CFD 上机报告 IV

陈伟
1901110037

2020 年 6 月 19 日

目录

1 问题描述

1. 参考 mit18086 navierstokes.m 和 mit18086 navierstokes.pdf，可以修改该程序，计算双剪切层问题 [CFDLect08-incom cn.pdf 的第 122 页] 和计算后台阶层流问题 [见 bkwd step.pdf 或 http://homepages.rpi.edu/ bhatts8/bkwd.html].

Stokes 方程组:

$$
\begin{aligned}
&u_t + p_x = -\left(u^2\right)_x - (uv)_y + \frac{1}{Re}\left(u_{xx} + u_{yy}\right) \\
&v_t + p_y = -(uv)_x - \left(v^2\right)_y + \frac{1}{Re}\left(v_{xx} + v_{yy}\right) \\
&u_x + v_y = 0
\end{aligned} \tag{1}
$$

1.1 Example1. 双剪切层问题

初始条件, $0 \leqslant x \leqslant 1, 0 \leqslant y \leqslant 1$,

$$
u(\boldsymbol{x}, 0) = \begin{cases} \tanh[(y-0.25)/\rho], & y \leq 0.5 \\ \tanh[(0.75-y)/\rho], & y > 0.5 \end{cases} \qquad v(\boldsymbol{x}, 0) = \delta \sin(2\pi x)
$$

情况 1: $\rho = 1/30, \delta = 0.05, \nu = 10^{-4}$;

情况 2: $\rho = 1/100, \delta = 0.05, \nu = 1/20000$.

周期边界条件，网格 256×256, 512×512, $T = 0.8, 1.8$. 画涡等值线.

1.2 Example2. 后台阶层流问题

$Re = 50$, 初始条件:

$$U(x,y) = \begin{cases} \begin{cases} 0, & 0 \leqslant y \leqslant 1 \\ 1.5[1-4(y-1.5)^2], & 1 \leqslant y \leqslant 2 \end{cases} \\ \\ \begin{cases} 1.5(1-x/40)(2-y)(2y-1), & 1.5(1-x/40)(2-y)(2y-1) \geqslant 0 \\ 0, & 1.5(1-x/40)(2-y)(2y-1) < 0 \end{cases} \end{cases}$$

$V(x,y) = 0.$

边界条件:

Walls: U = V = 0; $\frac{\partial P}{\partial n} = 0$

Inlet: $\mathrm{U}(0,y,t) = \begin{cases} 0, & 0 \leqslant y \leqslant 1 \\ 1.5[1-4(y-1.5)^2], & 1 \leqslant y \leqslant 2 \end{cases}$, V = 0

Exit: $\frac{\partial U}{\partial x} = 0, \frac{\partial V}{\partial x} = 0$

2 算法

对于不可压 NS 方程(**??**), 通过如下步骤更新求解:

非线性项 : 显示更新非线性项, 如下:

$$\frac{U^* - U^n}{\Delta t} = -((U^n)^2)_x - (U^n V^n)_y$$
$$\frac{V^* - V^n}{\Delta t} = -((U^n V^n))_x - ((V^n)^2)_y$$

更新速度 : 显示隐式更新速度, 如下:

$$\frac{U^{**} - U^*}{\Delta t} = \frac{1}{Re}(U^{**}_{xx} + U^{**}_{yy})$$
$$\frac{V^{**} - V^*}{\Delta t} = \frac{1}{Re}(V^{**}_{xx} + V^{**}_{yy})$$

压力矫正 ：通过 (U^{**}, V^{**}) 矫正压力项如下:

$$\frac{U^{**}-U^{n}}{\Delta t}=-(P^{n+1})_x$$
$$\frac{V^{**}-V^{n}}{\Delta t}=-(P^{n+1})_y$$

向量上来看也即是:

$$\frac{\boldsymbol{U}^{n+1}-\boldsymbol{U}^{**}}{\Delta t}=-\nabla P$$

两边同时作用散度算子，再用不可压条件有:

$$-\Delta P^{n+1}=-\frac{1}{\Delta t}\nabla\cdot\boldsymbol{U}^{**}$$

这样我们可以如下更新:

(a) Compute $F^n=\nabla\cdot\mathbf{U}^{**}$

(b) Solve Poisson equation $-\Delta P^{n+1}=-\frac{1}{\Delta t}F^n$

(c) Compute $\mathbf{G}^{n+1}=\nabla P^{n+1}$

(d) Update velocity field $\mathbf{U}^{n+1}=\mathbf{U}^{**}-\Delta t\mathbf{G}^{n+1}$

对于区域 $[0,L_x]\times[0,L_y]$, 分别 x,y 方向剖分 n_x,n_y 份,h_x,h_y 表示网格尺寸,$x_i=ih_x, x_{i\pm1/2}=(i\pm1/2)h_x$, 对 y 方向同理。在采用交错网格下我们分别来讨论三个步骤具体做法，其中

$$U_{i,j\pm1/2}\approx u(x_i,y_{j\pm1/2})$$
$$V_{i\pm1/2,j}\approx v(x_{i\pm1/2},y_j)$$
$$P_{i\pm1/2,j\pm1/2}\approx p(x_{i\pm1/2},y_{j\pm1/2})$$

这样我们的点的个数有:

field quantity	interior resolution	resolution with boundary points
pressure P	$n_x\times n_y$	$(n_x+2)\times(n_y+2)$
velocity component U	$(n_x-1)\times n_y$	$(n_x+1)\times(n_y+2)$
velocity component V	$n_x\times(n_y-1)$	$(n_x+2)\times(n_y+1)$

2.1 更新非线性项

为了得到 $((U^n)^2)_x, (U^nV^n)_y$ 在 $(x_i, y_{j-1/2})$ 处的逼近, 我们需要 $U^n_{i\pm1/2,j-1/2}$ 与 $U^n_{i,j}, V^n_{i,j}, U^n_{i,j-1}, V^n_{i,j-1}$ 处的值, 为此用:

$$
\begin{aligned}
U^n_{i-1/2,j-1/2} &\approx \frac{U^n_{i-1,j-1/2} + U^n_{i,j-1/2}}{2} \\
U^n_{i+1/2,j-1/2} &\approx \frac{U^n_{i,j-1/2} + U^n_{i+1,j-1/2}}{2} \\
U^n_{i,j-1} &\approx \frac{U^n_{i,j-1/2} + U^n_{i,j-3/2}}{2} \\
U^n_{i,j} &\approx \frac{U^n_{i,j+1/2} + U^n_{i,j-1/2}}{2} \\
V^n_{i,j-1} &\approx \frac{V^n_{i-1/2,j-1} + V^n_{i+1/2,j-1}}{2} \\
V^n_{i,j} &\approx \frac{V^n_{i-1/2,j} + V^n_{i+1/2,j}}{2}
\end{aligned}
$$

对于内点 $1 \leqslant i \leqslant n_x - 1, 1 \leqslant j \leqslant n_y$, 关于 $U^n_{i\pm1/2,j-1/2}$, 我们会用到左右边界值:

$$U^n_{0,j-1/2}, U_{n_x,j-1/2},\ j = 1, 2, \cdots, n_y$$

关于 $U^n_{i,j}, U^n_{i,j-1}$, 我们会用到上下边界值:

$$U^n_{i,-1/2}, U^n_{i,n_y+1/2},\ i = 1, 2, \cdots, n_x - 1$$

关于 $V^n_{i,j-1}, V^n_{i,j}$, 我们会用到上下边界值:

$$V^n_{i,0}, V^n_{i,n_y},\ i = 1, 2, \cdots, n_x - 1$$

我们记上面方式得到的值分别为 $\bar{U}^n$(对 U^n 沿 x 方向取平均), $\bar{U}^v$,(对 U^n 沿 y 方向取平均), $\bar{V}^h$(对 V^n 沿 y 方向取平均.) 同样的讨论对 $(U^nV^n)_x, ((V^n)^2)_y$ 适用. 再记 $\bar{V}^v$ 表示 V^n 沿 x 方向取平均. 这样我们有:

$$
\begin{aligned}
\frac{U^* - U^n}{\Delta t} &= -((\bar{U}^h)^2)_x - (\bar{U}^v\bar{V}^h)_y \\
\frac{V^* - V^n}{\Delta t} &= -(\bar{U}^v\bar{V}^h)_x - ((\bar{V}^v)^2)_y
\end{aligned}
$$

为了得到稳定格式，引入 $\gamma \in [0, 1]$ 参数，

$$\gamma = \min\left(1.2 \cdot \Delta t \cdot \max\left(\max_{i,j}\left|U_{i,j-1/2}\right|, \max_{i,j}\left|V_{i-1/2,j}\right|\right), 1\right)$$

以及

$$(\bar{U}^h)_{i-1/2,j-1/2} = \frac{U_{i,j-1/2} - U_{i-1,j-1/2}}{2}$$
$$(\bar{U}^v)_{i,j} = \frac{U_{i,j+1/2} + U_{i,j-1/2}}{2}$$

同样对 V^n 得到对应的值, 这样更新格式也就是:

$$\begin{aligned}
\tfrac{U^*-U}{\Delta t} &= -\left(\left(\bar{U}^h\right)^2 - \gamma\left|\bar{U}^h\right|\tilde{U}^h\right)_x - \left(\left(\bar{U}^v\bar{V}^h\right) - \gamma\left|\bar{V}^h\right|\tilde{U}^v\right)_y \\
\tfrac{V^*-V}{\Delta t} &= -\left(\left(\bar{U}^v\bar{V}^h\right) - \gamma\left|\bar{U}^v\right|\tilde{V}^h\right)_x - \left(\left(\bar{V}^v\right)^2 - \gamma\left|\bar{V}^v\right|\tilde{V}^v\right)_y
\end{aligned}$$

2.2 隐式更新速度

我们有:

$$U^{**}_{i,j-1/2} - \frac{\Delta t}{Re}\left(\frac{U^{**}_{i+1,j-1/2} - 2U^{**}_{i,j-1/2} + U^{**}_{i-1,j-1/2}}{h_x^2} + \frac{U^{**}_{i,j+1/2} - 2U^{**}_{i,j-1/2} + U^{**}_{i,j-3/2}}{h_y^2}\right) = U^*_{i,j-1/2}$$
$$V^{**}_{i-1/2,j} - \frac{\Delta t}{Re}\left(\frac{U^{**}_{i+1/2,j} - 2U^{**}_{i-1/2,j} + U^{**}_{i-3/2,j}}{h_x^2} + \frac{U^{**}_{i-1/2,j+1} - 2U^{**}_{i-1/2,j} + U^{**}_{i-1/2,j-1}}{h_y^2}\right) = V^*_{i-1/2,j}$$

特别对于边界项，由不同的边界项会影响到矩阵的对角元的第一个和最后一个值，如对于 U 是以矩阵形式存储时，

$$K_1 = \frac{1}{h^2}\begin{pmatrix} a_{11} & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & a_{11} \end{pmatrix}$$

则 $\Delta u = K_1 U + U K_1$, 其中

a_{11}	type of boundary condition
1	Neumann
2	Dirichlet for point on the boundary
3	Dirichlet for boundary between two points

更具体我们留到具体例子讨论.

2.3 压力矫正

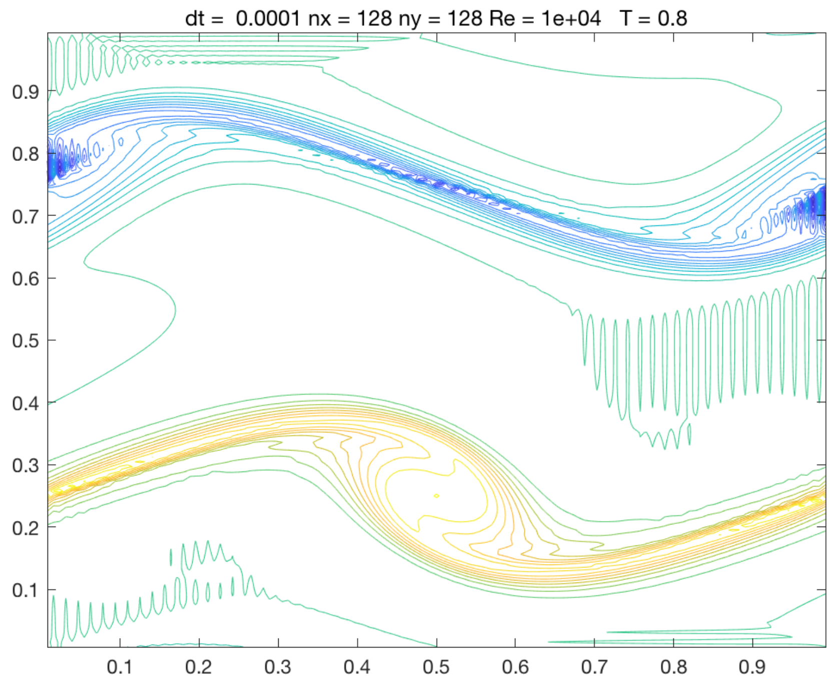
同样我们先解方程:

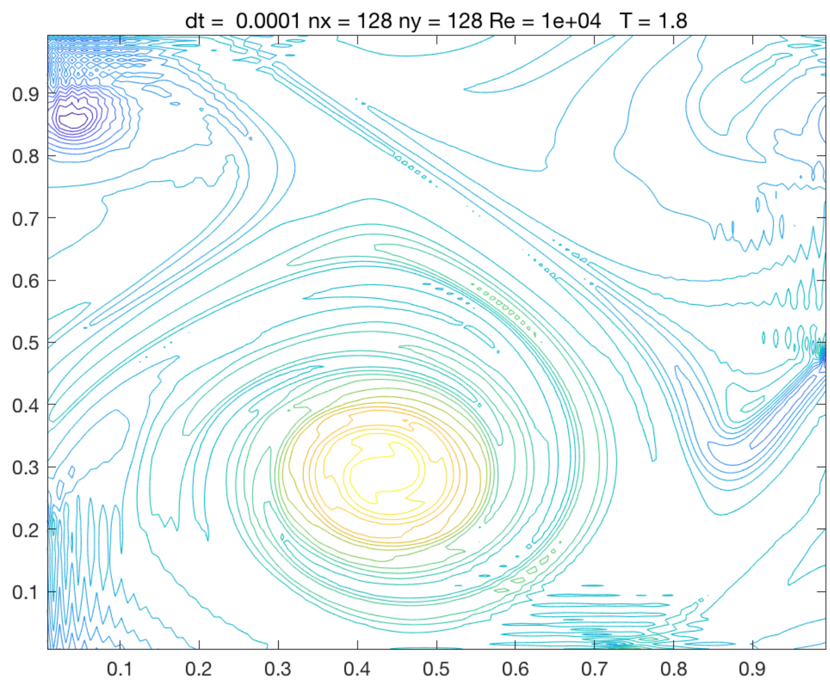
$$-\Delta P^{n+1} = -\frac{1}{\Delta t} \cdot U^{**}$$

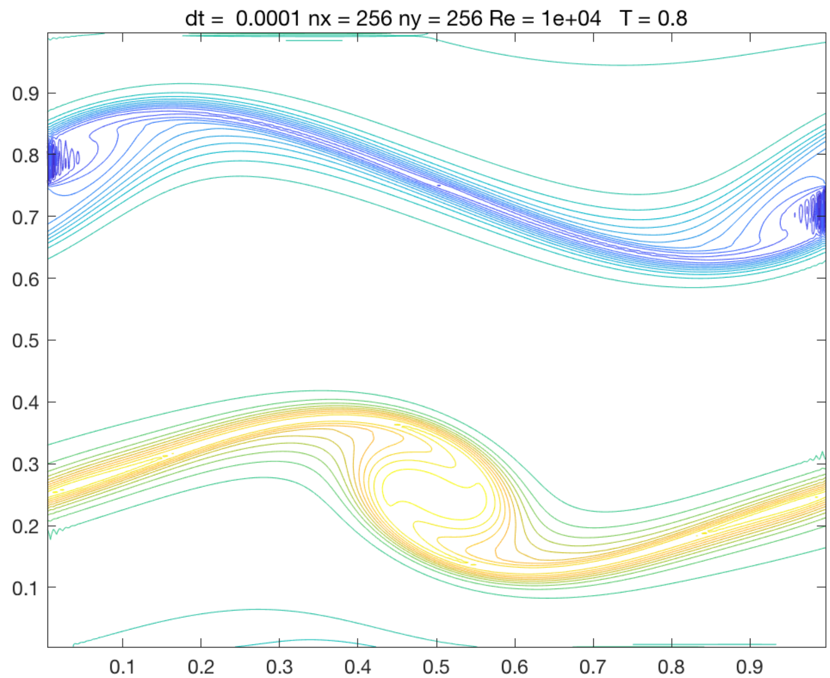
对于 P 为 Neumann 边界时, 有 $N(L_p) = \mathrm{span}(\boldsymbol{e})$, 其中 $\boldsymbol{e} = (1, 1, \cdots, 1)^T$，处理方式是在 $L_p(1,1)$ 加 1 或者 $L_p(\mathrm{end}, \mathrm{end})$ 加 1 即可。具体参见 [**?**]. 最后再用 ∇P^{n+1} 更新得到 U^{n+1} 即可。

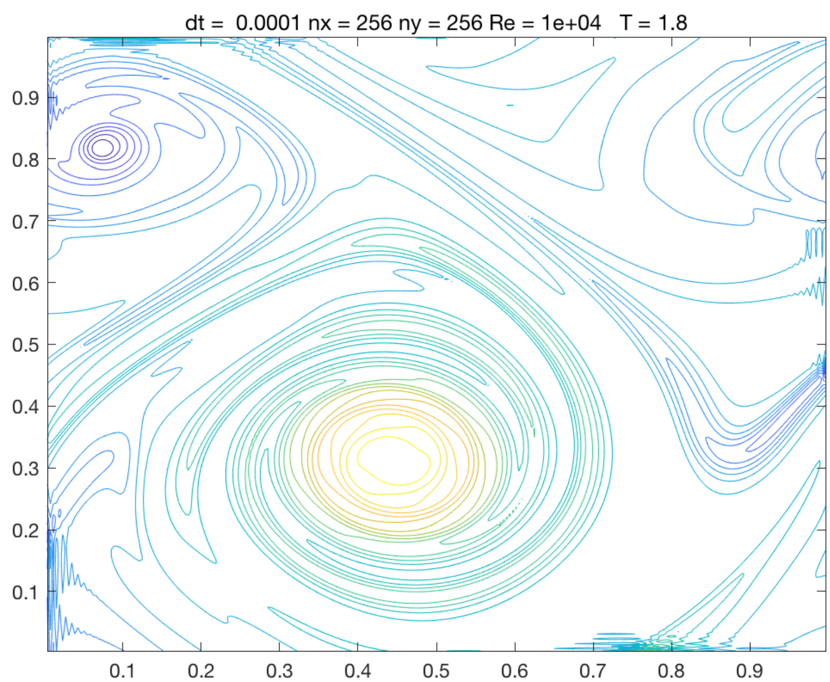
3 双剪切层问题

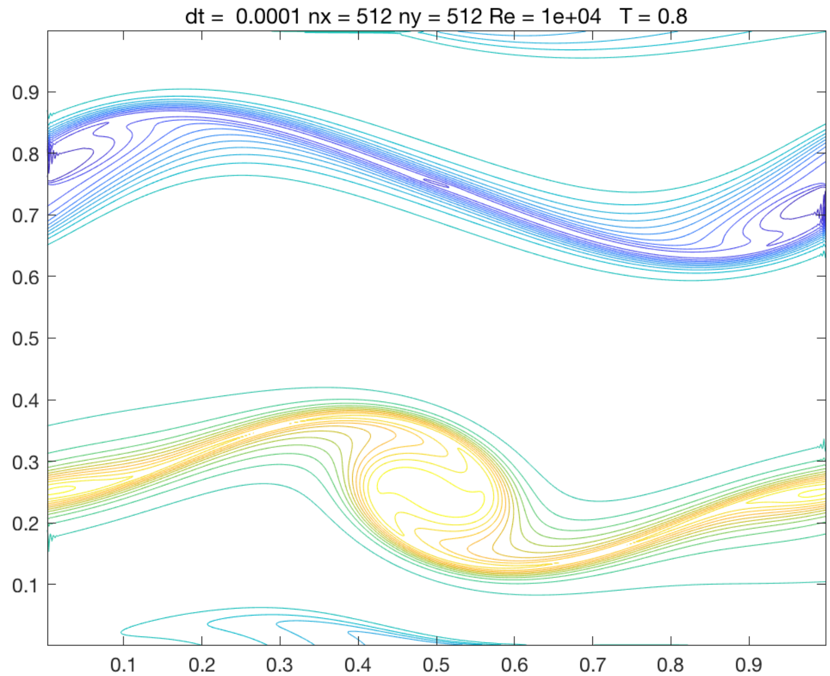
由于采用的是周期边界，所以关于 u, v 的边界层 $uE, uW, uS, uN, vE, vW, vS, vN$，我们可以用内点平均给出, 如 u 的左边界，我们可以用 u 在内部 x 方向的第一层和最后一层取平均给出。这样对于情况一，结果如下.

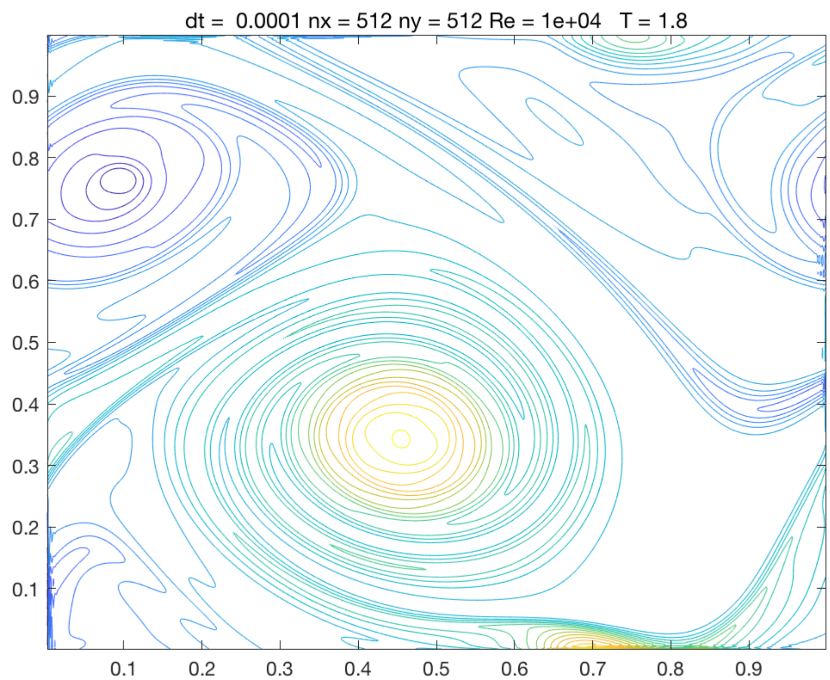

dt = 0.0001 nx = 128 ny = 128 Re = 1e+04 T = 0.8

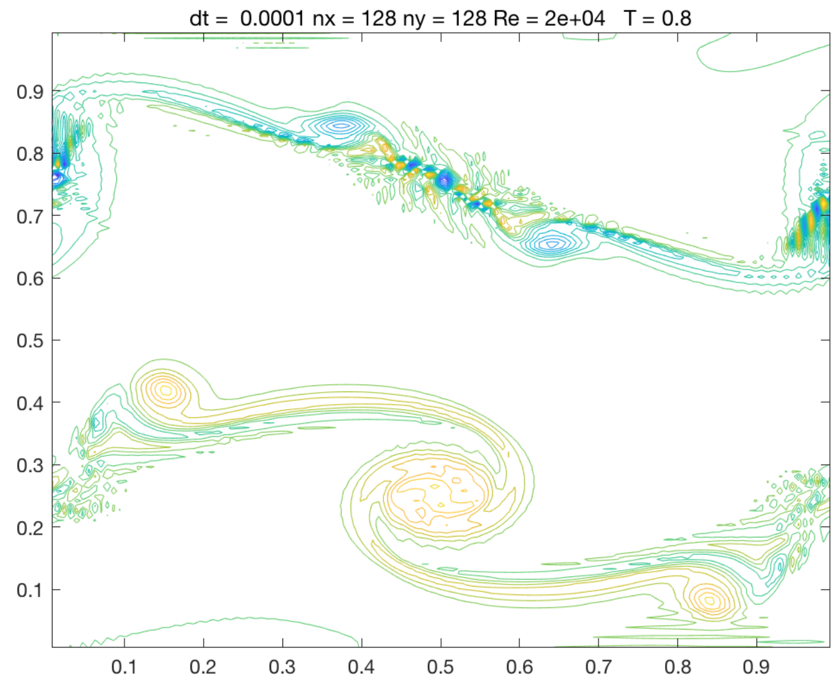

dt = 0.0001 nx = 128 ny = 128 Re = 1e+04 T = 1.8

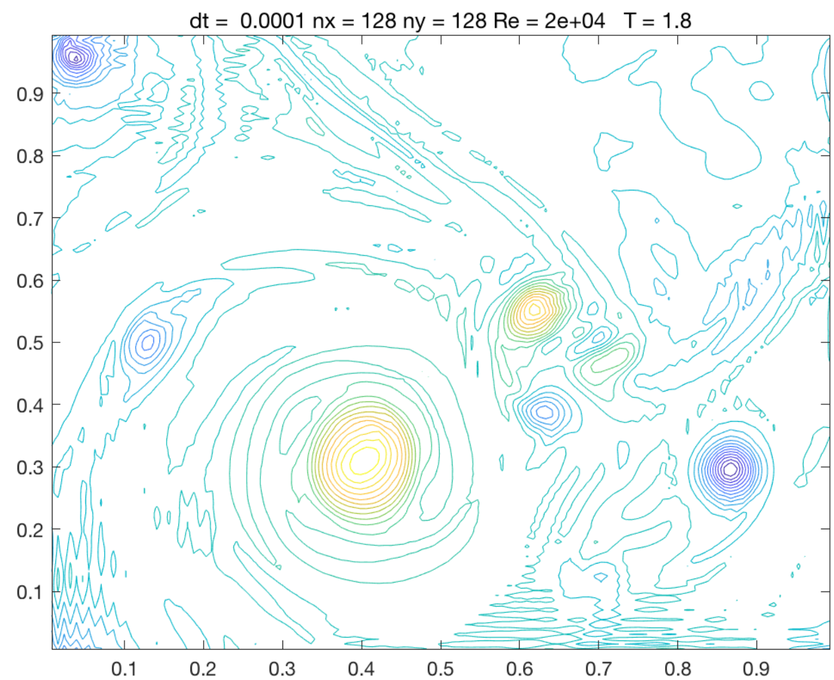

dt = 0.0001 nx = 256 ny = 256 Re = 1e+04 T = 0.8
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9

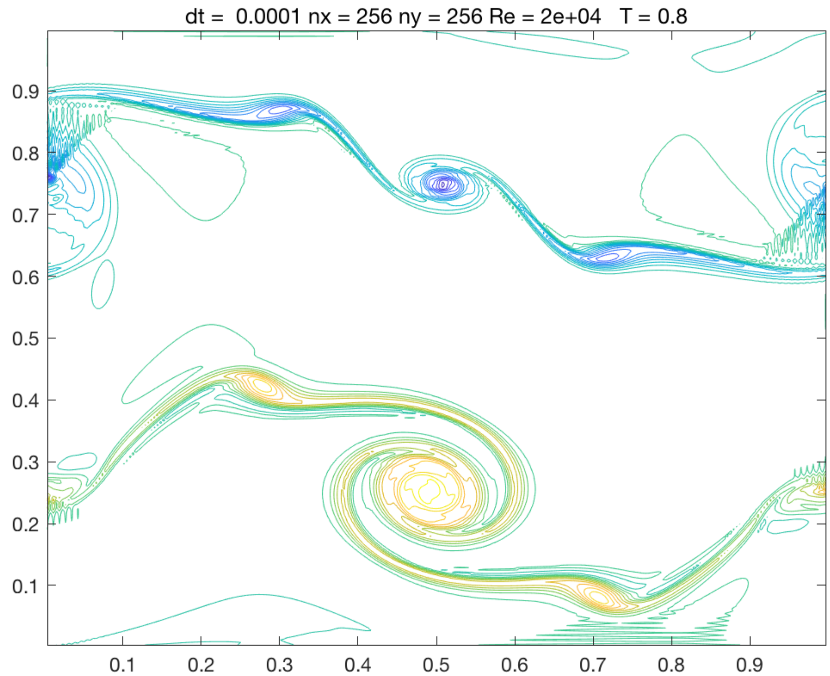

dt = 0.0001 nx = 256 ny = 256 Re = 1e+04 T = 1.8
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9

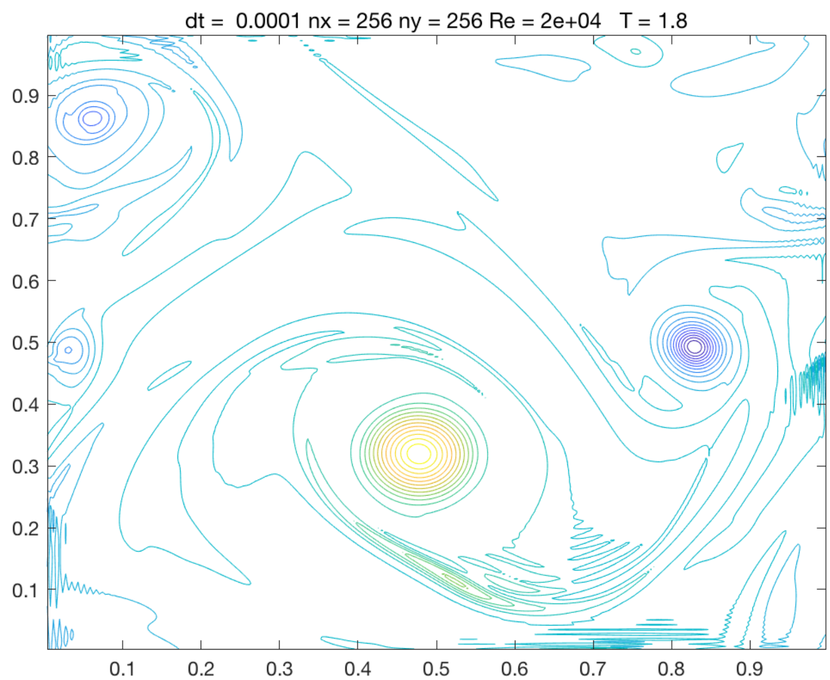

dt = 0.0001 nx = 512 ny = 512 Re = 1e+04 T = 0.8
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9

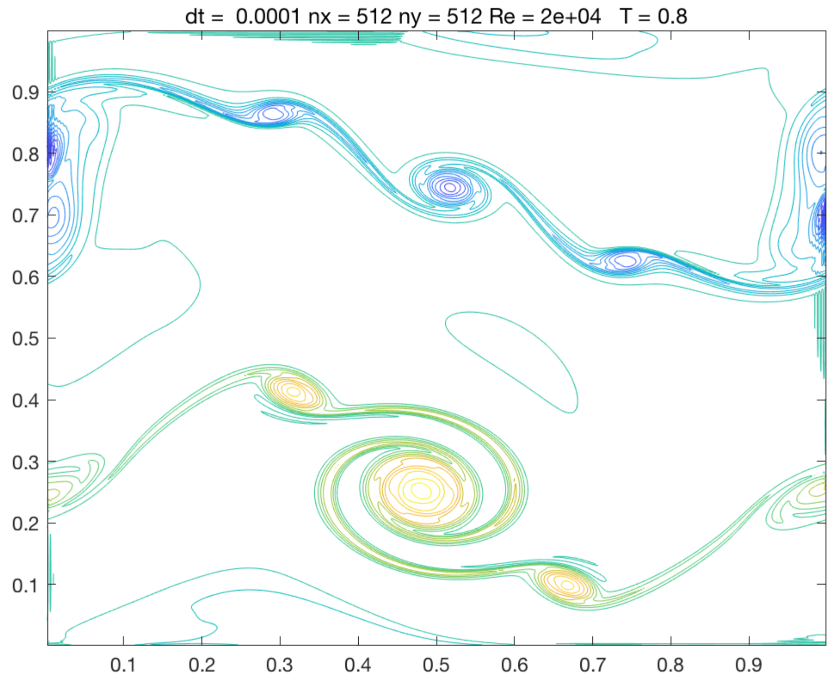

dt = 0.0001 nx = 512 ny = 512 Re = 1e+04 T = 1.8
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9

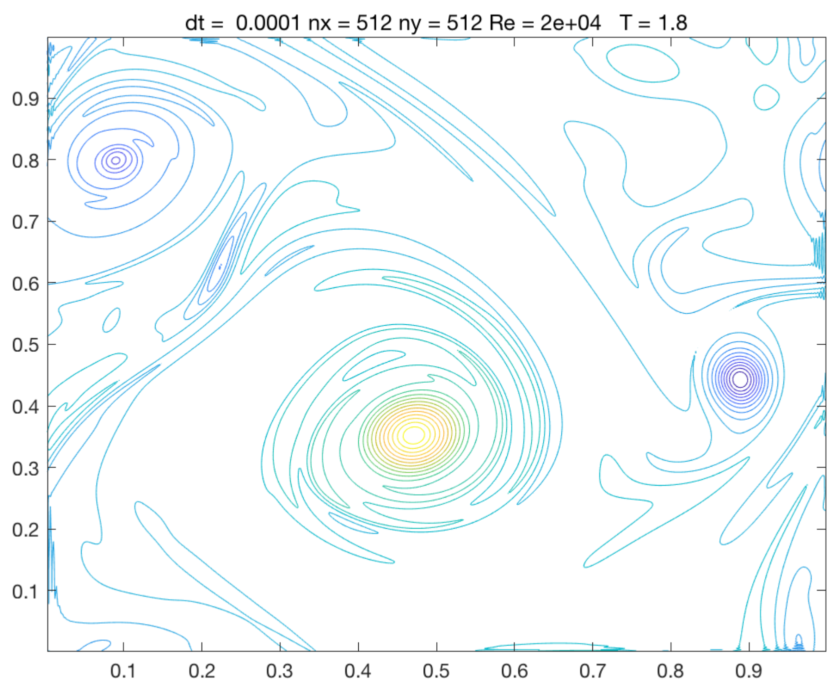

dt = 0.0001 nx = 128 ny = 128 Re = 2e+04   T = 0.8
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9


dt = 0.0001 nx = 128 ny = 128 Re = 2e+04   T = 1.8
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9


dt = 0.0001 nx = 256 ny = 256 Re = 2e+04 T = 0.8
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9


dt = 0.0001 nx = 256 ny = 256 Re = 2e+04 T = 1.8
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9


dt = 0.0001 nx = 512 ny = 512 Re = 2e+04 T = 0.8
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9


dt = 0.0001 nx = 512 ny = 512 Re = 2e+04 T = 1.8
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

4 后台阶问题

后台阶问题会涉在出流边界有 $\frac{\partial u}{\partial n} = \frac{\partial v}{\partial n} = 0$, 这样我们用 U^n, V^n 的右边界值更新 uE, vE, 最后的计算结果如下:

图 1: 第一种初值

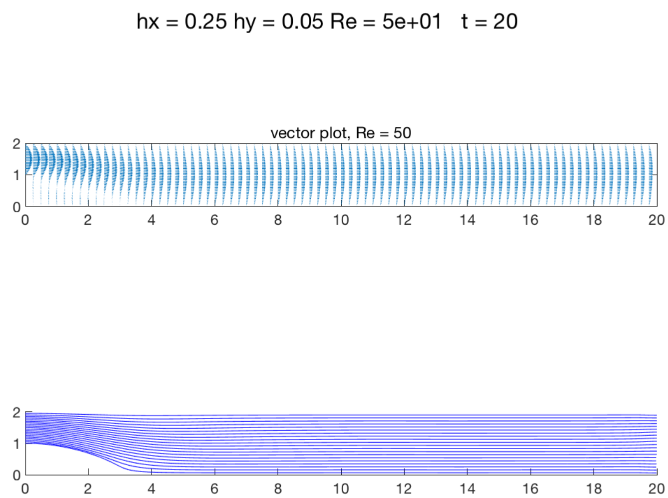
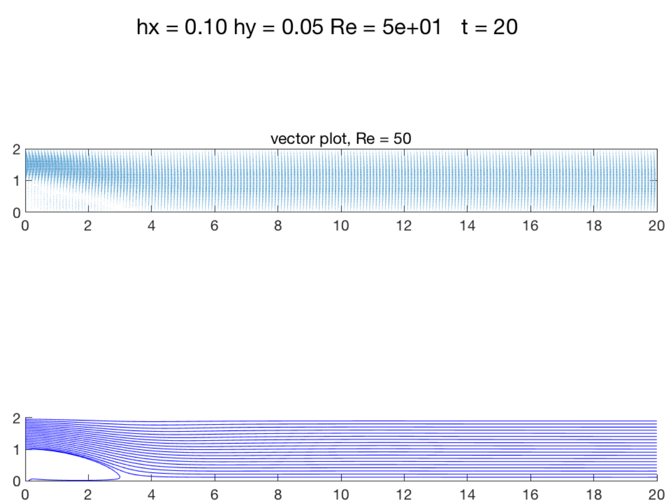


图 2: 第二种初值



5 代码说明

本次提交代码文件分别包括 main.m, mit18086_navierstokeshw.m Readme.txt 等 5 个文件，具体说明参见 Readme.txt.

参考文献

[1] Benjamin Seibold, A compact and fast Matlab code solving the incompressible Navier-Stokes equations on rectangular domains, Applied Mathematics, Massachusetts Institute of Technology, March 31, 2008.