# 计算流体力学

# 第五次作业

王煜沣

2200011013

2025.4

## 作业要求：

在单位正方形内, 求解不可压缩流动。其中只有上边界是是水平运动的, 其他边界都是固定的。当上边界以均匀速度 1 运动时, 角点处存在奇性, 使得此问题和网格是有关的。此时角点处涡量的大小依赖于角点处的网格尺度 ,如图1所示,角点涡量为 。为了避免这种速度场在角点附近的奇性,可以对上边界给出一个速度分布

此分布在角点处函数值和导数值均为零,具有比较好的连续性。

因为所有的边界都是固壁边界, 方腔流动中不需要人为引入出口处的边界条件, 计算中的随意性比较少, 做为经典算例, 通常用来验证格式的计算精度。 通常可以比较的量有中间线上的速度剖面,主涡涡心的位置和流函数值,角附近的二次涡的位置, 三次涡的位置等。

可以使用C/Fortran/Python/Matlab等语言,编程计算正方形内的流场,运动粘度 ,画出流线图,并选取以上可以比较的量简单讨论计算结果。

## 数理算法原理

流动初态如图所示，本题为二维流动问题。初始时刻，只有上边界水平运动，其他边界都是固定边界。假设流动不可压，由于黏性，上边界驱动内部流场运动，产生涡旋。涡量流函数方法每步通过更新计算域中各点涡量和流函数值来迭代求解，流动速度则通过流函数求方向导数得到，适用于二维不可压问题的求解。并且本题可以比较的量主要是流函数值、涡心位置等，所以本问题采用涡量流函数方法求解具有天然优势。

求解的方程组为如下的涡量流函数方程：

边界条件为壁面的流函数值或其法向导数值（也就是切向速度）

注意这里的切向 *τ* 是流动区域的内法向 *n* 逆时针旋转 90°的结果。

对于微分方程，空间导数项可以用中心差分格式离散，时间导数项可以用向前Euler法，具体的计算过程可以分如下三个步骤：

1. 计算 *n* + 1 时刻内点的涡量
2. 计算 *n* + 1 时刻的流函数，方程和边界条件为  
   “homework4”中用对称Gauss-Seidel迭代求解了拉普拉斯方程，采用松弛因子加速了求解过程，此处也用类似的方法，具体步骤为：  
   （1）上行更新  
   （2）下行更新  
   松弛因子为，最终每次更新流函数的方法为：  
   误差取二范数：  
   另外本题中，边界上只有切向速度，故流函数沿切向保持不变，而且流函数差一个常数对计算没有影响，所以可以设置边界上流函数值为0。
3. 计算 *n* + 1 时刻边界上的涡量  
   采用壁面涡量的Thom公式

初始时刻，将整个计算域的流函数*ψ*和涡量*ω*设置为0，最终当*ω*的误差小于允差或者迭代步数超过限制时停止迭代。

注意，本问题的CFL数应小于1，故时间步长应小于空间步长计算才能稳定。

## 代码生成与调试

'code'文件夹中包含本次作业的所有代码，其中“homework5.m”为主程序，运行该程序可以得到所有结果图；“iterateStreamfunctionField.m”为求解流函数拉普拉斯方程的函数，主程序每步运算都会调用该函数。此处只展示主程序中的核心代码部分：

%% Main Loop

Loopout = 1e-4;

iteration = 0;

maxIterations = 100000;

deltaOmega = inf;

while (deltaOmega > Loopout) && (iteration < maxIterations)

iteration = iteration + 1;

%% Step 1: Calculate Vorticity for Inner Points

Omega\_old = Omega;

for i = 2:Ny+1

for j = 2:Nx+1

Omega(i,j) = Omega\_old(i,j) + dt \* (...

(Psi(i+1,j) - Psi(i-1,j)) / (2\*dy) \* (Omega\_old(i,j+1) - Omega\_old(i,j-1)) / (2\*dx) - ...

(Psi(i,j+1) - Psi(i,j-1)) / (2\*dx) \* (Omega\_old(i+1,j) - Omega\_old(i-1,j)) / (2\*dy) + ...

mu \* ((Omega\_old(i+1,j) - 2\*Omega\_old(i,j) + Omega\_old(i-1,j)) / dy^2 + ...

(Omega\_old(i,j+1) - 2\*Omega\_old(i,j) + Omega\_old(i,j-1)) / dx^2));

end

end

%% Step 2: Calculate Streamfunction

Psi = iterateStreamfunctionField(Psi, relaxationFactor, Nx+2, Ny+2, Lx, Ly, Omega, 1e-6);

%% Step 3: Calculate Vorticity for Boubndary Points

for i = 2:Ny+1

Omega(i, 1) = -2 \* (Psi(i, 2) - Psi(i, 1) + vLeft(i-1) \* dx) / dx^2; % Left wall

Omega(i, end) = -2 \* (Psi(i, end-1) - Psi(i, end) + vRight(i-1) \* dx) / dx^2; % Right wall

end

for j = 2:Nx+1

Omega(1, j) = -2 \* (Psi(2, j) - Psi(1, j) + vBottom(j-1) \* dy) / dy^2; % Bottom wall

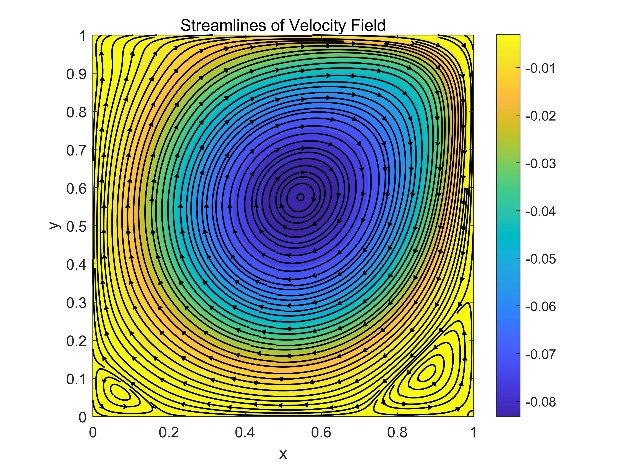
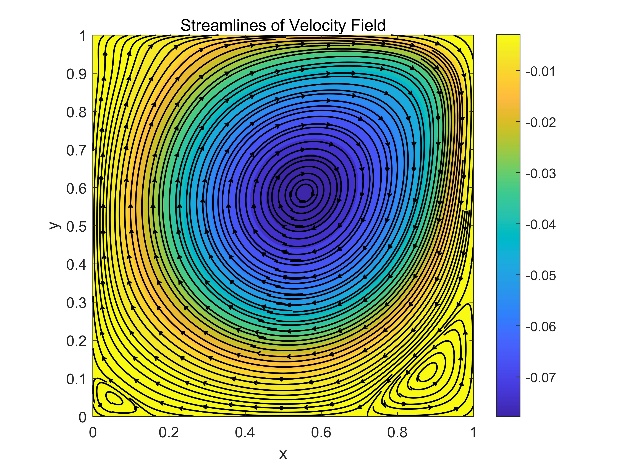
Omega(end, j) = -2 \* (Psi(end-1, j) - Psi(end, j) + vTop(j-1) \* dy) / dy^2; % Top wall

end

error = (Omega - Omega\_old).^2;

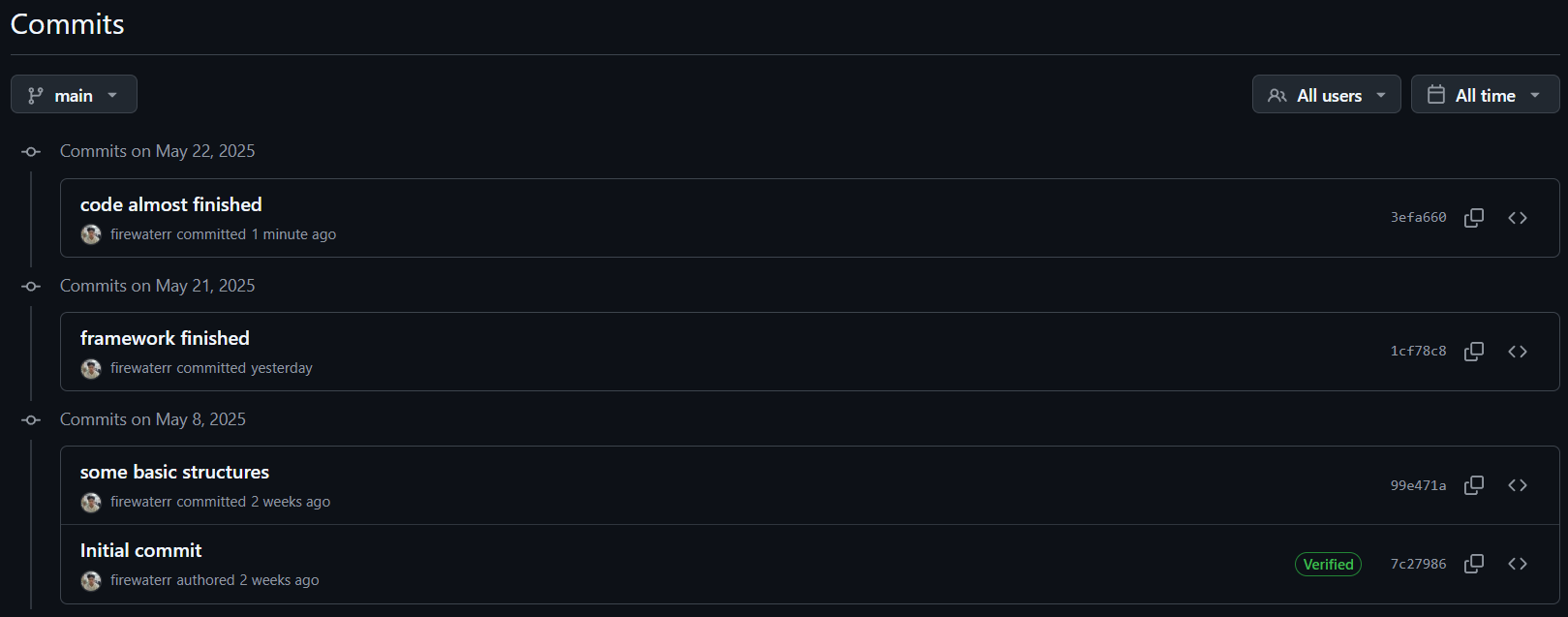
deltaOmega = sqrt(mean(error(:)));

end

首先进行网格无关性验证。此处比较网格数100\*100和200\*200两种情形下的流线图区别。设置最大CFL数为0.5，即时间步长为0.5\*网格尺度

如图为两种不同网格数下流线图的对比，左图为100\*100网格，右图为200\*200网格。可见两者相差不大，而200\*200网格算例相对100\*100计算时间成本约20倍，为了节约计算成本，接下来的结果都是在100\*100网格的情形下得到的

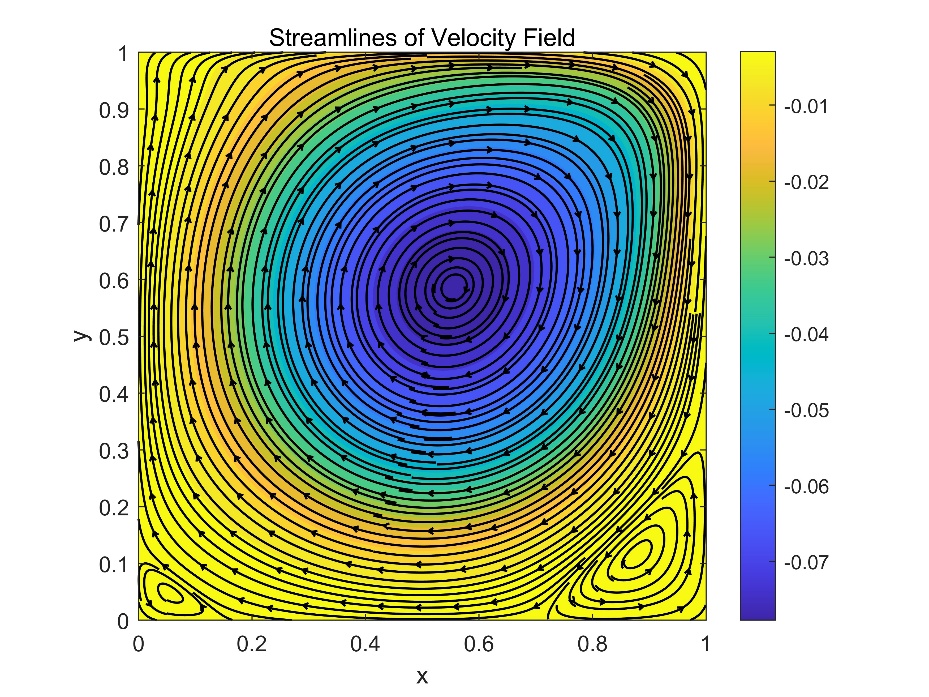
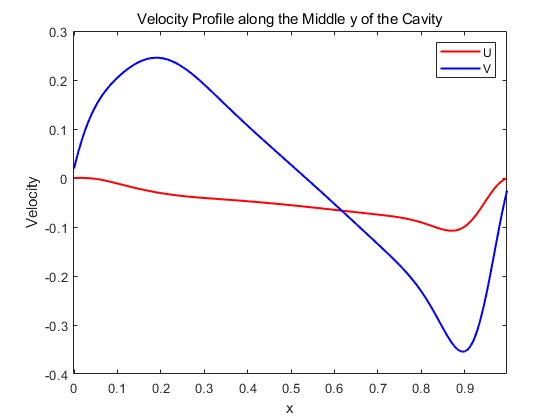
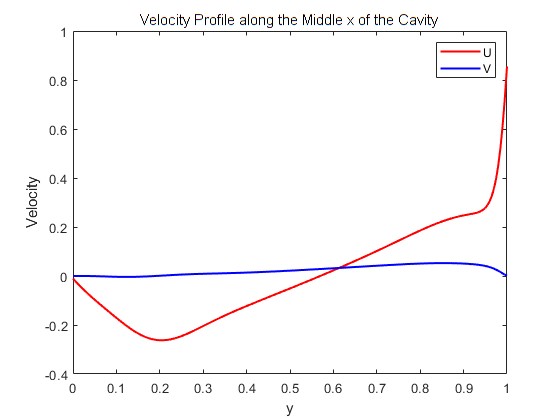
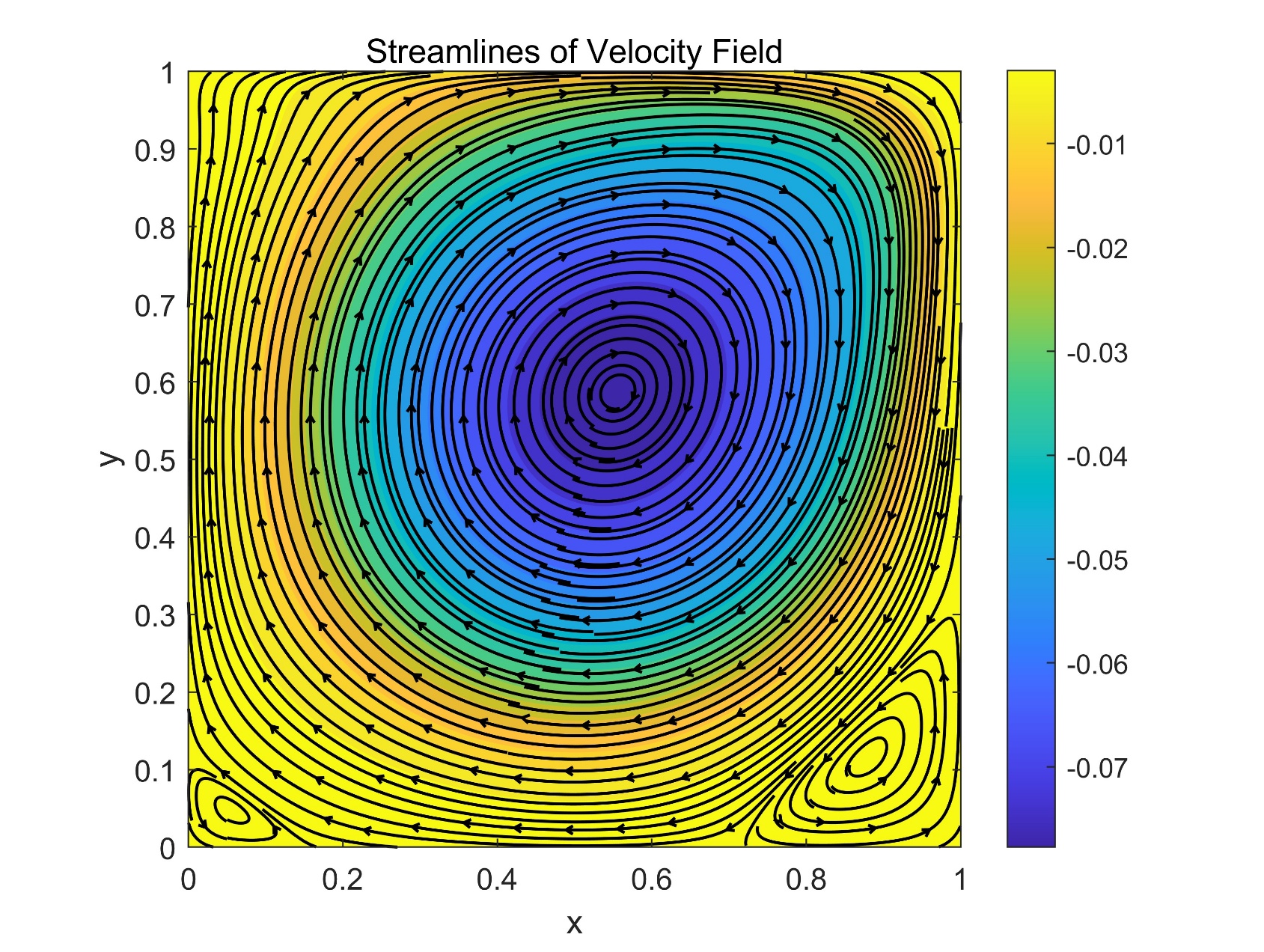
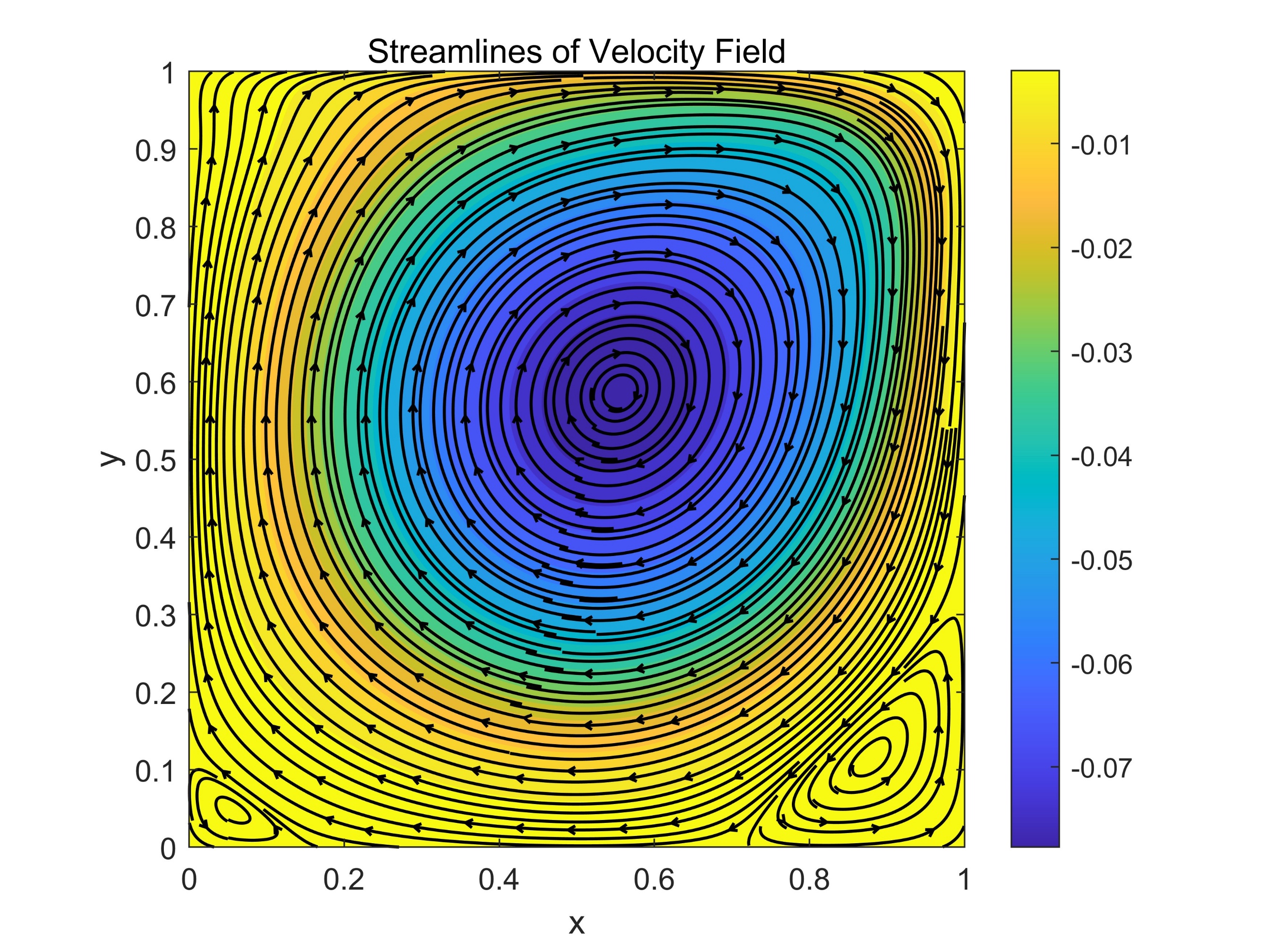
GitHub提交记录截图如下：



GitHub网址为：[firewaterr/CFDHW5: fifth homework of CFD](https://github.com/firewaterr/CFDHW5)欢迎助教检查

## 结果讨论和物理解释

此处比较一些重要的结果

1. 流线图  
     
   如图为方腔流动的流线图。右侧不同代表流函数的等值图。因为流函数为常数的图线就是流线，所以流函数等值图与流线图几乎完全重合。两者不同之处体现在角落处。角落的位置由于流函数的值很接近于0，所以流函数等值图上没有体现出流线的方向。但是流线图只代表当地流场的方向，与流动速度大小无关，所以可以体现在图线上。
2. 中间线上的速度剖面  
     
   由图可知，由于流线图的涡旋性，x中轴线上主要是水平速度u在起作用，y较小时水平速度向左，y较大时水平速度向右；y中轴线上主要是垂直速度v在起作用，x较小时水平速度向上，x较大时水平速度向下，正好与涡量的方向保持一致。由无滑移边界条件，这两个速度在边界处均趋向于壁面的边界速度。
3. 主涡涡心位置和流函数值  
   由计算结果可以看出，主涡涡心位置正好为流函数的极小值点，算出流函数极值点，即可通过流函数极小值点的位置确定主涡涡心位置。由此代码的输出结果为：  
   最小值: -0.077786  
   横坐标 (行): 0.54902  
   纵坐标 (列): 0.58824  
   如图所示为主涡：  
   
4. 角附近的二次涡、三次涡位置  
   与（3）同理，不过角附近二次涡、三次涡的流函数均为当地极大值点。用类似的方法可以确定涡心位置。代码的输出结果为：  
   二次涡：  
   最大值: 0.00079545  
   横坐标 (行): 0.86275  
   纵坐标 (列): 0.12745  
   如图所示为角落处的二次涡：  
     
   三次涡：三次涡的极大值远小于二次涡极大值，此处直接读图中坐标：  
   最大值: 2.37147e-05  
   横坐标 (行): 0.0606061  
   纵坐标 (列): 0.0505051  
   如图所示为角落处的三次涡  
   