UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH
Swansea University Prifysgol Abertawe
CENTRALE NANTES
Universität Stuttgart

MSc in Computational Mechanics

Finite Elements in Fluids

MATLAB Assignment 4: Unsteady Navier-Stokes equations

Submitted By:
Mario Alberto Méndez Soto

Submitted To:
Prof. Pablo Saez
Prof. Antonio Huerta

Spring Semester, 2019

1 Unsteady cavity flow problem

1.1 Governing equations

Unsteady isothermal viscous fluid phenomena, where inertial forces (driven by pressure and viscous terms) become important, can be mathematically described using the so-called Navier Stokes equations:

$$\begin{cases} \mathbf{v}_t - \nu\nabla^2\mathbf{v} + (\mathbf{v}\cdot\nabla)\mathbf{v} + \nabla p = \mathbf{b} & \text{in } \Omega \\ \nabla\cdot\mathbf{v} = 0 & \text{in } \Omega \\ \mathbf{v} = \mathbf{v}_D & \text{on } \Gamma_D \\ \mathbf{n}\cdot\sigma = \mathbf{t} & \text{on } \Gamma_N \end{cases} \tag{1}$$

In this case, since the fluid is considered to be isothermal, coupling with the equation of energy conservation is not necessary. Instead, only conservation of mass and momentum is considered. Generally, any Navier-Stokes solution can be entirely characterized by the Reynolds Number $Re = \dfrac{V_{ref}L_{ref}}{\nu}$.

The weak formulation of system of equations (1) (with only Dirichlet boundary conditions) can be obtained, as usual, by multiplying the system of equations by a vector weighting function $\mathbf{w}$ and a scalar weighting function q, such that $\mathbf{w} \in \mathcal{V} = \left\{\mathbf{w} \in \mathcal{H}^1(\Omega)/\mathbf{w} = \mathbf{0} \text{ on } \Gamma_D\right\}$ and $q \in \mathcal{L}_2(\Omega)$. Afterwards, the Galerkin spatial discretization proceeds as previously done for Stokes and steady Navier-Stokes problems. Finally, the finite element discretization of the system of equations (1) becomes a system of semi-discrete equations for $t \in]0, T[$, as follows:

$$\begin{cases} \mathbf{M}\dot{\mathbf{u}}(t) + [\mathbf{K} + \mathbf{C}(\mathbf{v})]\mathbf{u} + \mathbf{G}p = \mathbf{b} \\ \mathbf{G}^T\mathbf{u}(t) = 0 \\ \mathbf{u}(\mathbf{0}) = \mathbf{v}(0) - \mathbf{v}_D(0) \end{cases} \tag{2}$$

To track the solution at different time steps, system (2) can be discretize in time using finite difference schemes (refer to section 2) or fractional-step procedures, in which the computations are decomposed into a sequence of two or more steps (refer to section 3).

1.2 Boundary conditions

For this report, the classical benchmark cavity problem was analyzed (see Figure (1)). For this problem, a confined incompressible isothermal flow is located in a square lid-driven cavity. The lid can move with a vertical velocity of 1, while the rest of the sides are fixed. Since two upper corners belong to the fixed vertical walls, singularities are introduced in the pressure field as it will be noticeable later on. Moreover, the lower part of the boundary is prescribed with zero pressure field (since only Dirichlet boundary conditions are considered).

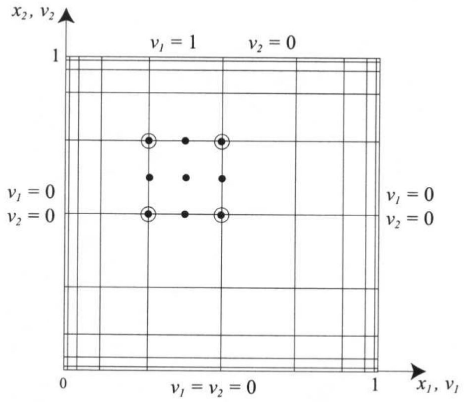


Fig. 1 – Cavity problem: problem statement and boundary conditions

2 Semi-implicit first order monolitic scheme

2.1 Mathematical description

The semi-implicit discretization scheme is obtained using a procedure based on the theta-family of methods. For this single-step method and neglecting truncation errors, value u^{n+1} at time t^{n+1} is determined from value u^n at t^n, as follows:

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = \theta \mathbf{u}_t(t^{n+1}) + (1-\theta)\mathbf{u}_t(t^n) \tag{3}$$

The parameter θ is taken to be in the interval $[0, 1]$ and determines whether the scheme is implicit or explicit. Moreover, for $\theta \geq 0.5$ the methods are unconditionally stable and produce non-spurious results regardless of the time discretization used.

To obtain the corresponding system of equations for the unsteady Navier Stokes equations, we can substitute a rearranged form of the momentum equation (first equation in system (2)):

$$\mathbf{u_t} = \frac{\mathbf{b} - (\mathbf{K} + \mathbf{C}(v))\mathbf{u} - \mathbf{G}p}{\mathbf{M}} \tag{4}$$

into equation (3).

Thus, the aforementioned equation becomes:

$$\frac{\Delta \mathbf{u}}{\Delta t} = \theta \Big(\frac{\mathbf{b} - (\mathbf{K} + \mathbf{C}(v^{n+1}))\mathbf{u}^{n+1} - \mathbf{G}p^{n+1}}{\mathbf{M}} \Big) + (1-\theta) \Big(\frac{\mathbf{b} - (\mathbf{K} + \mathbf{C}(v^n))\mathbf{u}^n - \mathbf{G}p^n}{\mathbf{M}} \Big) \tag{5}$$

where $\Delta \mathbf{u} = \mathbf{u}^{n+1} - \mathbf{u}^n$.

In particular the semi-implicit method uses an implicit scheme ($\theta > 0$), but treating the convection term explicitly, i.e. evaluating the convection matrix at t^n, instead of $t^{n+\theta}$. Thus, the resultant system is linear and it can be solved without resorting to non-linear solution methods (Newton-Raphson or Picard's for instance). Rearranging equation (5) and evaluating the convection matrix at t^n, it yields:

$$\big(\mathbf{M} + \theta\Delta t(\mathbf{K} + \mathbf{C}^n)\big)\Delta\mathbf{u} + \Delta t\theta\mathbf{G}\Delta p = \Delta t\big(\mathbf{b} - (\mathbf{K} + \mathbf{C}^n)\mathbf{u}^n - \mathbf{G}p^n\big) \tag{6}$$

Furthermore, since the incompressibility condition must be satisfied at any time t, we can impose it directly to $\Delta\mathbf{u}$. Thus, the final discretized version of the unsteady Navier Stokes system of equations can be written as:

$$\begin{cases} \big(\mathbf{M} + \theta\Delta t(\mathbf{K} + \mathbf{C}^n)\big)\Delta\mathbf{u} + \Delta t\theta\mathbf{G}\Delta p = \Delta t\big(\mathbf{b} - (\mathbf{K} + \mathbf{C}^n)\mathbf{u}^n - \mathbf{G}p^n\big) \\ \mathbf{G}^T\Delta\mathbf{u} = 0 \end{cases} \tag{7}$$

Using a matrix notation, the system becomes:

$$\begin{bmatrix} \mathbf{M} + \theta\Delta t(\mathbf{K} + \mathbf{C}^n) & \Delta t\theta\mathbf{G} \\ \mathbf{G}^T & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{p} \end{bmatrix} = \begin{bmatrix} \Delta t\big(\mathbf{b} - (\mathbf{K} + \mathbf{C}^n)\mathbf{u}^n - \mathbf{G}p^n\big) \\ \mathbf{0} \end{bmatrix} \tag{8}$$

2.2 Computational implementation

Using the originally given code to solve the cavity flow problem, the subroutines were modified in order to utilize previous matricial computations also present in equation (8). The following are the lines added to the code in order to implement the semi-implicit method.

```
while step < nstep
    step = step +1;
    C = ConvectionMatrix(X,T,referenceElement,velo);
    Cred = C(dofUnk,dofUnk);
    fredn = fred - (K(dofUnk,dofDir)+C(dofUnk,dofDir))*valDir;

        % Matricial system of equations
        Atot = [Mred+teta*dt*(Kred+Cred)   dt*teta*Gred'
         Gred    zeros(nunkP)];
        btot = [dt*(fredn-(Kred+Cred)*veloVect(dofUnk)-Gred'*pres);
        zeros(nunkP,1)];

        % Computation of velocity and pressure increment
        solInc = Atot\btot;

        % Update of the solution
        veloInc = zeros(ndofV,1);
        veloInc(dofUnk) = solInc(1:nunkV);
        presInc = solInc(nunkV+1:end);
        velo = velo + reshape(veloInc,2,[])';
        pres = pres + presInc;
end
```

3 Chorin-Temam projection method

3.1 Mathematical description

The principle involved in this methodology consists of the computation of the velocity and pressure fields in separate steps using an intermediate velocity. For a purely Dirichlet problem, the first step includes the viscous and convective terms of equation (1) and aims to find an intermediate velocity field, $\mathbf{v}_{int}^{n+1}$, as follows:

$$\begin{cases} \dfrac{\mathbf{v}_{int}^{n+1} - \mathbf{v}^{n}}{\Delta t} + (\mathbf{v}^{*} \cdot \nabla)\mathbf{v}^{**} - \nu \nabla^{2} \mathbf{v}^{**} = \mathbf{b}^{n+1} \\ \mathbf{v}_{int}^{n+1} = \mathbf{v}_{D}^{n+1} \end{cases} \tag{9}$$

where the velocities $\mathbf{v}^{*}$ and $\mathbf{v}^{**}$ are chosen based on the preferred way to treat the convective term (explicitly, semi-implicitly or implicitly).

For a semi-implicit and implicit cases, a discretization of equations (9) yields the following matricial equation:

$$\mathbf{M}_1 \Big(\frac{\mathbf{v}_{int}^{n+1} - \mathbf{v}^{n}}{\Delta t} \Big) + (\mathbf{C}(\mathbf{v}^{*}) + \mathbf{K})\mathbf{v}_{int}^{n+1} = \mathbf{f}^{n+1} \tag{10}$$

where $\mathbf{f}$ accounts for the body forces $\mathbf{b}$ and the Dirichlet boundary conditions.

The second step of the Chorin-Temam method determines the fields $\mathbf{v}^{n+1}$ and $\mathbf{p}^{n+1}$ by solving the following system:

$$\begin{cases} \dfrac{\mathbf{v}^{n+1} - \mathbf{v}_{int}^{n+1}}{\Delta t} + \nabla p^{n+1} = 0 \\ \nabla \cdot \mathbf{v}^{n+1} = 0 \\ \mathbf{n} \cdot \mathbf{v}^{n+1} = \mathbf{n} \cdot \mathbf{v}_{D}^{n+1} \end{cases} \tag{11}$$

Discretization of equation (11) leads to the matricial system:

$$\begin{cases} \mathbf{M}_2 \Big(\dfrac{\mathbf{v}^{n+1} - \mathbf{v}_{int}^{n+1}}{\Delta t} \Big) + \mathbf{G}\mathbf{p}^{n+1} = \mathbf{0} \\ \mathbf{G}^{T} \mathbf{v}^{n+1} = 0 \end{cases} \tag{12}$$

3.2 Computational implementation

Using the originally given code to solve the cavity flow problem, the subroutines were modified in order to utilize previous matricial computations also present in equations (10) and (12). The following are the lines added to the code in order to implement the Chorin-Temam projection method.

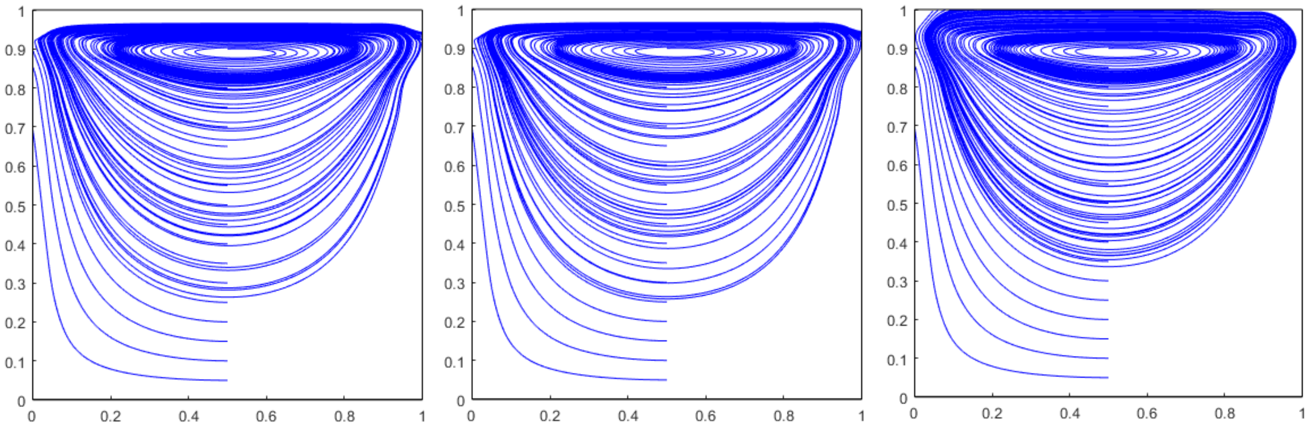
```
while step < nstep
    step = step +1;
    C = ConvectionMatrix(X,T,referenceElement,velo);
    Cred = C(dofUnk,dofUnk);
    fredn = fred - (K(dofUnk,dofDir)+C(dofUnk,dofDir))*valDir;

        % FIRST STEP
        btot = dt*fredn+Mred*veloVect(dofUnk);
        Atot = Mred+dt*(Cred+Kred);
        Z = Atot\btot;

        % SECOND STEP
        btot = [Mred*Z; zeros(nunkP,1)];
        Atot = [Mred Gred'*dt; Gred zeros(nunkP)];
        aux = Atot\btot;

        veloInc = zeros(ndofV,1);
        veloInc(dofUnk) = aux(1:nunkV);
        presInc = aux(nunkV+1:end);
        velo = reshape(veloInc,2,[])';
        pres = presInc;
end
```

4 Results and discussion

Using the implementation presented in the previous sections, the time-dependent cavity problem was solved using Q2Q1 elements. For the case in study, a Reynolds number equal to 200 (with viscosity equal to 0.2) was selected. Figures (2) - (6) show the results obtained with 100 time steps using the semi-implicit (with $\theta = 1$ and $\theta = 0.5$) and Chorin-Temam projection methods.

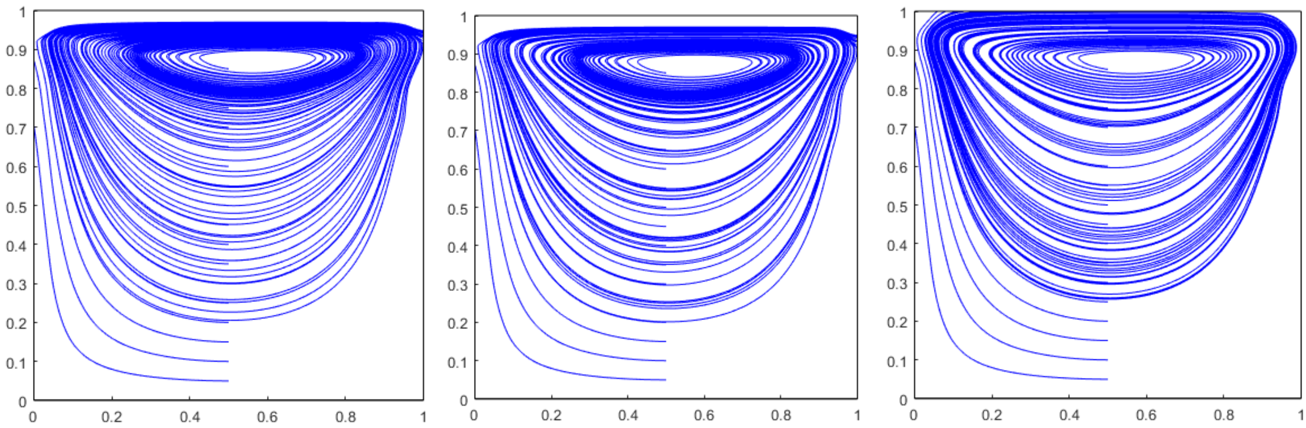
The obtained results prove how the computations are qualitatively the same for all time steps with some negligible differences regarding density of the streamlines in the figures. The only significant difference between Chorin-Temam and the semi-implicit methods is the upper position of the flow, which is predicted by Chorin-Temam to be slightly higher in comparison with the semi-implicit solution.

Figure (7) shows the final pressure field found using the Chorin-Temam projection method. Solution for the semi-implicit methods is omitted since it perfectly coincides with the results by the Chorin-Temam method.



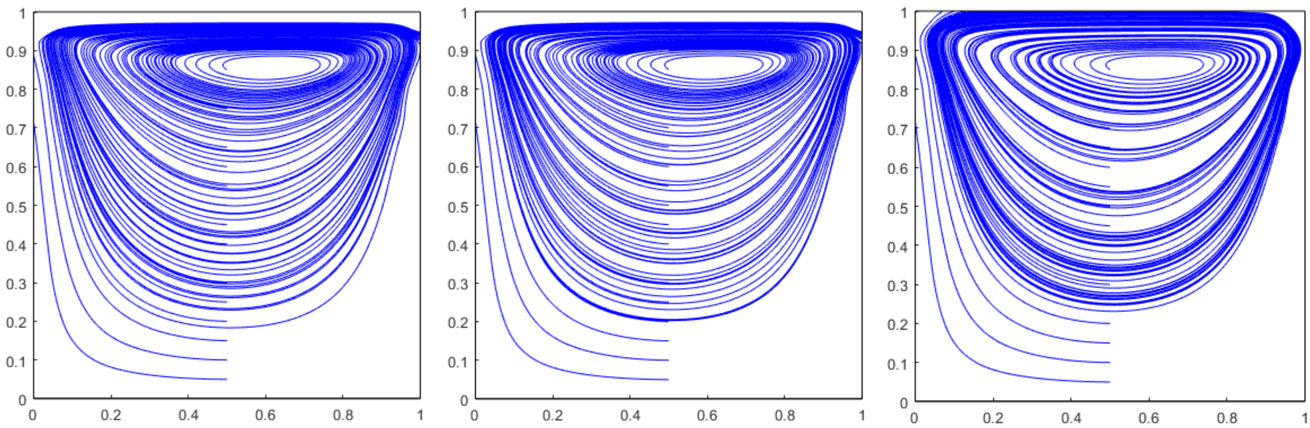
(a) $\theta = 1$ (b) $\theta = 0.5$ (c) Chorin-Temam

Fig. 2 – Streamlines at time step 20



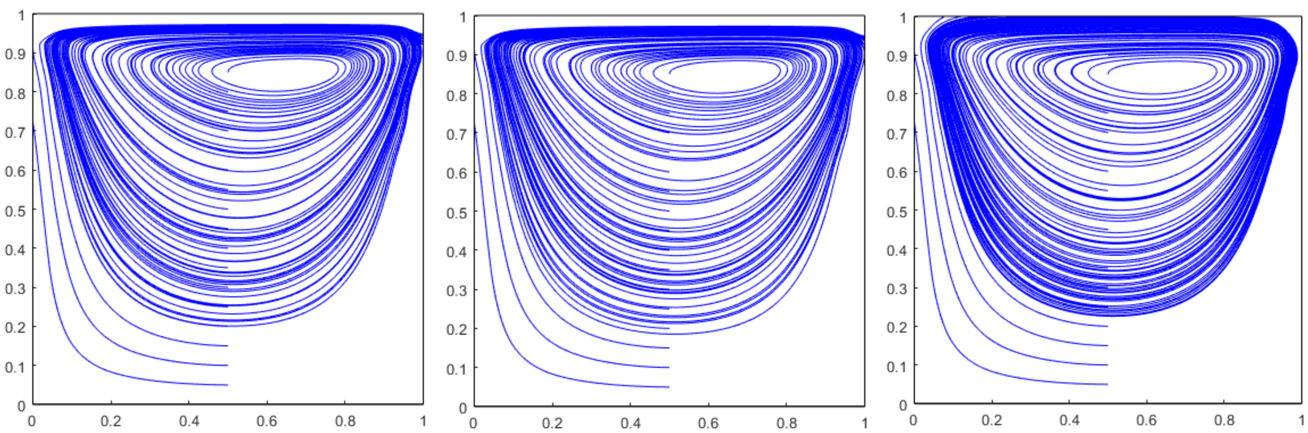
(a) $\theta = 1$ (b) $\theta = 0.5$ (c) Chorin-Temam

Fig. 3 – Streamlines at time step 40



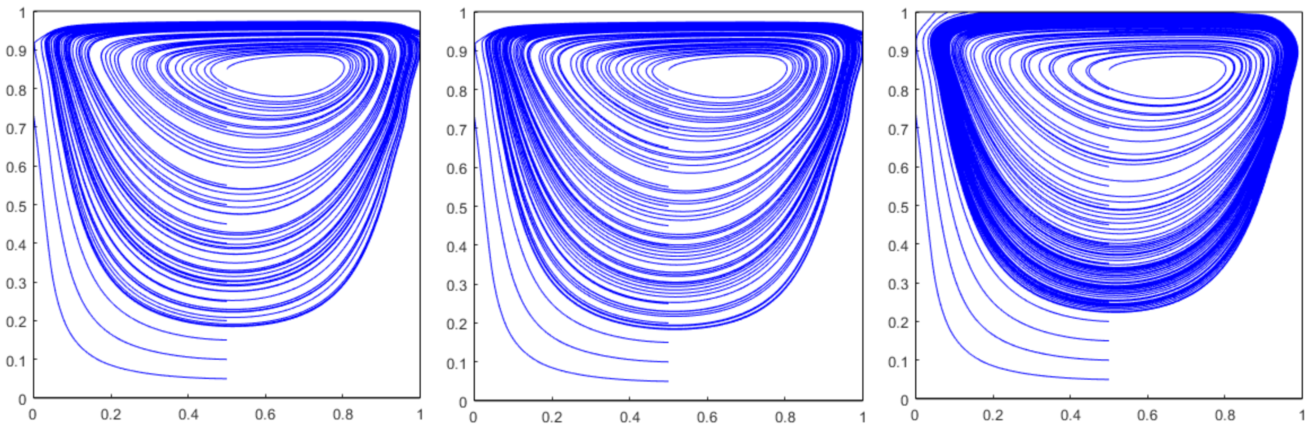
(a) $\theta = 1$ (b) $\theta = 0.5$ (c) Chorin-Temam

Fig. 4 – Streamlines at time step 60



(a) $\theta = 1$ (b) $\theta = 0.5$ (c) Chorin-Temam

Fig. 5 – Streamlines at time step 80



(a) $\theta = 1$ (b) $\theta = 0.5$ (c) Chorin-Temam

Fig. 6 – Streamlines at time step 100

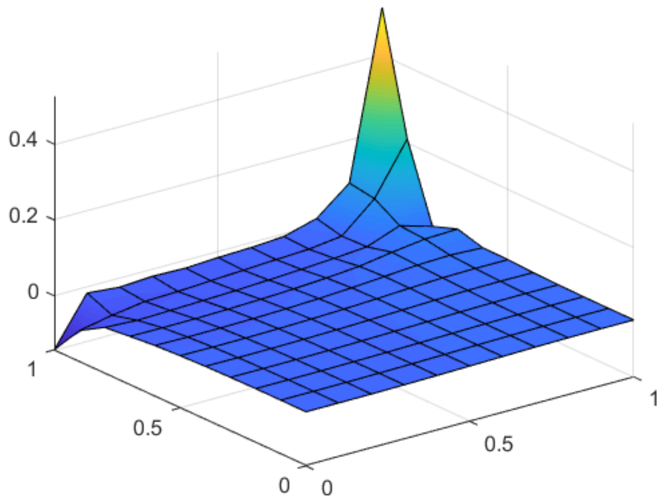


Fig. 7 – Final pressure field