Indian Institute of Technology, Kanpur
Academic Semester: 2021-22 (I)

ME634: Advanced Computational Fluid Dynamics
Instructors: Prof. Anikesh Pal

ASSIGNMENT 4
SOLVING FOR A LID-DRIVEN CAVITY USING RK3-CN

Date: 14th April, 2021

Pragya Patel
17807477

Contents

Appendix (Formulation)

Input Values:

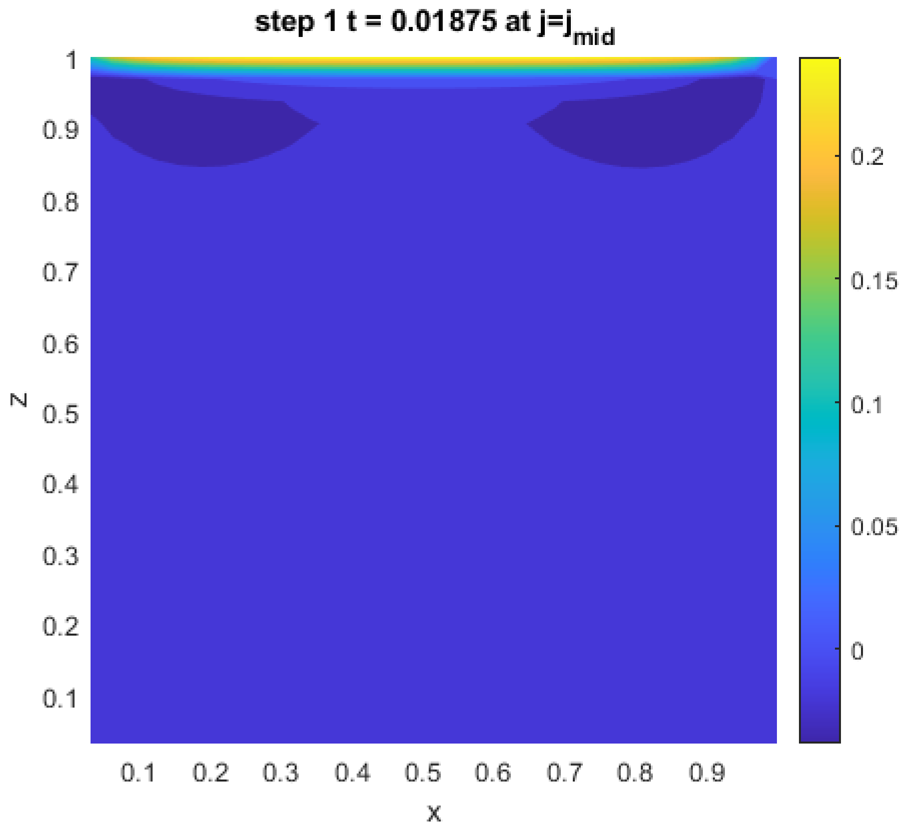
CFL = 1.2; L = 1; Lx = 1; Ly = 1; Lz = 1;

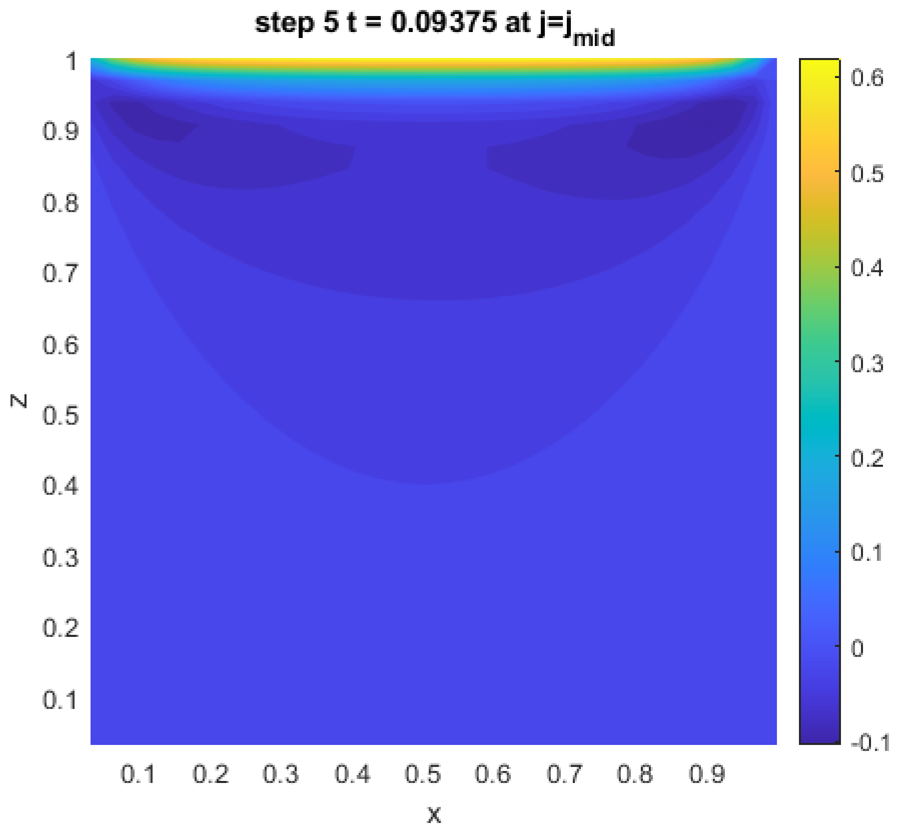
N = 32; Nx = 32; Ny = 32; Nz = 32;

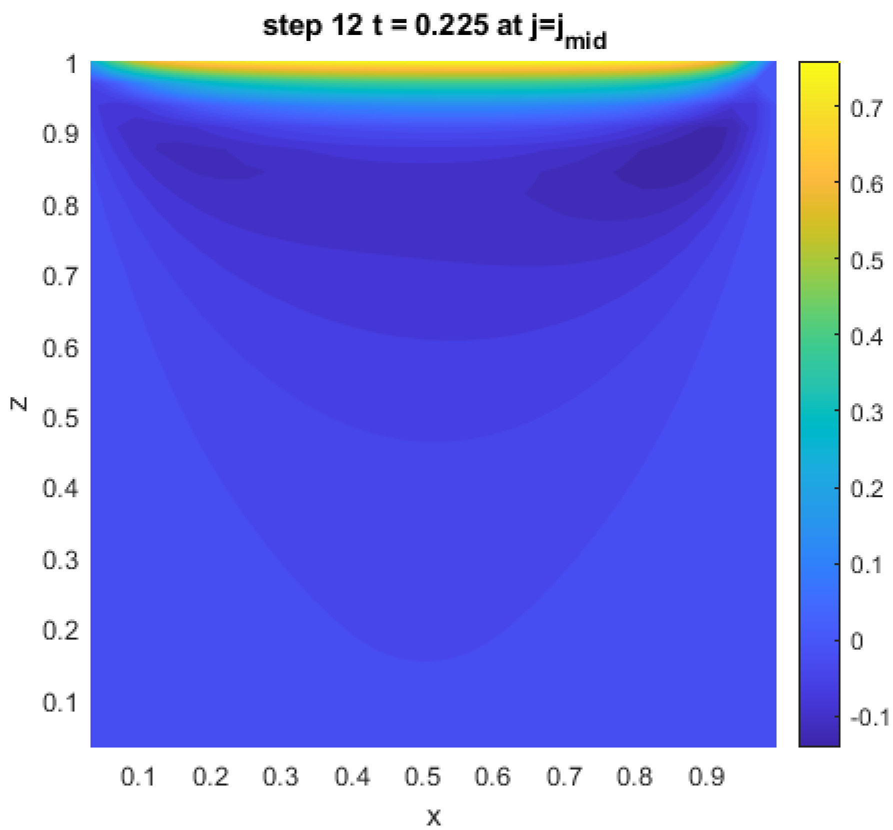
Re = 100; U = 1; (U at lid) nu = 0.01;

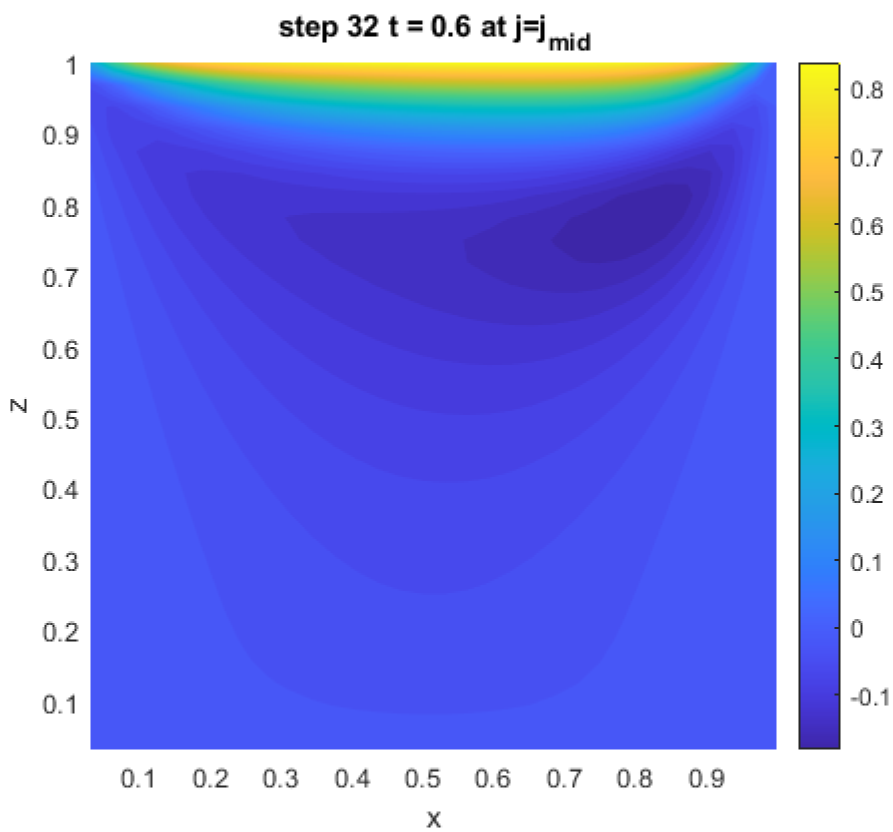
Plots of u(j=j-midplane) at different time steps

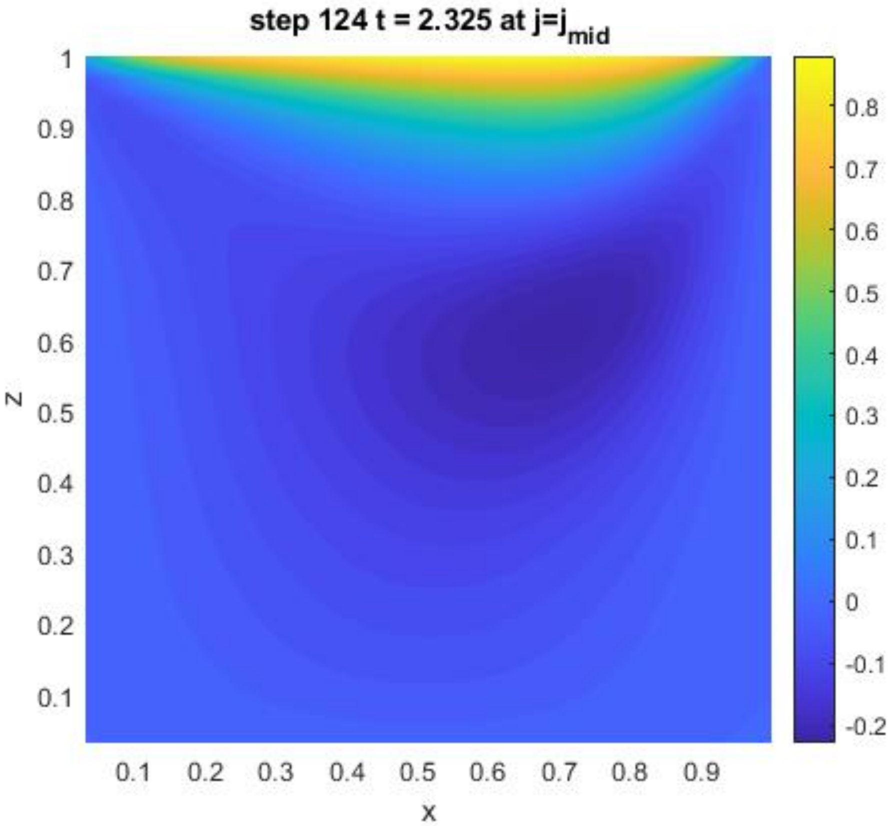
Note: x and y axes show ith and kth nodes

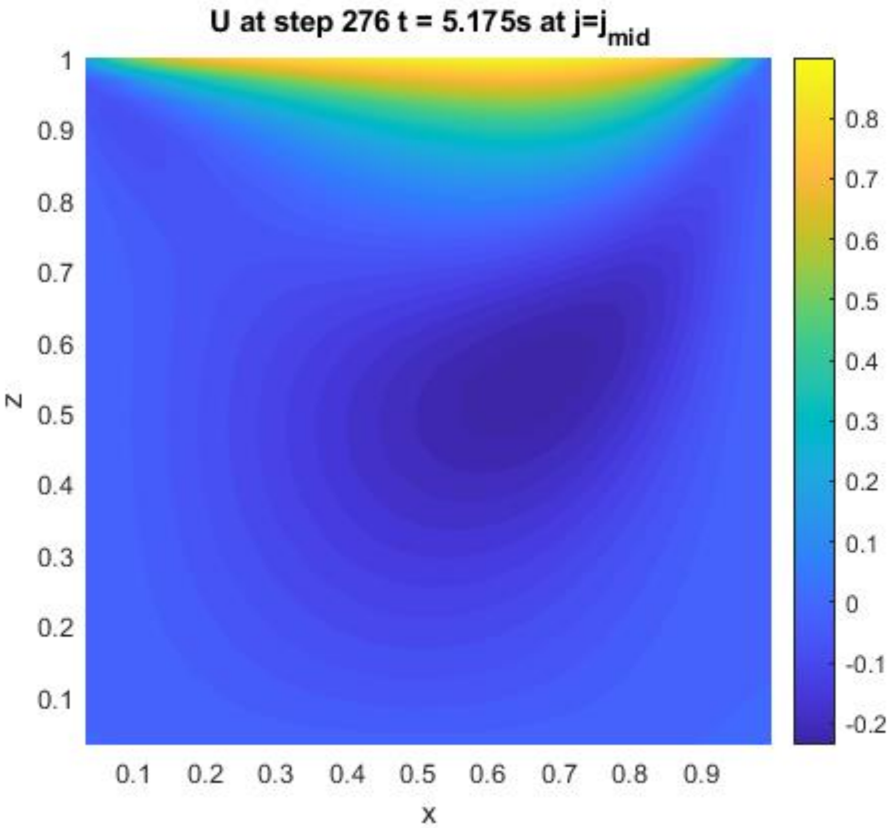




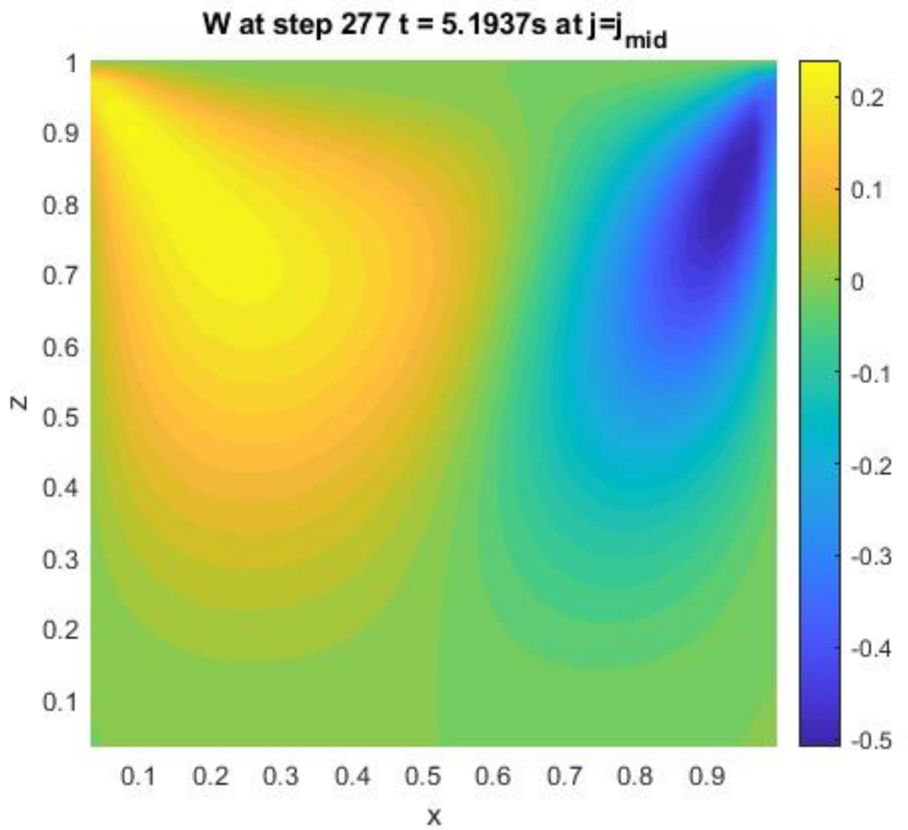

step 12 t = 0.225 at j=j_mid
z
x
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
-0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7

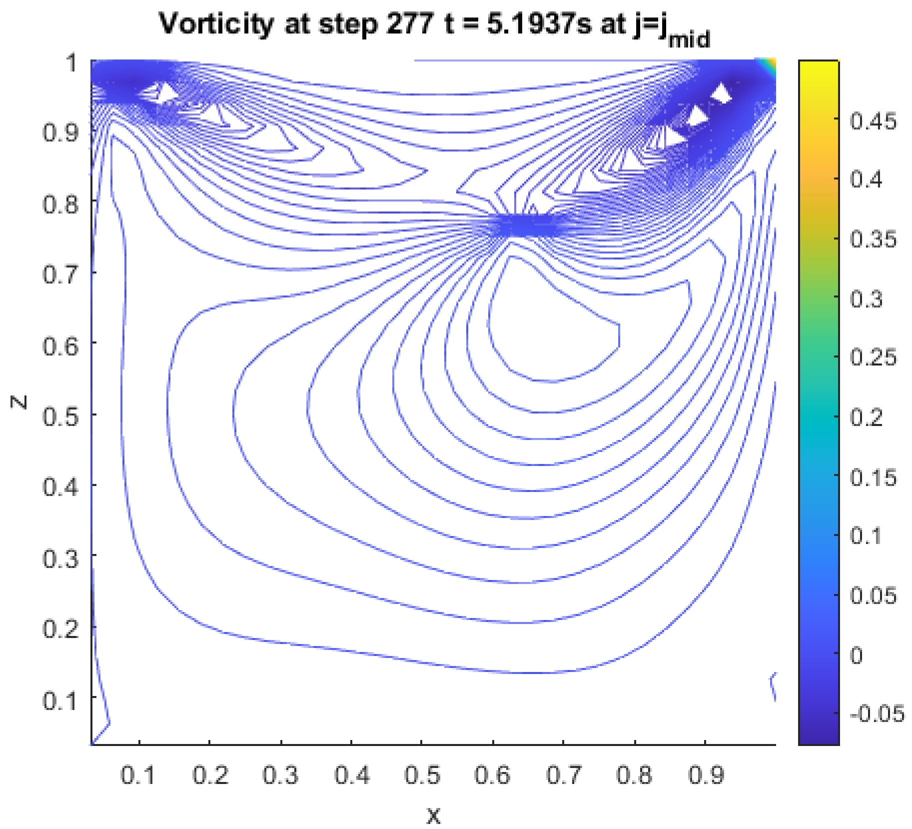

step 32 t = 0.6 at j=j_mid
z
x
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
-0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8


step 124 t = 2.325 at j=j_mid
z
x
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
-0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8


U at step 276 t = 5.175s at j=j_mid
z
x
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
-0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8

Plots of w(j=j-midplane) and vorticity at steady-state





ME634
ASSGN 4

PRAGYA PATEL
17807477

3D Lid Driven Cavity (laminar) — using RK3 CN Alg

A) Incompressible N-S momentum eqn:

$$\frac{\partial \underline{u}}{\partial t} + (\underline{u} \cdot \nabla)\underline{u} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \underline{u} + \vec{g}$$

assuming no body forces, non-dimensionalize using the variables:

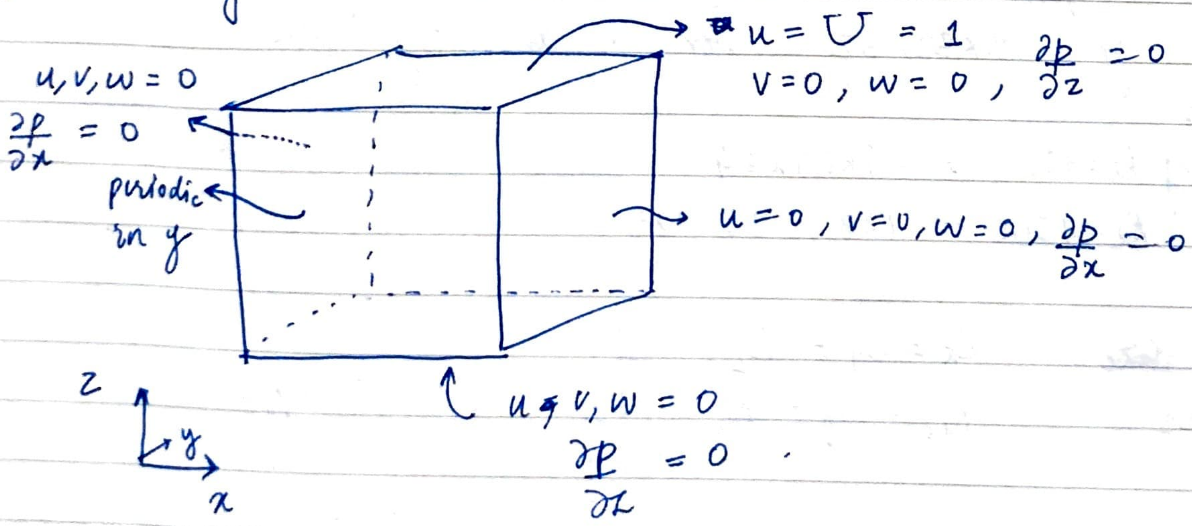
$$\vec{x}^* = \frac{x}{L_z} \qquad u^* = \frac{u}{U_0} \qquad v^* = \frac{v}{U_0} \qquad w^* = \frac{w}{U_0}$$

$$\nabla^* = L_z \nabla \qquad p^* = \frac{p}{\rho U_0^2} \quad \text{and} \quad t^* = \frac{t U_0}{L_z}$$

Non-dimensionalized:

$$\frac{\partial \underline{u}^*}{\partial t} + (\underline{u}^* \cdot \nabla^*)u^* = -\nabla^* p + \frac{1}{Re}\nabla^{*2}\vec{u}^*.$$

B) Boundary Conditions



C) Domain Discretization

- We follow a mixture of explicit & implicit discretization (Crank Nicolson for implicit part)

Diffusive part : $\underbrace{\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}}_{\text{explicit}} + \underbrace{\frac{\partial^2}{\partial z^2}}_{\text{implicit}}$

Convective part : $\left.\begin{array}{l} \frac{\partial(uu)}{\partial x} + \frac{\partial(uv)}{\partial y} + \frac{\partial(uw)}{\partial z} \\ \frac{\partial(vu)}{\partial x} \ldots\ldots \\ \ldots\ldots \end{array}\right\}$ explicit.

$\Rightarrow$ [EXPLICIT] = [CONVEC] + [E-DIFF] _ (a/c section (A))

↳ Grid

- For this assignment, uniform grid was used.
- It has been found that the ~~gri~~ value of Re = 100 flow can be treated by a grid size of 32×32×32 for a converging soln.

Similarly Re = 400 → 64×64×64
Re = 1000 → 128×128×128

Domain size $L_x = L_y = L_z = 1$
No. of cells $N_x = N_y = N_z = 32$ → for Re = 100
⋮ ... 400
⋮ ... 1000

D) Time discretization

~~a~~ – We ~~fo~~ follow the Runge-Kutta method (RK3) to discretize the unsteady term ~~fo~~
(⇒ RK3 for explicit part).

↳ Time steps.

for the system to reach convergence, it must satisfy the CFL criterion

for 3D :

$$\Delta t_x = \min\left(C \frac{\Delta x(i)}{u(i,j,k)}\right) = \min\left(C \frac{\Delta x}{u(i,j,k)}\right)$$

(uniform grid here)

$$\Delta t_y = \min\left(C \frac{\Delta y}{v(i,j,k)}\right) \qquad \Delta t_z = \min\left(C \frac{\Delta z}{w(i,j,k)}\right)$$

$$\Delta t = \min(\Delta t_x, \Delta t_y, \Delta t_z)$$

↳ Further, we define a matrix with coefficients relevant to the RK3 scheme :

m	C_1	C_2	C_3	
1	0	1/3	1/3	→ coeffs for RK3 step 1
2	-5/9	15/16	5/12	→ for step 2.
3	-153/128	8/15	1/4	→ 3.

where, for each RK-substep, $\Delta t_{RK}^{m} = C_3^{m} \Delta t$

E) Solving for the flow

1. liddriven.m : The main file, uses functions
   a) coeff-uni : coeff matrices for uniform grid
   b) updatebc : for updating u, v, w and p boundary conditions

looped
   c) deltat : for calculating Δt (as in (D))
   d) rk31, rk32, rk33 } rk3 substeps
      u, v, w
      urk1, vrk1, wrk1
      urk2, vrk2, wrk2.
      ↳ final output unew, vnew, wnew
   e) L2norm : for checking convergence of u.
   f) midplane : plotter

1. a. –

1. b. → u, v, w and p boundary conditions (as in (B))

1. c. → deltat takes in the u, v, w matrices, as well as the global variable "CFL" constant.

1. d. → RK1 : uses

qu1 qv1 qw1 & utemp2 ...

(i) coeff-rk3 : for constants (as in (D))

(ii) urhs, vrhs, wrhs } calculates the explicit part i.e. [CONVECU] + [DIFFU] when transferred to the RHS.

(iii) thomas : Thomas algorithm solver for getting the u^*, v^* & w^*

↳ along k-columns, at each (i,j) pair

(iv) mgsolver : as done in ~~a~~ assignment 3.

└ takes C1, C2, C3, C4 matrices as defined in the main file "liddriven.m".

↓

solves

↓

Poisson's Eqⁿ : $\nabla^2 p = \frac{1}{\Delta t_{rk}} \cdot (\nabla \vec{u}^*)$

calc using thomas alg. & updatebc

↓

prk1

correction term is then added. $\left[-\Delta t \frac{(\delta P)}{\Delta x} \ldots \right]$

qu2, qv2, qw2 ready for next RK

similarly rk32

rk33

↓

u, v, w at the end of 3 rk substeps — (each including the steps for solving the implicit part).