Navier-Stokes numerical example

Problem formulation

Navier-Stokes problem:

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

Weak form:

$$\begin{aligned} &a(\mathbf{w},\mathbf{v}) + c(\mathbf{w},\mathbf{v},\mathbf{v}) + b(\mathbf{w},p) = (\mathbf{w},\mathbf{b}) + (\mathbf{w},\mathbf{t})_{\Gamma_N} && \forall\mathbf{w}\in\mathcal{V} \\ &b(\mathbf{v},q) = 0 && \forall q\in\mathcal{Q} \end{aligned}$$

$$\begin{aligned} a(\mathbf{w},\mathbf{v}) &= \nu(\nabla\mathbf{w},\nabla\mathbf{v} \\ b(\mathbf{v},q) &= -(q,\nabla\cdot\mathbf{v}) \\ c(\mathbf{w},\mathbf{v},\mathbf{v}) &= (\mathbf{w},(\mathbf{v}\cdot\nabla)\mathbf{v} = \int_\Omega \mathbf{w}\cdot(\mathbf{v}\cdot\nabla)\mathbf{v}\,d\Omega \end{aligned}$$

Galerkin discretization:

$$\begin{pmatrix} \mathbf{K}+\mathbf{C}(\mathbf{v}) & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{h} \end{pmatrix}$$

Discretization of the convective term for the Piccard's method

The following code shows the discretization for the linearized form of the convection matrix on the element level.

```
function [C] = ConvectionMatrix(X,T,referenceElement,velo)
...
    v_igaus = N_ig*u_e;
    Ce = Ce + Ngp'*(v_igaus(1)*Nx+v_igaus(2)*Ny)*dvolu;
...
end
```

Discretization of the Newton-Raphson scheme

The following code shows the discretization for the linearized form of the convection matrix on the element level.

```
function [C1,C2] = ConvectionMatrix(X,T,referenceElement,velo)
...
    v_igaus = N_ig*u_e;
    Ce1 = Ce1 + Ngp'*(v_igaus(1)*Nx+v_igaus(2)*Ny)*dvolu;
    Ce2 = Ce2 + Ngp'*([nx ; ny]*u_e)'*Ngp*dvolu;
...
```

The following code shows the implementation of the Newton-Raphson algorithm in the file `mainNavierStokes.m`:

```
...
    [C1,C2] = ConvectionMatrixNR(X,T,referenceElement,velo);
    Cred1 = C1(dofUnk,dofUnk);
    Cred2 = C2(dofUnk,dofUnk);

    A = [Kred + Cred1   Gred';
      Gred      zeros(nunkP)];

    Atot = A;
    btot = [fred - (C1(dofUnk,dofDir))*valDir; zeros(nunkP,1)];

    J = [Kred + Cred1 + Cred2   Gred';
      Gred      zeros(nunkP)];

    F = Atot*sol0 - btot;

    % Computation of velocity and pressure increment

    solInc =-J\F;

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

Results and convergence study

To compare the Piccard's method with the Newton-Raphson scheme the cavity flow flow problem was computed with quadrilateral elements (Q2Q1). Figure 1 and Figure 2 show that in both cases the solution is stable. As expected, it is hard to see any differences with respect to the accuracy of the results. A distinctive property of the two methods becomes only visible when looking at the convergence plots in Figure 3 where the residual F is plotted against the number of iterations. For the Piccard's scheme a linear relation between the error estimate and the number of iteration can be observed. The Newton-Raphson method on the other hand shows a quadratic behaviour.

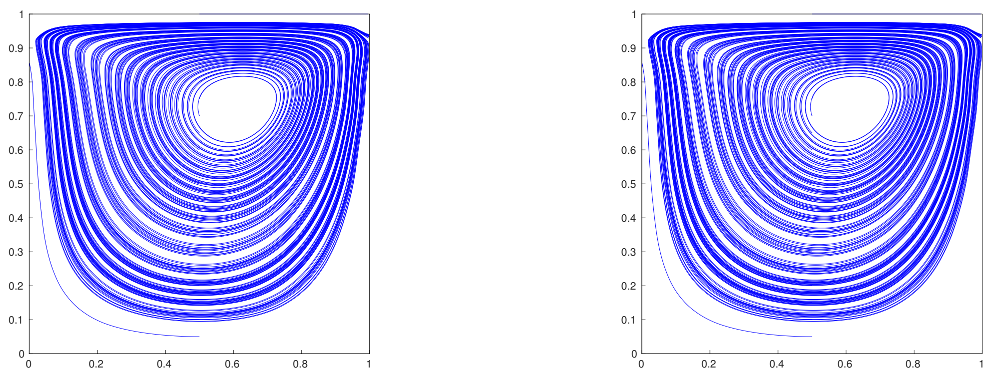


Figure 1: Streamlines for the Q2Q1 element obtained with Piccard's (left) and Newton-Raphson (right) scheme

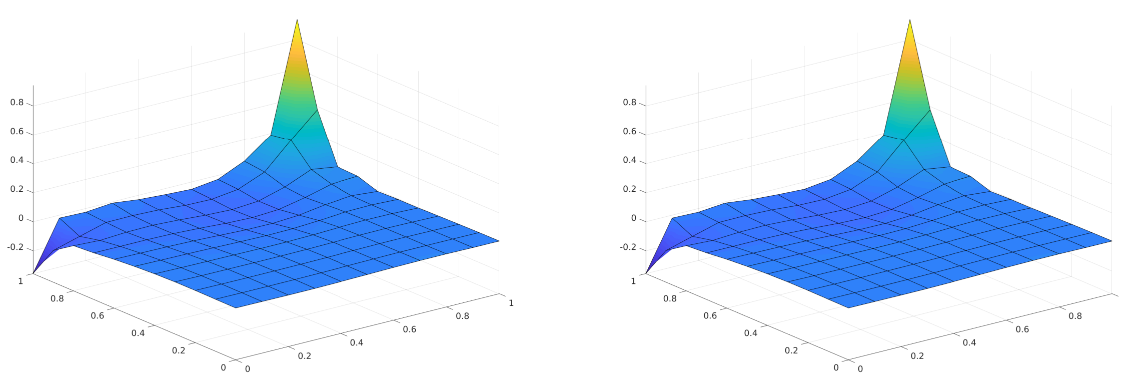
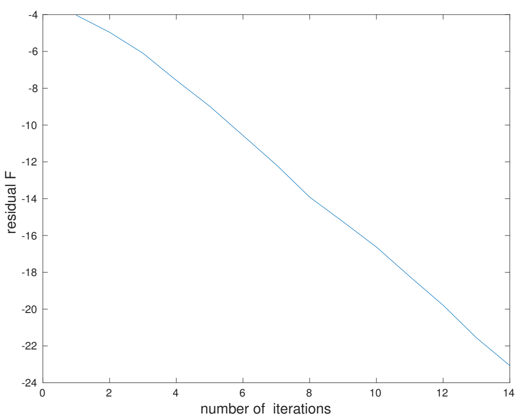


Figure 2: Streamlines for the Q2Q1 element obtained with Piccard's (left) and Newton-Raphson (right) scheme



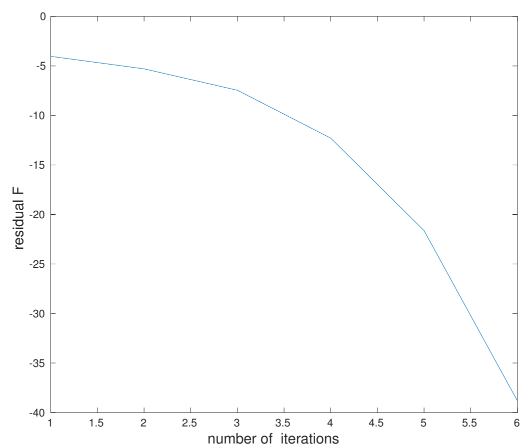


Figure 3: Convergence plots for the Piccard's scheme (left) and the Newton-Raphson scheme (right)