except at the right boundary even without running till the end time.

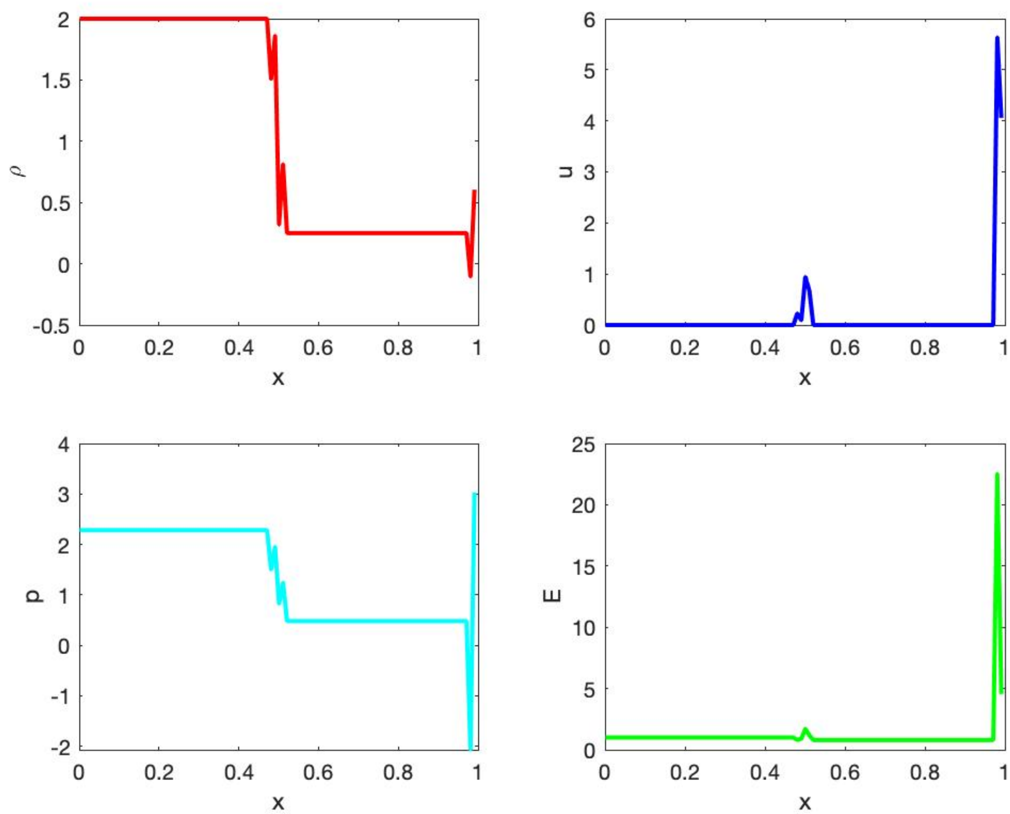

Figure 15: Plots for SOD shock using Ideal MHD Equations

5 Conclusion

In this project, I have successfully implemented the SOD shock problem using the Euler, and Ideal MHD equations in the MATLAB. Also, the kinetic solution is also compared with the fluid solutions using the open source plasma simulation software called 'Gkeyll'. The Euler equations show the results accurately which is implemented with Roe flux.Although the Ideal MHD implementation had some issues of running the program till the end time, I was able to show that the rarefaction, and shock waves will propagate by implementing it using Lax-Friedrichs flux. In addition to that, HLL flux is implemented and the results are shown for that for Ideal MHD equations. The Roe flux is also implemented for the Ideal

MHD equations in MATLAB. The entire program is implemented using the finite volume method called MUSCL scheme with two step Runge-Kutta scheme. The MUSCL scheme is also implemented with second-order, and fully upwind scheme. The CFL number is used for dynamic time stepping.

6 References

1. Patricia Cargo and Gerard Gallice, Roe Matrices for Ideal MHD and Systematic Construction of Roe Matrices for Systems of Conservation Laws, Journal of Computational Physics 136, 446-466 (1997).