计算流体力学期末大作业

朱林-2200011028

2025 年 6 月 19 日

1 数理算法原理

1.1 问题描述

1.1.1 物理情形

Sod 激波管问题是一个一维理想气体流动问题：无限长管道中，初始时刻 $(t=0)$ 在 $x=0$ 处有一薄膜分隔两侧气体：

- 左侧 $(x<0)$: 高压区，状态为 (ρ_L, u_L, p_L)
- 右侧 $(x>0)$: 低压区，状态为 (ρ_R, u_R, p_R)

薄膜在 $t=0^+$ 时刻瞬时破裂，两侧气体开始相互作用，产生复杂的波系结构。

1.1.2 标准初始条件

采用以下无量纲初始条件：

$$\begin{aligned} \text{左侧:}\quad & \rho_L = 1.0, u_L = 0.0, p_L = 1.0 \\ \text{右侧:}\quad & \rho_R = 0.125, u_R = 0.0, p_R = 0.1 \end{aligned}$$

1.2 控制方程

流动由一维欧拉方程描述：

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial f(\mathbf{U})}{\partial x} = 0 \tag{1}$$

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

其中总能密度 $E = \rho e = \rho(C_v T + \frac{1}{2}u^2)$。

1.3 Riemann 问题精确解

1.3.1 波系结构

根据空气动力学知识，该 Sod 激波管中可能出现三种波：

- 激波：流体密度、速度、压力均发生突变，满足 Rankine-Hugoniot (R-H) 关系式。

- 接触间断：流体仅密度发生突变，速度与压力不变。
- 膨胀波（稀疏波）：一种等熵波，其内部物理量连续、光滑，头、尾物理量连续但导数不连续(弱间断)，且 Riemann 不变量不变。

对于一维 sod 激波管问题，薄膜破裂后将形成向左传播的膨胀波、向右传播的接触间断和激波，如图1。这些波将流场划分为五个特征区域（如图2所示[1]）：

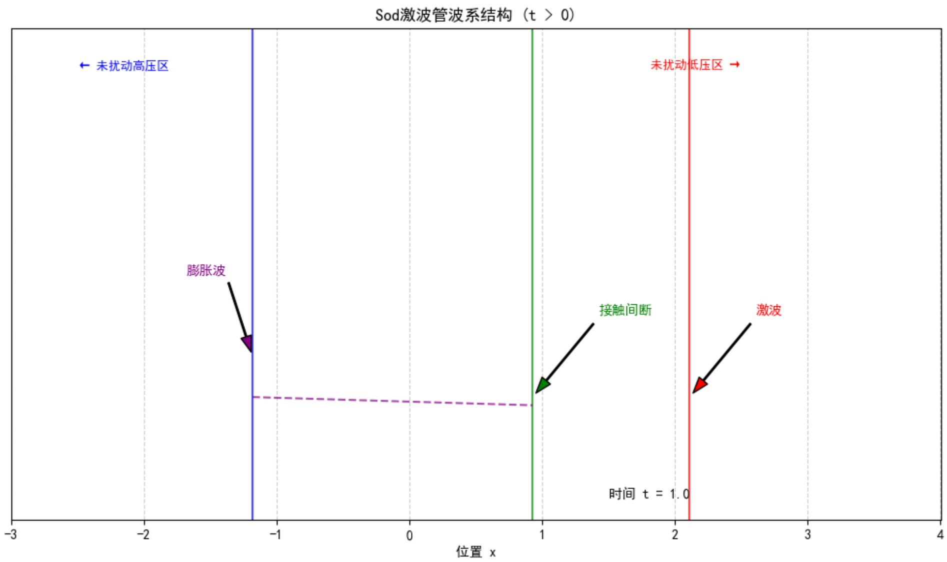

图 1: Sod 激波管典型波系结构（t>0）

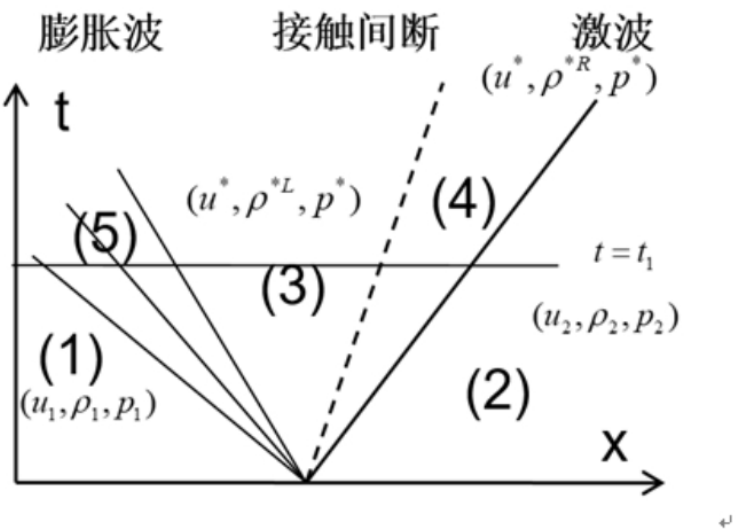

图 2: Sod 激波管典型波系结构

- **区域 1** 未扰动的左侧高压区，保持初始状态 (ρ_L, u_L, p_L)
- **区域 2** 未扰动的右侧低压区，保持初始状态 (ρ_R, u_R, p_R)

[1] url:https://blog.csdn.net/Nidebear/article/details/109300513

- **区域 3** 膨胀波后，状态为 (ρ_2, u^*, p^*)
- **区域 4** 接触间断与激波间均匀区，状态为 (ρ_3, u^*, p^*)
- **区域 5** 膨胀波内部

其中 u^* 和 p^* 为接触间断处的速度和压力，各波位置随时间线性变化：

$$x_{\text{left}} = -c_L t, \quad x_{\text{contact}} = u^* t, \quad x_{\text{shock}} = W_s t$$

W_s 为激波传播速度，$c_L = \sqrt{\gamma p_L / \rho_L}$ 为左侧声速。

1.3.2 解析解表达式

解析解通过求解以下方程组获得:

1-3 两区, 等熵关系式

$$\frac{p^*}{(\rho^{*L})^\gamma} = \frac{p_1}{(\rho_1)^\gamma} \tag{3}$$

$$u_1 + \frac{2c_1}{\gamma - 1} = u^* + \frac{2c^L}{\gamma - 1} \tag{4}$$

其中, $c^L = \sqrt{\gamma p^* / \rho^{*L}}$。

2-4 两区, 激波 R-H 关系式

$$\begin{cases} \rho_2 \left(u_2 - Z_2\right) = \rho^{*R} \left(u^* - Z_2\right) \\ \rho_2 u_2 \left(u_2 - Z_2\right) + p_2 = \rho^{*R} u^* \left(u^* - Z_2\right) + p^* \\ E_2 \left(u_2 - Z_2\right) + u_2 p_2 = E^{*R} \left(u^* - Z_2\right) + p^* u^* \end{cases} \tag{5}$$

以上变量说明从略。综上 5 个方程、5 个未知数，故方程组可解，求解方法为联立以上两个方程组，解出 3、4 区内速度对压力的依赖关系，有

$$u^* = u_1 - f(p^*, p_1, \rho_1)$$

其中，满足

$$f(p^*, p_i, \rho_i) = \frac{2c_i}{\gamma - 1} \left[\left(\frac{p^*}{p_i} \right)^{\frac{\gamma-1}{2\gamma}} - 1 \right] \tag{6}$$

注意到，激波、膨胀波前后速度-压力的依赖关系可写成统一的形式：

左波（激波或膨胀波）

$$u^* = u_1 - f(p^*, p_1, \rho_1)$$

右波（激波或膨胀波）

$$u^* = u_2 + f(p^*, p_2, \rho_2)$$

以上 u^*, p^* 表示 3、4 区内的速度与压力，其中

$$f(p^*, p_i, \rho_i) = \begin{cases} \frac{p^* - p_i}{\rho_i c_i \left[\frac{\gamma+1}{2\gamma} \left(\frac{p^*}{p_i} \right) + \frac{\gamma-1}{2\gamma} \right]^{1/2}}, & p^* > p_i \\ \frac{2c_i}{\gamma-1} \left[\left(\frac{p^*}{p_i} \right)^{\frac{\gamma-1}{2\gamma}} - 1 \right], & p^* < p_i \end{cases} \tag{7}$$

求解上式可得到 3、4 区内的压力，然后可以解得速度和密度。

膨胀波内部 对于膨胀波内部物理量的计算，首先由波头传播速度 $u_1 - c_1$ 与波尾传播速度 $u^* - c^{*L}$ 可计算膨胀波的范围。在膨胀波区内，利用特征相容关系和等熵关系计算物理量，可利用简单波的特性来简化计算。以下直接给出各个物理量的计算表达式：

$$
\begin{aligned}
&c(t,x) = \frac{\gamma-1}{\gamma+1}\left(u_1 - \frac{x}{t}\right) + \frac{2}{\gamma+1}c_1 \\
&u(x,t) = c + x/t \\
&p = p_1\,(c/c_1)^{2\gamma/\gamma-1} \\
&\rho = \gamma p/c^2
\end{aligned}
$$

综上所述，一维 Riemann 问题的精确解的求解思路与方程介绍完毕。本文 Sod 激波管参考精确解程序来自于 GitLab 上的项目 simple shock tube calculator [1]。

1.4 数值计算方法

1.4.1 计算域与网格

计算域默认设置为 $x \in [-5, 5]$，时间计算域为 $t \in [0, 2.0]$，该范围足以捕捉 Sod 问题中激波、接触间断和膨胀波的完整演化过程。空间离散采用均匀网格划分，网格间距 Δx 由计算域长度和网格数动态确定。时间步长 Δt 根据 CFL 条件自适应调整：

$$
\Delta t = \mathrm{CFL} \cdot \frac{\Delta x}{\max(|u| + c)} \tag{8}
$$

边界条件采用无反射处理：

$$
\begin{aligned}
U_0 &= U_1 \\
U_{N+1} &= U_N
\end{aligned}
$$

此边界处理可有效抑制数值反射，确保波系在计算域内自由传播。网格收敛性研究表明，接触间断分辨率对网格依赖性显著，需足够细密的网格才能准确捕捉密度突变特征。

1.4.2 激波捕捉格式

TVD 格式（Minmod 限制器） 总变差定义：

$$
TV(U^n) = \sum_{i=-\infty}^{\infty} |U_{i+1}^n - U_i^n| \tag{9}
$$

TVD 条件：

$$
TV(U^{n+1}) \leq TV(U^n) \tag{10}
$$

通量限制器格式：

$$
F_{i+\frac{1}{2}} = F_i + \frac{1}{2}\phi(r_i)(F_{i+1} - F_i) \tag{11}
$$

梯度比：

$$
r_i = \frac{U_i - U_{i-1}}{U_{i+1} - U_i} \tag{12}
$$

Minmod 限制器函数：

$$
\phi(r) = \mathrm{minmod}(1, r) = \begin{cases} 0 & r \leq 0 \\ r & 0 < r < 1 \\ 1 & r \geq 1 \end{cases} \tag{13}
$$

性质：

- 满足二阶精度条件：$\phi(1) = 1$
- 满足 TVD 条件：$0 \leq \phi(r) \leq \min(2, 2r)$
- 在间断处退化为单调的一阶格式

群速度控制（GVC）格式 通量修正方程：

$$\mathbf{F}^{GVC} = \mathbf{F}^{base} + \epsilon \Delta x^3 \frac{\partial^3 \mathbf{U}}{\partial x^3} \tag{14}$$

三阶导数离散：

$$\left.\frac{\partial^3 U}{\partial x^3}\right|_i \approx \frac{-U_{i-2} + 2U_{i-1} - 2U_{i+1} + U_{i+2}}{2\Delta x^3} \tag{15}$$

自适应控制系数：

$$\epsilon = C \cdot |u \pm c|, \quad C \in [0.1, 0.5] \tag{16}$$

修正后群速度：

$$v_g^{num} = v_g^{exact} - 4\epsilon k^2 \Delta x^2 \tag{17}$$

WENO 格式 基本框架

1. 选择模板：$S_0 = \{I_{i-2}, I_{i-1}, I_i\}$, $S_1 = \{I_{i-1}, I_i, I_{i+1}\}$, $S_2 = \{I_i, I_{i+1}, I_{i+2}\}$
2. 重构多项式：

$$p_0(x) = \frac{1}{3}U_{i-2} - \frac{7}{6}U_{i-1} + \frac{11}{6}U_i \tag{18}$$

$$p_1(x) = -\frac{1}{6}U_{i-1} + \frac{5}{6}U_i + \frac{1}{3}U_{i+1} \tag{19}$$

$$p_2(x) = \frac{1}{3}U_i + \frac{5}{6}U_{i+1} - \frac{1}{6}U_{i+2} \tag{20}$$

WENO-JS 光滑指示器：

$$\beta_k = \sum_{l=1}^{2} \Delta x^{2l-1} \int_{x_{i-1/2}}^{x_{i+1/2}} \left(\frac{d^l p_k(x)}{dx^l}\right)^2 dx \tag{21}$$

$$\beta_0 = \frac{13}{12}(U_{i-2} - 2U_{i-1} + U_i)^2 + \frac{1}{4}(U_{i-2} - 4U_{i-1} + 3U_i)^2 \tag{22}$$

$$\beta_1 = \frac{13}{12}(U_{i-1} - 2U_i + U_{i+1})^2 + \frac{1}{4}(U_{i-1} - U_{i+1})^2 \tag{23}$$

$$\beta_2 = \frac{13}{12}(U_i - 2U_{i+1} + U_{i+2})^2 + \frac{1}{4}(3U_i - 4U_{i+1} + U_{i+2})^2 \tag{24}$$

非线性权重：

$$\alpha_k = \frac{d_k}{(\beta_k + \varepsilon)^p}, \quad \omega_k = \frac{\alpha_k}{\sum_{m=0}^{2} \alpha_m} \tag{25}$$

$$d_0 = 0.3, \quad d_1 = 0.6, \quad d_2 = 0.1 \tag{26}$$

$$\varepsilon = 10^{-6}, \quad p = 2 \tag{27}$$

WENO-Z

改进权重：

$$\tau_5 = |\beta_0 - \beta_2| \tag{28}$$

$$\alpha_k^Z = d_k \left(1 + \frac{\tau_5}{\beta_k + \varepsilon}\right) \tag{29}$$

$$\omega_k^Z = \frac{\alpha_k^Z}{\sum_{m=0}^{2} \alpha_m^Z} \tag{30}$$

优势：光滑区 $\omega_k \to d_k$，收敛阶 $\mathcal{O}(\Delta x^5)$

1.4.3 通量处理方法

通量矢量分裂 (FVS)　Steger-Warming 分裂

$$\mathbf{F} = \mathbf{F}^+ + \mathbf{F}^- = \mathbf{R}\ ^+\mathbf{R}^{-1}\mathbf{U} + \mathbf{R}\ ^-\mathbf{R}^{-1}\mathbf{U} \tag{31}$$

$$\lambda^\pm = \frac{\lambda \pm |\lambda|}{2} \tag{32}$$

一维欧拉方程形式：

$$\mathbf{f}^\pm = \frac{\rho}{2\gamma} \begin{bmatrix} (\gamma-1)\lambda_1^\pm + \lambda_3^\pm + \lambda_5^\pm \\ 2(\gamma-1)\lambda_1^\pm u + \lambda_3^\pm (u+c) + \lambda_5^\pm (u-c) \\ (\gamma-1)\lambda_1^\pm u^2 + \frac{1}{2}\lambda_3^\pm (u+c)^2 + \frac{1}{2}\lambda_5^\pm (u-c)^2 + \frac{3-\gamma}{2(\gamma-1)}(\lambda_3^\pm + \lambda_5^\pm)c^2 \end{bmatrix} \tag{33}$$

Van Leer 分裂

质量通量分裂：

$$(\rho u)^\pm = \pm\frac{\rho c}{4}(M \pm 1)^2, \quad |M| \le 1 \tag{34}$$

通量函数：

$$\mathbf{F}^\pm = \begin{bmatrix} (\rho u)^\pm \\ (\rho u)^\pm \left[\frac{(\gamma-1)u \pm 2c}{\gamma}\right] \\ (\rho u)^\pm \left[\frac{((\gamma-1)u \pm 2c)^2}{2(\gamma^2-1)}\right] \end{bmatrix} \tag{35}$$

AUSM 分裂

通量分解：

$$\mathbf{F} = \dot{m}\ \ + \mathbf{P}, \qquad = [1, u, H]^T \tag{36}$$

界面通量：

$$\mathbf{f}_{1/2} = \dot{m}_{1/2\ \ L/R} + \mathbf{P}_{1/2} \tag{37}$$

质量流量：

$$\dot{m}_{1/2} = c_{1/2} \begin{cases} M_{1/2}\rho_L & M_{1/2} > 0 \\ M_{1/2}\rho_R & \text{其他} \end{cases}, \quad M_{1/2} = M_L^+ + M_R^- \tag{38}$$

Lax-Friedrichs 分裂

$$\mathbf{F}^\pm = \frac{1}{2}(\mathbf{F} \pm \alpha\mathbf{U}), \quad \alpha = \max|\lambda_i| \tag{39}$$

通量差分分裂 (FDS) Roe 方法

Roe 平均：

$$\hat{\rho} = \sqrt{\rho_L \rho_R} \tag{40}$$

$$\hat{u} = \frac{\sqrt{\rho_L} u_L + \sqrt{\rho_R} u_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{41}$$

$$\hat{H} = \frac{\sqrt{\rho_L} H_L + \sqrt{\rho_R} H_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{42}$$

通量计算：

$$\mathbf{f}_{1/2} = \frac{1}{2}(\mathbf{F}_L + \mathbf{F}_R) - \frac{1}{2}\sum_{k=1}^{3} |\hat{\lambda}_k| \hat{\alpha}_k \hat{\mathbf{r}}_k \tag{43}$$

HLL 方法

通量计算：

$$\mathbf{f}_{\text{HLL}} = \begin{cases} \mathbf{f}_L & S_L \geq 0 \\ \frac{S_R \mathbf{f}_L - S_L \mathbf{f}_R + S_L S_R (\mathbf{U}_R - \mathbf{U}_L)}{S_R - S_L} & S_L < 0 < S_R \\ \mathbf{f}_R & S_R \leq 0 \end{cases} \tag{44}$$

波速估计：

$$S_L = \min(u_L - c_L, \tilde{u} - \tilde{c}), \quad S_R = \max(u_R + c_R, \tilde{u} + \tilde{c}) \tag{45}$$

Lax-Wendroff 方法

二阶中心格式：

$$\mathbf{f}_{i+1/2} = \mathbf{f}\left(\mathbf{u}_{i+1/2}^{\text{LW}}\right) \tag{46}$$

中间状态：

$$\mathbf{u}_{i+1/2}^{\text{LW}} = \frac{1}{2}(\mathbf{u}_i + \mathbf{u}_{i+1}) - \frac{\Delta t}{2\Delta x}(\mathbf{f}_{i+1} - \mathbf{f}_i) \tag{47}$$

1.4.4 时间推进格式

三阶 Runge-Kutta (SSP-RK3)：

$$\mathbf{U}^{(1)} = \mathbf{U}^n + \Delta t \mathcal{L}(\mathbf{U}^n) \tag{48}$$

$$\mathbf{U}^{(2)} = \frac{3}{4}\mathbf{U}^n + \frac{1}{4}\left[\mathbf{U}^{(1)} + \Delta t \mathcal{L}(\mathbf{U}^{(1)})\right] \tag{49}$$

$$\mathbf{U}^{n+1} = \frac{1}{3}\mathbf{U}^n + \frac{2}{3}\left[\mathbf{U}^{(2)} + \Delta t \mathcal{L}(\mathbf{U}^{(2)})\right] \tag{50}$$

稳定性条件：

$$\text{CFL} = \max(|u| + c)\frac{\Delta t}{\Delta x} \leq 1.5 \tag{51}$$

符号说明

符号	物理意义
ρ	密度
u	速度
p	压强
E	总能
c	声速 $c = \sqrt{\gamma p/\rho}$
H	总焓 $H = (E + p)/\rho$
γ	比热比
$\mathbf{R}, \mathbf{L}$	右/左特征向量矩阵
λ	特征值
Δx	空间步长
Δt	时间步长

2 代码生成与调试

由于本次的代码量大，且要求复杂，所以编写代码的过程中严格遵循模块化思路，在编写之初就按照功能划分了多个模块，分别为：

- **main.py**：主程序入口，负责整体流程控制
- **config.py**：配置模块，存储参数设置和初始条件
- **time_integration.py**：时间积分模块，包含三阶 Runge–Kutta 方法实现
- **utils**：工具函数库，包含数据处理、可视化、调用精确解、边界条件处理等功能
- **initialization**: 初始条件模块，负责设置初始状态和网格划分
- **flux**: 通量计算模块，包含 FVS、FDS 等格式实现
- **schemes**: 数值格式模块，包含 TVD、WENO 等格式实现

对于每个模块，在编写完成后都进行了单元测试，确保各个模块功能正确。主程序通过调用这些模块实现整体流程控制。测试所用的程序在 **test** 目录下，包含了对各个模块的单元测试脚本。接下来我将按照编写的顺序对各个模块进行介绍。

2.1 initialization

在初始化模块下，我一共编写两个文件: 一个是 **domain_setup.py**，用于设置计算域和网格划分；另一个是 **sod_initial.py**，用于设置初始条件。

在 test 目录中的 **initialization_test.py** 中，我对这两个模块进行了单元测试，确保它们能够正确设置计算域和初始条件。

2.2 utils

在工具函数模块下，我编写了 **boundary.py**、**exact_solution.py**、**visualization.py** 三个文件，同时还将来自 GitLab[1] 的代码 **gitlab_sod_analytical.py** 也放置于此模块下。其中 **boundary.py** 实现了三种边界条件处理，**exact_solution.py** 在调用 GitLab 上的代码的基础上实现了 Sod 激波管问题的精确解计算，**visualization.py** 实现了结果可视化功能。

在 test 目录中的 **boundary_test.py**、**exact_solution_test.py** 和 **visualization_test.py** 中，我对这些工具函数进行了单元测试，确保它们能够正确处理边界条件、计算精确解和可视化结果。

2.3 flux

在通量计算模块下，我编写了 **flux_fvs.py** 和 **flux_fds.py** 两个文件，分别实现了 FVS 和 FDS 格式的通量计算。其中 **flux_fvs.py** 实现了 Steger-Warming、Van Leer、AUSM 和 Lax-Friedrichs 四种通量分裂方法，**flux_fds.py** 实现了 HLL、Roe 和 Lax-Wendroff 三种通量差分分裂方法。但是其中的 Roe 方法存在问题，我在尝试多次后仍然未能解决 bug，所以在择时请勿使用 Roe 方法。

2.4 schemes

在数值格式模块下，我编写了 **tvd.py**、**weno.py** 和 **gvc.py** 三个文件，分别实现了 TVD 格式、WENO 格式和群速度控制格式。其中 **tvd.py** 实现了 Minmod 限制器的 TVD 格式，**weno.py** 实现了 WENO-JS 和 WENO-Z 格式，**gvc.py** 实现了群速度控制格式。

在 test 目录中的 **tvd_fvs_rk3_test.py** 中，我对于 TVD、FVS 和 RK3 格式进行了单元测试，确保在这一组合下能够正确计算 Sod 激波管问题的数值解。

在 test 目录中的 **weno_test.py** 中，我对 WENO 格式进行了单元测试，确保其能够正确处理 Sod 激波管问题的数值解。

在 test 目录中的 **gvc_test.py** 中，我对群速度控制格式进行了单元测试，确保其能够正确处理 Sod 激波管问题的数值解。

在 test 目录中的 **fds_test.py** 中，我在使用 TVD 格式的基础上，测试了 FDS 格式的通量计算，确保其能够正确处理 Sod 激波管问题的数值解。

2.5 time_integration.py

主目录下，我编写了 **time_integration.py**，用于实现三阶 Runge-Kutta 时间积分方法和计算 Δt。

2.6 config.py

在主目录下，我编写了 **config.py**，用于存储参数设置和初始条件。该文件包含了所有需要的参数，如计算域、网格划分、初始条件等。此外，在这个文件中，用户可以自主选择使用的数值格式和通量计算方法。这样可以方便地进行参数调整和格式切换，而无需修改其他代码。

2.7 main.py

作为项目的主程序入口，**main.py** 负责整体流程控制。它首先导入配置文件中的参数设置，然后调用各个模块实现 Sod 激波管问题的数值计算和可视化。

2.8 使用方法

在项目根目录下，用户可以通过以下命令运行主程序：

```
python main.py
```

运行后，程序将自动执行以下步骤：

1. 导入配置文件中的参数设置
2. 调用初始化模块设置计算域和初始条件
3. 调用工具函数模块处理边界条件和计算精确解
4. 调用通量计算模块计算通量
5. 调用数值格式模块进行数值计算
6. 调用时间积分模块进行时间推进
7. 可视化结果并保存图像文件到 **result** 目录下

```python
# 计算域参数
self.nx = 200
self.x_min = -5.0
self.x_max = 5.0
self.t_end = 1.0

# 物理参数
self.gamma = 1.4
self.cv = 1.0

# 数值方法选择
self.scheme = 'tvd'        # 激波捕捉格式: 'tvd', 'gvc', 'weno'
self.flux_method = 'fvs'   # 通量处理方法: 'fvs', 'fds'
self.flux_type = 'van_leer'
                           # 具体通量类型:
                           # FVS: 'steger_warming', 'van_leer', 'ausm', 'lax_friedrichs'
                           # FDS: 'hll', 'lax_wendroff', 'roe（存在bug）'
self.limiter = 'minmod'    # TVD限制器类型
self.weno_variant = 'z'    # WENO变体: 'js', 'z'

# 边界条件
self.bc_type = 'non-reflective'  # 'non-reflective', 'periodic', 'fixed'
self.num_ghost = 3               # 虚单元层数
```

图 3: config.py 参数设置示例

运行完成后，用户可以在 **result** 目录下找到生成的图像文件，这些图像展示了 Sod 激波管问题的数值解和精确解的对比。如果需要修改参数设置或选择不同的数值格式和通量计算方法，只需编辑 **config.py** 文件即可。

2.9 代码提交信息

在项目开发过程中，我使用了 Git 进行版本控制。以下是一些关键的提交信息

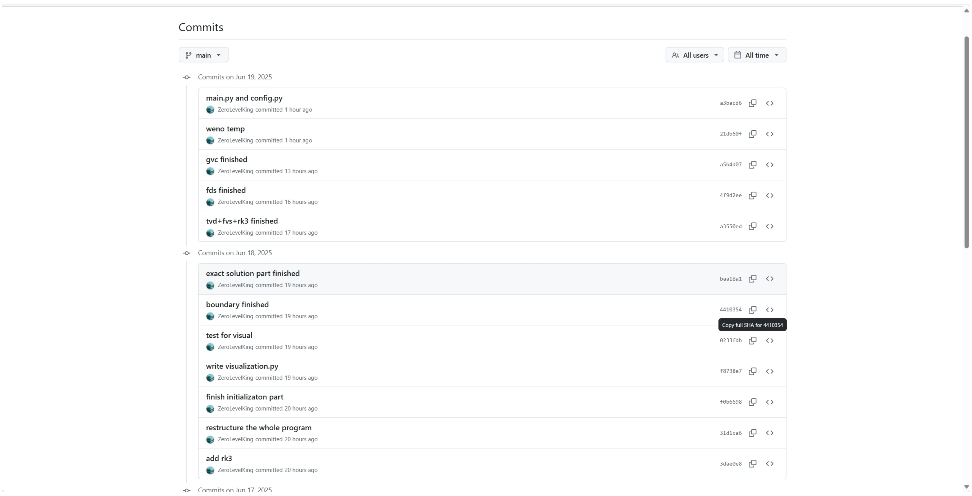

图 4: Git 提交信息示例 1

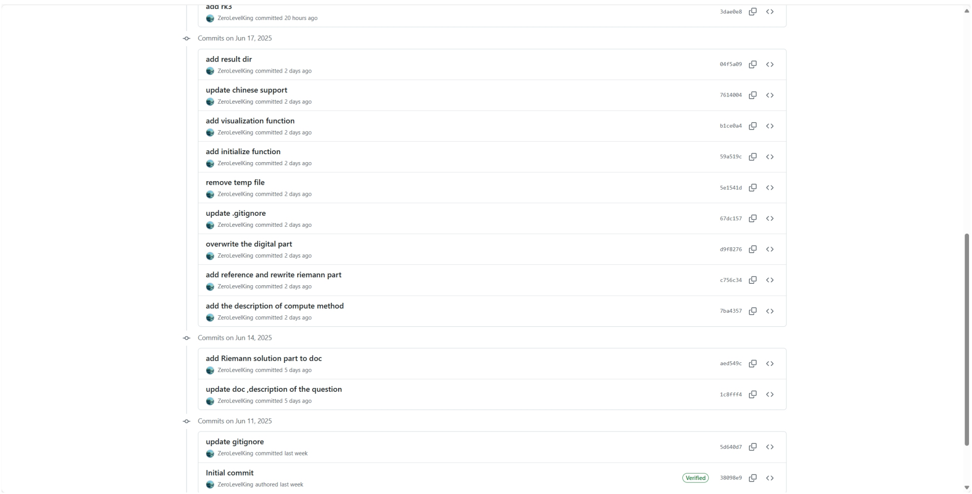

图 5: Git 提交信息示例 2

AI 工具使用声明表

使用内容	工具名称	使用目的
hw2.tex 1-9 行、图片插入	Github Copilot	调整 pdf 格式，调用宏包，省略插入图片的重复性工作
main.py 6-15 行	DeepSeek	修正 matplotlib 中文显示问题
ReadMe.md 框架	DeepSeek	在 DeepSeek 的帮助下生成一个框架，在此基础上增加而来
.gitignore	Github Copilot	针对于 python 和 latex 的.gitignore 文件，完全由 Copilot 生成

参考文献

[1] fantaz. Simple shock tube calculator, 2021. Accessed: 2025-06-17. URL: https://gitlab.com/fantaz/simple_shock_tube_calculator.