Computational Fluid Dynamics

GUO QILONG

April 21, 2024

Contents

Project 2

0.1 Problem

Solve Euler equations

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} = 0,\\
&\frac{\partial \rho u}{\partial t} + \frac{\partial (\rho u^2 + p)}{\partial x} = 0,\\
&\frac{\partial \rho v}{\partial t} + \frac{\partial \rho u v}{\partial x} = 0,\\
&\frac{\partial e}{\partial t} + \frac{\partial u(e + p)}{\partial x} = 0
\end{aligned}$$

numerically using the HLLC method. Consider the following two problems:

1. A Riemann problem with the initial condition

$$\rho(x,0) = \begin{cases} 1, & \text{if } x < 0, \\ 3, & \text{if } x > 0, \end{cases} \quad u(x,0) = 0, \; v(x,0) = \begin{cases} -1, & \text{if } x < 0, \\ 1, & \text{if } x > 0, \end{cases} \quad p(x,0) = 1.$$

   Present the solution at $t = 1, 2, 3$.

2. The Woodward-Colella blast wave problem:

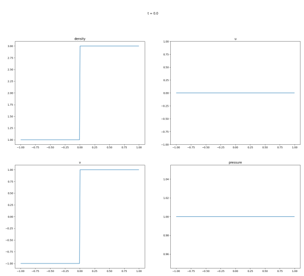
$$\rho(x,0)=1,\ u(x,0)=0,\ v(x,0)=\begin{cases}-10, & \text{if } x\in(0,0.5),\\ 20 & \text{if } x\in(0.5,1)\end{cases},\ p(x,0)=\begin{cases}1000, & \text{if } x\in(0,0.1),\\ 0.01, & \text{if } x\in(0.1,0.9),\\ 100, & \text{if } x\in(0.9,1),\end{cases}$$

with reflective boundary conditions

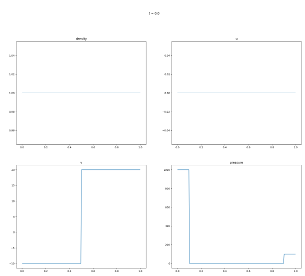
$$u(0,t)=u(1,t)=0.$$

Present the solutions at time $t=0.01,0.02,0.03,0.038$.

Preview of the Initial Value:



(a) IV of Problem 1



(b) IV of Problem 2

0.2 Numerical Scheme

The main Numerical Scheme is to utilize Finite Volume Method and HLLC method.

0.2.1 Simplification

With

$$p=(\gamma-1)\left[e-\frac{1}{2}\rho(u^2+v^2)\right], \tag{1}$$

one dimensional Euler equations with two velocity components becomes:

$$\begin{aligned}
&\frac{\partial\rho}{\partial t}+u\frac{\partial\rho}{\partial x}+\rho\frac{\partial u}{\partial x}=0,\\
&\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+\frac{1}{\rho}\frac{\partial p}{\partial x}=0,\\
&\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}=0,\\
&\frac{\partial p}{\partial t}+\gamma p\frac{\partial u}{\partial x}+u\frac{\partial p}{\partial x}=0
\end{aligned} \tag{2}$$

Therefore

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{B}(\mathbf{w})\frac{\partial \mathbf{w}}{\partial x} = 0, \tag{3}$$

where

$$\mathbf{w} = \begin{pmatrix} \rho \\ u \\ v \\ p \end{pmatrix}, \quad \mathbf{B}(\mathbf{w}) = \begin{pmatrix} u & \rho & 0 & 0 \\ 0 & u & 0 & 1/\rho \\ 0 & 0 & u & 0 \\ 0 & \gamma p & 0 & u \end{pmatrix}.$$

The eigenvalues and eigenvectors of $\mathbf{B}$ are

$$\lambda^1 = u - c, \qquad \lambda^2 = u, \qquad \lambda^3 = u, \qquad \lambda^4 = u + c;$$

$$\mathbf{r}^1 = \begin{pmatrix} 1 \\ -c/\rho \\ 0 \\ c^2 \end{pmatrix}, \quad \mathbf{r}^2 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{r}^3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad \mathbf{r}^4 = \begin{pmatrix} 1 \\ c/\rho \\ 0 \\ c^2 \end{pmatrix},$$

where

$$c = \sqrt{\frac{\gamma p}{\rho}}.$$

Let $\mathbf{q} = (\rho, \rho u, \rho v, e)^T$, the Euler equations can be written as:

$$\frac{\partial \mathbf{q}}{\partial t} + \frac{\partial \mathbf{f}(\mathbf{q})}{\partial x} = 0,$$

where

$$\mathbf{f}(\mathbf{q}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ u(e+p) \end{pmatrix}, \quad p = (\gamma - 1)\left[e - \frac{1}{2}\rho(u^2 + v^2)\right]$$

In this project, set $\gamma = 1.4$.

0.2.2 Spatial discretization

$$-1 = x_{-N+\frac{1}{2}} < \cdots < 0 = x_{\frac{1}{2}} < \cdots < x_{N-\frac{1}{2}} < x_{N+\frac{1}{2}} = 1,$$

where $\Delta x = \frac{1}{N}$.

Finite Volume Method:

$$
\begin{aligned}
\mathbf{Q}_i^n &= \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{q}(x, n\Delta t)dx \\
&= \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \begin{pmatrix} \rho(x,y,n\Delta t) \\ \rho u(x,y,n\Delta t) \\ \rho v(x,y,n\Delta t) \\ e(x,y,n\Delta t) \end{pmatrix} dx \\
&\approx \begin{pmatrix} \rho(x_i, n\Delta t) \\ \rho(x_i, n\Delta t)u(x_i, n\Delta t) \\ \rho(x_i, n\Delta t)v(x_i, n\Delta t) \\ e(x_i, n\Delta t) \end{pmatrix} := \begin{pmatrix} \rho_i^n \\ \rho_i^n u_i^n \\ \rho_i^n v_i^n \\ e_i^n \end{pmatrix},
\end{aligned}
$$

where Δt is chosen small to ensure the stability.

Conservative scheme:

$$
\mathbf{Q}_i^{n+1} = \mathbf{Q}_i^n - \frac{\Delta t}{\Delta x}(\mathbf{F}_{i+\frac{1}{2}}^n - \mathbf{F}_{i-\frac{1}{2}}^n), \quad i = (-N, \cdots,)1\cdots N,
$$

HLLC solver:

$$
\mathbf{F}(\mathbf{Q}_i^n, \mathbf{Q}_{i+1}^n) = \mathbf{F}_{i+\frac{1}{2}}^n = \begin{cases} \mathbf{f}(\mathbf{Q}_i^n), & \text{if } \lambda_L \geq 0, \\ \mathbf{F}_L^* = \mathbf{f}(\mathbf{Q}_i^n) + \lambda_L(\mathbf{Q}_L^* - \mathbf{Q}_i^n), & \text{if } \lambda_L < 0 < \lambda^*, \\ \mathbf{F}_R^* = \mathbf{f}(\mathbf{Q}_{i+1}^n) + \lambda_R(\mathbf{Q}_R^* - \mathbf{Q}_{i+1}^n), & \text{if } \lambda_* \leq 0 < \lambda_R, \\ \mathbf{f}(\mathbf{Q}_{i+1}^n), & \text{if } \lambda_R \leq 0. \end{cases}
$$

where

$$
\lambda_L = \min\{u_i^n - \sqrt{\frac{\gamma p_i^n}{\rho_i^n}},\ u_{i+1}^n - \sqrt{\frac{\gamma p_{i+1}^n}{\rho_{i+1}^n}}\}, \quad \lambda_R = \max\{u_i^n + \sqrt{\frac{\gamma p_i^n}{\rho_i^n}},\ u_{i+1}^n + \sqrt{\frac{\gamma p_{i+1}^n}{\rho_{i+1}^n}}\},
$$

$$
\lambda^* = \frac{\lambda_R \rho_{i+1}^n u_{i+1}^n - \lambda_L \rho_i^n u_i^n - [\rho_{i+1}^n (u_{i+1}^n)^2 + p_{i+1}^n - \rho_i^n (u_i^n)^2 - p_i^n]}{\lambda_R \rho_{i+1}^n - \lambda_L \rho_i^n - (\rho_{i+1}^n u_{i+1}^n - \rho_i^n u_i^n)},
$$

$$
\rho_L^* = \frac{\lambda_L - u_i^n}{\lambda_L - \lambda^*}\rho_i^n, \qquad \rho_R^* = \frac{\lambda_R - u_{i+1}^n}{\lambda_R - \lambda^*}\rho_{i+1}^n,
$$

$$
e_L^* = \frac{\lambda_L - u_i^n}{\lambda_L - \lambda^*}e_i^n + \frac{\lambda^* - u_i^n}{\lambda_L - \lambda^*}p_i^n + \rho_i^n \frac{\lambda^*(\lambda^* - u_i^n)(\lambda_L - u_i^n)}{\lambda_L - \lambda^*},
$$

$$
e_R^* = \frac{\lambda_R - u_{i+1}^n}{\lambda_R - \lambda^*}e_{i+1}^n + \frac{\lambda^* - u_{i+1}^n}{\lambda_R - \lambda^*}p_{i+1}^n + \rho_{i+1}^n \frac{\lambda^*(\lambda^* - u_{i+1}^n)(\lambda_L - u_{i+1}^n)}{\lambda_R - \lambda^*},
$$

$$
\mathbf{Q}_L^* = (\rho_L^*, \rho_L^*\lambda^*, \rho_L^* v_i^n, e_L^*)^T, \qquad \mathbf{Q}_R^* = (\rho_R^*, \rho_R^*\lambda^*, \rho_R^* v_{i+1}^n, e_R^*)^T.
$$

For simpler programming, i.e.

$$
e_L^* = \frac{\lambda_L}{\lambda_L - \lambda^*}e_i^n + \frac{\lambda_*}{\lambda_L - \lambda^*}[p_i^n + \rho_i^n (u_i^n)^2] - \frac{u_i^n(e_i^n + p_i^n)}{\lambda_L - \lambda^*} + \frac{\lambda_*^2 \lambda_L}{\lambda_L - \lambda^*}\rho_i^n - \frac{\lambda_*(\lambda_* + \lambda_L)}{\lambda_L - \lambda^*}\rho_i^n u_i^n,
$$

so that I could plug in the terms of $\mathbf{Q}$ and $\mathbf{F}$.

0.2.3 Boundary conditions:

1. For first question, indeed no boundary condition is needed, but the infinite ending is impossible, so mannully choose two ending points -1 and 1, determined by reflective boundary condition, and make sure the evolution will not hit the boundary in the limited time.

2. For second question, which is with reflective boundary conditions, create ghost cell averages by extrapolation. The results are simply below :

$$\begin{aligned} u_0^n &= -u_1^n, \quad u_N^n = -u_{N+1}^n \\ v_0^n &= v_1^n, \quad v_N^n = v_{N+1}^n \\ \rho_0^n &= \rho_1^n, \quad \rho_N^n = \rho_{N+1}^n \\ p_0^n &= p_1^n, \quad p_N^n = p_{N+1}^n \\ e_0^n &= e_1^n, \quad e_N^n = e_{N+1}^n. \end{aligned}$$

0.2.4 2nd-Order accuracy

To obtain the 2nd-order accuracy, use:

Linear Reconstruction:

$$\sigma_{i,j} = \text{minmod}\left(\frac{\mathbf{Q}_{i+1} - \mathbf{Q}_i}{\Delta x}\right), \tag{4}$$

Semi-discritization:

$$\frac{d\mathbf{Q}_i}{dt} = -\frac{1}{\Delta x}\left[\mathbf{F}\left(\mathbf{Q}_i + \frac{\Delta x}{2}\sigma_i, \mathbf{Q}_{i+1} - \frac{\Delta x}{2}\sigma_{i+1}\right) - \mathbf{F}\left(\mathbf{Q}_{i-1} + \frac{\Delta x}{2}\sigma_{i-1}, \mathbf{Q}_i - \frac{\Delta x}{2}\sigma_i\right)\right], \tag{5}$$

where $\mathbf{F}(\cdot,\cdot)$ is the numerical fluxes in the previous first-order scheme, just plug in the reconstructed terms and do it all again.

ODEs solver

Apply second-order time intergrators to the ODEs system:

$$\frac{\partial \mathbf{Q}}{\partial t} = \mathbf{R}(\mathbf{Q}),$$

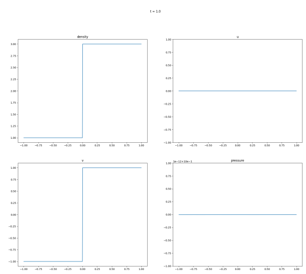
where $\mathbf{R}_i(\mathbf{Q})$ is Equation 5. Utilize Midpoint Method:

$$\begin{aligned} \mathbf{Q}^{n+\frac{1}{2}} &= \mathbf{Q}^n + \frac{\Delta t}{2}\mathbf{R}(\mathbf{Q}) \\ \mathbf{Q}^{n+1} &= \mathbf{Q}^n + \Delta t \mathbf{R}(\mathbf{Q}^{n+\frac{1}{2}}) \end{aligned} \tag{6}$$

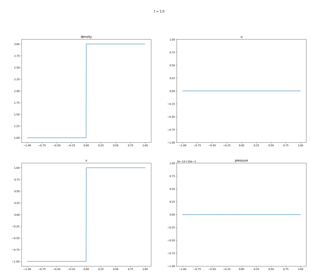
0.3 Results

Results by HLLC for Problem 1 and Problem 2 are as follow below in Figure 2 and Figure 3, with both 1st-order and 2nd-order. The source code is attached in the .zip file.

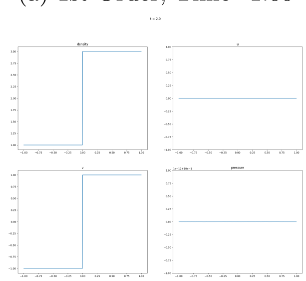
Problem 1



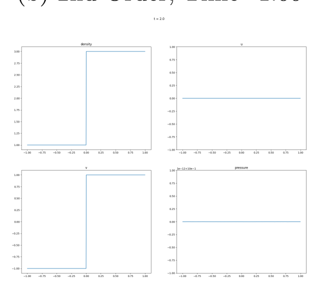
(a) 1st-Order, Time=1.00



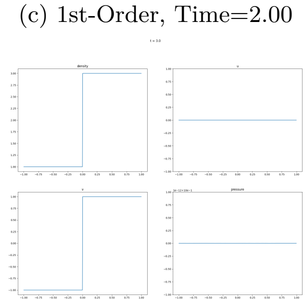
(b) 2nd-Order, Time=1.00



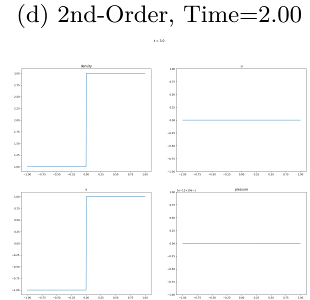
(c) 1st-Order, Time=2.00



(d) 2nd-Order, Time=2.00



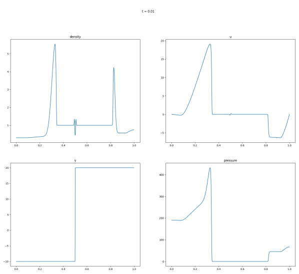
(e) 1st-Order, Time=3.00



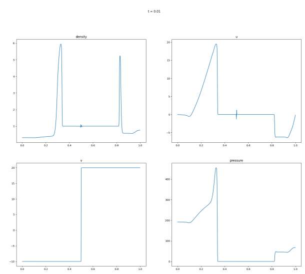
(f) 2nd-Order, Time=3.00

Figure 2: $x \in [-1, 1]$, $N = 500$, $\Delta t = 4/2000$

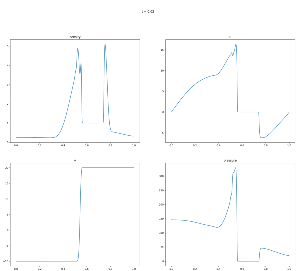
Problem 2



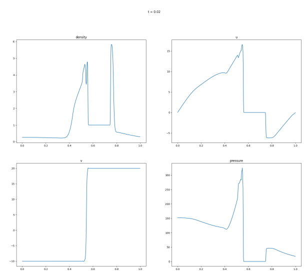
(a) 1st-Order, Time=0.01



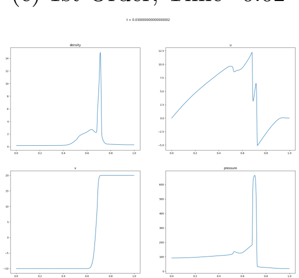
(b) 2nd-Order, Time=0.01



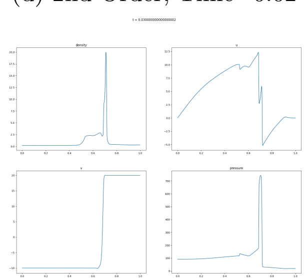
(c) 1st-Order, Time=0.02



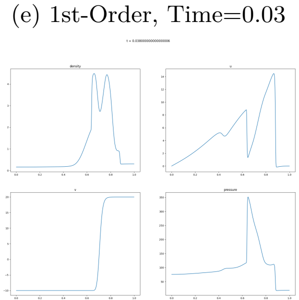
(d) 2nd-Order, Time=0.02



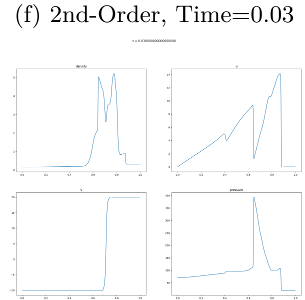
(e) 1st-Order, Time=0.03



(f) 2nd-Order, Time=0.03



(g) 1st-Order, Time=0.038



(h) 2nd-Order, Time=0.038

Figure 3: $x \in [0, 1]$, $N = 500$, $\Delta t = 0.04/2000$

1 Reference

1. Cai, Z. (2024, Feb.). Lecture 19 [Computational Fluid Dynamics].