Final Project of Numerical Methods for PDE

Finite Volume Method for Euler Equation

Zejian You
Columbia University
April 27, 2022

Abstract

Keywords:

1 Problem Description

- Mass Conservation

$$\rho_t + (\rho u)_x = 0.$$

- Momentum Conservation

$$(\rho u)_t + (\rho u^2 + p)_x = 0.$$

- Energy Conservation

$$E = \rho e + \frac{1}{2}\rho u^2.$$

$$E_t + (u(E+p))_x = 0.$$

To close the equation system, we need one more equation
- Equation of State

$$p = \rho e(\gamma - 1)$$

1.1 Hyperbolic structure of the 1D Euler equations

The governing equation could be written into matrix form:

$$\frac{\partial \mathbf{V}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{V})}{\partial x} = 0 \quad \text{or}$$

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

Introduce *primitive variable* $\mathbf{q} = [\rho, u, p]^T$ and write above equation in quasi-linear form with primitive variable:

$$\begin{bmatrix} \rho \\ u \\ p \end{bmatrix}_t + \begin{bmatrix} u & \rho & 0 \\ 0 & u & 1/\rho \\ 0 & \gamma p & u \end{bmatrix} \begin{bmatrix} \rho \\ u \\ p \end{bmatrix}_x = 0$$

$$\mathbf{q}_t + \frac{\partial \bar{\mathbf{F}}(\mathbf{q})}{\partial \mathbf{q}} \mathbf{q}_x = 0$$

1.2 Finite Volume Method

Based on Green's formula, we have

$$\nabla \cdot \mathbf{F}(\mathbf{V}) = \oint_{\partial T_j} \mathbf{F}(\mathbf{V}) \cdot \mathbf{n} ds$$

For finite volume discretization, we have[1]

$$\oint_{\partial T_j} \mathbf{F}(\mathbf{V}) \cdot \mathbf{n} ds \approx \sum_{e_{ik} \in \partial T_j} \oint_{e_{jk}} \bar{\mathbf{F}}(\mathbf{V}_j, \mathbf{V}_k) \cdot \mathbf{n}_{jk} dl$$

$$Q_j^{n+1} = Q_j^n - \frac{\Delta t_n}{\Delta x_j} (F_{j+1/2}^n - F_{j-1/2}^n) \tag{1}$$

1.3 Eigenvalue and Eigenvector of Euler Flux Jacobian

1. Flux Jacobian of primitive variable is equal to

$$\frac{\partial \bar{\mathbf{F}}(\mathbf{q})}{\partial \mathbf{q}} = \begin{bmatrix} u & \rho & 0 \\ 0 & u & 1/\rho \\ 0 & \gamma p & u \end{bmatrix}$$

The eigenvalues and corresponding eigenvectors are

$$\lambda_1 = u - c, \qquad \lambda_2 = u, \qquad \lambda_3 = u + c$$

$$\mathbf{r}_1 = \begin{bmatrix} -\rho/c \\ 1 \\ -\rho c \end{bmatrix} \quad \mathbf{r}_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{r}_3 = \begin{bmatrix} \rho/c \\ 1 \\ \rho c \end{bmatrix}$$

2. Flux of variable $\mathbf{V}$ equal to

$$\bar{\mathbf{F}}(\mathbf{V}) = \begin{bmatrix} V_2 \\ \frac{V_2^2}{V_1} + \left(V_3 - \frac{V_2^2}{2V_1}\right)(\gamma - 1) \\ \frac{\gamma V_2 V_3}{V_1} - \frac{V_2^3}{2V_1^2}(\gamma - 1) \end{bmatrix}$$

Flux Jacobian of variable $\mathbf{V}$ equal to [1]

[1] with $p = (E - \frac{1}{2}\rho u^2)(\gamma - 1)$, and $H = \frac{E+p}{\rho}$

$$\frac{\partial \bar{\mathbf{F}}(\mathbf{V})}{\partial \mathbf{V}} = \begin{bmatrix} u & 1 & 0 \\ \frac{\gamma-3}{2}u^2 & (3-\gamma)u & \gamma-1 \\ \frac{\gamma-1}{2}u^3 - uH & H-(\gamma-1)u^2 & \gamma u \end{bmatrix}$$

Then the Euler Equation becomes

$$\mathbf{V}_t + \frac{\partial \bar{\mathbf{F}}(\mathbf{V})}{\partial \mathbf{V}} \mathbf{V}_x = 0$$

The eigenvalues and corresponding eigenvectors of Jacobian matrix is

$$\lambda_1 = u - c, \qquad \lambda_2 = u, \qquad \lambda_3 = u + c$$

$$\mathbf{r}_1 = \begin{bmatrix} 1 \\ u-c \\ H-uc \end{bmatrix} \quad \mathbf{r}_2 = \begin{bmatrix} 1 \\ u \\ \frac{1}{2}u^2 \end{bmatrix} \quad \mathbf{r}_3 = \begin{bmatrix} 1 \\ u+c \\ H+uc \end{bmatrix}$$

Here $c = \sqrt{(\gamma-1)(H-\frac{1}{2}u^2)}$. [2]

- **linear degenerate**

1.4 Conservative Flux

1.4.1 Lax-Friedrichs Numerical Flux

1.4.2 Refined Lax-Friedrichs Numerical Flux

1.4.3 Roe Numerical Flux

[3]

- **Roe Average**

$$\bar{u} = \frac{\sqrt{\rho_r}u_r + \sqrt{\rho_l}u_l}{\sqrt{\rho_r} + \sqrt{\rho_l}}, \quad \bar{H} = \frac{\sqrt{\rho_r}H_r + \sqrt{\rho_l}H_l}{\sqrt{\rho_r} + \sqrt{\rho_l}} \tag{2}$$

2 Riemann Problem

2.1 Exact Riemann Problem

- **Rankine-Hugoniot jump condition**
  For a shock wave connecting a known left or right state q^* and an unknown middle state q, the moving speed s of the shock should obey Eq.(3).[2]

$$s(q^* - q) = f(q^*) - f(q) \tag{3}$$

- **Hugoniot locus**
- **Lax entropy condition**
  This physical entropy condition is equivalent to the mathematical condition that for a 1-shock to be physically relevant, the 1-characteristics must impinge on the shock (the Lax entropy condition). If the entropy condition is violated, the 1-characteristics would spread out, allowing the insertion of an expansion fan (rarefaction wave).

  - Shocks appear in regions where characteristics converge, as in the traffic jam example above.
  - Rarefactions appear in regions where characteristics are spreading out, as in the green light example.[2]

- **Integral curves**
- **Riemann invariants**
- **similarity solutions**
-

[4]

2.2 Approximate Riemann Problem

Besides the exact Riemann solution, we could also obtain an approximate solution by linearizing the Euler Equation and replacing Jacobian of flux $\frac{\partial \mathbf{F}(\mathbf{V})}{\partial \mathbf{q}}$ by a linear operator $\hat{\mathbf{A}}(q_l, q_r)$ depending on the left and right status [3]. Furthermore, this approximate linear operator should satisfy:

1. Consistency: $\hat{\mathbf{A}}(q_l, q_r) \rightarrow f'(q)$ as $q_l, q_r \rightarrow q$
2. Hyperbolicity: $\hat{\mathbf{A}}$ must be diagonalizable with real eigenvalues, so that we can define the waves and speeds needed in the approximate solution.
3. Conservation: $\hat{\mathbf{A}}(q_l, q_r)(q_r - q_l) = f(q_r) - f(q_l)$

One common linear operator is the flux Jacobian of an average state:

$$\hat{\mathbf{A}}(q_l, q_r) = f'(\hat{q}), \quad \hat{q} \in [q_l, q_r] \tag{4}$$

For the system of hyperbolic equations, we could solve the system by decomposing the linear operator $\hat{\mathbf{A}}$ into the eigenvectors $\mathbf{R}$ and the matrix that eigenvalues on the diagonal $\boldsymbol{\Lambda}$ so that

$$\begin{aligned}
\mathbf{q_t} + \hat{\mathbf{A}}\mathbf{q_x} &= 0 \\
\mathbf{q_t} + \mathbf{R}\boldsymbol{\Lambda}\mathbf{R^{-1}}\mathbf{q_x} &= 0 \\
\mathbf{R^{-1}}\mathbf{q_t} + \boldsymbol{\Lambda}\mathbf{R^{-1}}\mathbf{q_x} &= 0 \\
\mathbf{w_t} + \boldsymbol{\Lambda}\mathbf{w_x} &= 0 \\
w_k(x,t) &= w_k(x - \lambda_k t, 0) \\
\mathbf{q}(\mathbf{x},\mathbf{t}) &= \sum_k w_k(x - \lambda_k t, 0)\mathbf{r_k}
\end{aligned}$$

2.3 Newton Iteration

2.3.1 Stationary Solution

For the stationary solution, we would have following iteration scheme [1]

$$\alpha \|\mathbf{R}_j^{(n)}\|_{l^1} \delta \mathbf{U}_j^{(n)} + \sum_{e_j k \in \partial T_j} \int_{e_{jk}} \left(\frac{\partial \bar{\mathbf{F}}^{(n)}}{\partial \mathbf{U}_j} \delta \mathbf{U}_j^{(n)} \right) \cdot \mathbf{n}_{jk} dl + \sum_{e_j k \in \partial T_j} \int_{e_{jk}} \left(\frac{\partial \bar{\mathbf{F}}^{(n)}}{\partial \mathbf{U}_k} \delta \mathbf{U}_k^{(n)} \right) \cdot \mathbf{n}_{jk} dl = -\mathbf{R}_j^{(n)}$$

1. Input $\mathbf{U}^{(0)}$ as the initial guess, and set $n = 0$

2. Use an approximate solver for the system to get a $\delta\mathbf{U}^{(n)}$
3. update $\mathbf{U}^{(n+1)}$ by $\mathbf{U}^{(n)} + \tau\delta\mathbf{U}^{(n)}$
4. Reconstruct $\mathbf{U}^{(n+1)}$ using its cell mean values to get a piecewise polynomial expression on each cell
5. Check if the residual $\mathbf{R}^{n+1}$ is small enough

2.3.2 PDE

2.4 Linear Multigrid Method for Jacobian Matrix

[5]

3 Error Analysis

4 Conclusion

References

1. Ruo Li, Xin Wang, and Weibo Zhao. A multigrid block LU-SGS algorithm for euler equations on unstructured grids. page 26.

2. David I. Ketcheson, Randall J. LeVeque, and Mauricio J. del Razo. *Chapter 13: Approximate solvers for the Euler equations of gas dynamics*. SIAM. https://epubs.siam.org/doi/pdf/10.1137/1.9781611976212.ch13.

3. P.L Roe. Approximate riemann solvers, parameter vectors, and difference schemes. 43(2):357–372.

4. Richard Sanders. An introduction to the finite volume method for the equations of gas dynamics. page 32.

5. Antony Jameson. Solution of the euler equations for two dimensional transonic flow by a multigrid method. 13(3):327–355.