双曲守恒律问题上机习题

我们将对给定的Burgers' 方程进行数值求解，并比较使用不同数值格式的结果。我们将考虑以下数值格式：

1. Godunov 格式
2. Lax-Wendroff 格式
3. 二阶 TVD 格式

问题描述

考虑 Burgers' 方程的初值问题：

$$
\begin{cases} \dfrac{\partial u}{\partial t} + \dfrac{\partial}{\partial x}\left(\dfrac{u^2}{2}\right) = 0 \\ u(x,0) = u_0(x) \end{cases} \tag{1}
$$

初值为：

$$
u_0(x) = \begin{cases} 1 & x \in [0.4, 0.6] \\ 0 & x \notin [0.4, 0.6] \end{cases} \tag{2}
$$

数值格式实现

Godunov 格式

Godunov 格式是基于求解局部黎曼问题来更新数值解的一个一阶格式。对于方程 $\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0$，其中 $f(u) = \frac{u^2}{2}$，Godunov 格式为：

$$
u_i^{n+1} = u_i^n - \frac{\Delta t}{\Delta x}\left(F_{i+1/2}^n - F_{i-1/2}^n\right) \tag{3}
$$

其中，$F_{i+1/2}^n$ 是黎曼问题在界面 $x_{i+1/2}$ 处的数值通量。

Lax-Wendroff 格式

Lax-Wendroff 格式是一种二阶格式，其通量计算为：

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{\Delta x}\left(f_{i+1/2} - f_{i-1/2}\right) + \frac{(\Delta t)^2}{2(\Delta x)^2}\left(\left(f'_{i+1/2}\right)^2 - \left(f'_{i-1/2}\right)^2\right) \tag{4}$$

其中，$f_{i+1/2}$ 和 $f_{i-1/2}$ 是通量，$f'_{i+1/2}$ 和 $f'_{i-1/2}$ 是通量的导数。

二阶 TVD 格式

我们将使用一种典型的二阶 TVD 格式，通常是 MUSCL-Hancock 格式。此格式结合了 MUSCL (Monotonic Upstream-centered Scheme for Conservation Laws) 和 Hancock 时间推进步骤。

首先，我们需要一个斜率限制器（如minmod限制器）来计算每个单元的斜率：

$$\Delta u_i = \text{minmod}\left(\frac{u_i - u_{i-1}}{\Delta x}, \frac{u_{i+1} - u_i}{\Delta x}\right) \tag{5}$$

然后，我们构建界面值：

$$u_{i+1/2}^L = u_i + \frac{1}{2}\Delta u_i \tag{6}$$

$$u_{i+1/2}^R = u_{i+1} - \frac{1}{2}\Delta u_{i+1} \tag{7}$$

最后，我们通过黎曼求解器计算通量并更新解。

计算和比较

我们将在以下Python脚本中实现这三种格式，并在相同初值条件下进行计算和比较结果。

```python
import numpy as np
import matplotlib.pyplot as plt

# 定义初始条件
def initial_condition(x):
    return np.where((x >= 0.4) & (x <= 0.6), 1, 0)
```

```python
# 定义 Godunov 格式
def godunov_flux(u_left, u_right):
    if u_left > u_right:
        if u_left <= 0:
            return 0.5 * u_left**2
        elif u_right >= 0:
            return 0.5 * u_right**2
        else:
            return 0
    else:
        if u_left >= 0:
            return 0.5 * u_left**2
        elif u_right <= 0:
            return 0.5 * u_right**2
        else:
            return 0

def godunov_step(u, dx, dt):
    flux = np.zeros_like(u)
    for i in range(1, len(u)):
        flux[i] = godunov_flux(u[i-1], u[i])
    return u - dt/dx * (flux[1:] - flux[:-1])

# 定义 Lax-Wendroff 格式
def lax_wendroff_step(u, dx, dt):
    f = 0.5 * u**2
    u_next = np.zeros_like(u)
    for i in range(1, len(u)-1):
        u_next[i] = u[i] - 0.5 * dt/dx * (f[i+1] - f[i-1]) + 0.5 *
(dt/dx)**2 * (
            (u[i+1] + u[i]) * (f[i+1] - f[i]) - (u[i] + u[i-1]) *
(f[i] - f[i-1])
        )
    return u_next

# 定义 TVD 格式
def minmod(a, b):
    if a * b <= 0:
```

```
        return 0
    else:
        return min(abs(a), abs(b)) * np.sign(a)

def tvd_step(u, dx, dt):
    u_next = np.zeros_like(u)
    flux = np.zeros_like(u)

    # 计算斜率
    slope = np.zeros_like(u)
    for i in range(1, len(u)-1):
        slope[i] = minmod((u[i] - u[i-1]) / dx, (u[i+1] - u[i]) /
dx)

    # 计算界面值
    u_L = u + 0.5 * slope * dx
    u_R = u - 0.5 * slope * dx

    # 计算通量
    for i in range(1, len(u)):
        flux[i] = godunov_flux(u_L[i-1], u_R[i])

    u_next[1:-1] = u[1:-1] - dt/dx * (flux[1:-1] - flux[:-2])
    return u_next

# 设置网格和时间步长
x = np.linspace(0, 1, 201)
dx = x[1] - x[0]
dt = 0.5 * dx
t_final = 0.2

# 初始条件
u0 = initial_condition(x)

# Godunov 格式计算
u_godunov = u0.copy()
t = 0
while t < t_final:
    u_godunov = godunov_step(u_godunov, dx, dt)
```

```python
    t += dt

# Lax-Wendroff 格式计算
u_lax_wendroff = u0.copy()
t = 0
while t < t_final:
    u_lax_wendroff = lax_wendroff_step(u_lax_wendroff, dx, dt)
    t += dt

# TVD 格式计算
u_tvd = u0.copy()
t = 0
while t < t_final:
    u_tvd = tvd_step(u_tvd, dx, dt)
    t += dt

# 绘图比较结果
plt.figure(figsize=(10, 6))
plt.plot(x, u0, label='Initial Condition', linestyle='--')
plt.plot(x, u_godunov, label='Godunov')
plt.plot(x, u_lax_wendroff, label='Lax-Wendroff')
plt.plot(x, u_tvd, label='TVD')
plt.xlabel('x')
plt.ylabel('u')
plt.title('Comparison of Different Numerical Schemes')
plt.legend()
plt.grid()
plt.show()
```

运行该脚本后，我们将得到三个数值格式的结果并进行比较。Godunov 格式是一阶格式，具有较大的数值耗散；Lax-Wendroff 格式是二阶格式，具有较小的数值耗散，但可能会产生非物理振荡；TVD 格式结合了高阶精度和总变差减小性质，可以在保持精度的同时避免非物理振荡。