Interdisziplinäres Zentrum
für Wissenschaftliches Rechnen (IWR)
Ruprecht–Karls–Universität Heidelberg

Prof. Dr. Vincent Heuveline
Dr. Philipp Gerstner

Exercise Sheet 10

30.06.2021 – 09.07.2021

Computational Fluid Dynamics (Summer Term 2021)

Exercise 1 (Programming). Variants of Time Discretizations (15 Points)
Consider the Navier-Stokes equations:

$$\begin{aligned} \partial_t \mathbf{v} - \nu \Delta \mathbf{v} + (\mathbf{v} \cdot \nabla)\mathbf{v} + \nabla p = \mathbf{f} \;\; &\text{in } \Omega := [0,1] \times [0,1] \\ \nabla \cdot \mathbf{v} = 0 \;\; &\text{in } \Omega \\ \mathbf{v} = 0 \;\; &\text{on } \partial\Omega. \end{aligned}$$

Here, the right-hand sides $\mathbf{f}$, g are chosen such that

$$\begin{aligned} \mathbf{v}_x^*(x,y,t) &= \sin(10\pi t)\sin(\pi x)^2 \sin(2\pi y) \\ \mathbf{v}_y^*(x,y,t) &= -\sin(10\pi t)\sin(2\pi x)\sin(\pi y)^2 \\ p^*(x,y,t) &= \sin(10\pi t)\cos(2\pi y) \end{aligned}$$

is a classical solution, i.e. $f(x,y) := \partial_t \mathbf{v} - \nu \Delta \mathbf{v} + (\mathbf{v} \cdot \nabla)\mathbf{v} + \nabla p$.

The goal of this exercise is to investigate different ways to discretize this system in time and different nonlinear solvers.

Variant a: Semi-implicit θ scheme : for $n = 1, \ldots, N$:

$$\begin{aligned} \frac{1}{k}(\mathbf{v}_n - \mathbf{v}_{n-1}) + \theta(-\nu\Delta\mathbf{v}_n + (\mathbf{v}_{n-1} \cdot \nabla)\mathbf{v}_n)& \\ +(1-\theta)(-\nu\Delta\mathbf{v}_{n-1} + (\mathbf{v}_{n-1} \cdot \nabla)\mathbf{v}_{n-1}) + \nabla p_n &= \theta\mathbf{f}(t_n) + (1-\theta)\mathbf{f}(t_n) \;\; \text{in } \Omega \\ \nabla \cdot \mathbf{v}_n &= 0 \;\; \text{in } \Omega \\ \mathbf{v}_n &= 0 \;\; \text{on } \partial\Omega \\ \mathbf{v}_0 &= 0 \end{aligned} \tag{1}$$

which can be written as linear system of the form

$$A_{semi}(\mathbf{v}_{n-1})X = b_n \tag{2}$$

Variant b: Fully-implicit θ scheme : for $n = 1, \ldots, N$:

$$\begin{aligned} \frac{1}{k}(\mathbf{v}_n - \mathbf{v}_{n-1}) + \theta(-\nu\Delta\mathbf{v}_n + (\mathbf{v}_n \cdot \nabla)\mathbf{v}_n)& \\ +(1-\theta)(-\nu\Delta\mathbf{v}_{n-1} + (\mathbf{v}_{n-1} \cdot \nabla)\mathbf{v}_{n-1}) + \nabla p_n &= \theta\mathbf{f}(t_n) + (1-\theta)\mathbf{f}(t_n) \;\; \text{in } \Omega \\ \nabla \cdot \mathbf{v}_n &= 0 \;\; \text{in } \Omega \\ \mathbf{v}_n &= 0 \;\; \text{on } \partial\Omega \\ \mathbf{v}_0 &= 0 \end{aligned} \tag{3}$$

which can be written as nonlinear system (to be solved for $\mathbf{v}_n$) of the form

$$F(\mathbf{v}_{n-1}, \mathbf{v}_n) = 0. \tag{4}$$

System (4) can either be solved by means of Newton's method (**variant b1**), see the previous exercise sheet or by the Picard iteration (**variant b2**):

a. given $\mathbf{v}_{n-1}$, K, ϵ

b. set $\mathbf{v}^{(0)} = \mathbf{v}_{n-1}$

c. compute initial residual $r_0 = \|F(\mathbf{v}_{n-1}, \mathbf{v}^{(0)})\|$

d. set $k = 1$

e. do until $k = K + 1$ or $r_k < \epsilon r_0$:

(a) compute $(\mathbf{v}^{(k)}, p^{(k)})$ as solution of

$$\begin{aligned}
\frac{1}{k}(\mathbf{v}^{(k)} - \mathbf{v}_{n-1}) + \theta(-\nu\Delta\mathbf{v}^{(k)} + (\mathbf{v}^{(k-1)} \cdot \nabla)\mathbf{v}^{(k)}) & \\
+(1-\theta)(-\nu\Delta\mathbf{v}_{n-1} + (\mathbf{v}_{n-1} \cdot \nabla)\mathbf{v}_{n-1}) + \nabla p^{(k)} &= \theta\mathbf{f}(t_n) + (1-\theta)\mathbf{f}(t_n) \;\text{ in } \Omega \\
\nabla \cdot \mathbf{v}^{(k)} &= 0 \;\text{ in } \Omega \\
\mathbf{v}^{(k)} &= 0 \;\text{ on } \partial\Omega \\
\mathbf{v}_0 &= 0
\end{aligned} \tag{5}$$

which can be written as linear system of the form

$$A_{picard}(\mathbf{v}^{(k-1)})X = b_n \tag{6}$$

(b) compute new residual $r_k = \|F(\mathbf{v}_{n-1}, \mathbf{v}^{(k)})\|$

(c) set $k = k + 1$

a. Modify the assembly routines of the class LocalFlowAssembler at the locations marked by TODO exercise A in ex9_CavityNavierStokes.h to compute the matrix and vector corresponding to (2).

b. Modify the assembly routines of the class LocalFlowAssembler at the locations marked by TODO exercise B in ex9_CavityNavierStokes.h to compute the matrix and vector corresponding to (6).

c. Implement the Picard iteration at the location marked by TODO exercise B2 in ex9_CavityNavierStokes.cc

d. Modify the assembly routines of the class LocalFlowAssembler at the locations marked by TODO exercise C in ex9_CavityNavierStokes.h to compute F and $D_{(\mathbf{v},p)}F$, used in Newton's method. See exercise sheet 9 for details.

e. The programm computes the following errors for each of the three solvers:

$$e_{H1,\mathbf{v}} := \left(k \sum_{n=1}^{k^{-1}} \|\nabla(\mathbf{v}^* - \mathbf{v}_h)\|^2 \right)^{\frac{1}{2}}$$

$$e_{L2,p} := \left(k \sum_{n=1}^{k^{-1}} \|(p^* - p_h)\|^2 \right)^{\frac{1}{2}}$$

Run the code on up to 8 MPI processes for all presented methods. Vary the time step size k, the mesh refinement level and θ (all parameters in the xml file) to see how the errors depend on the discretization parameters.

The corresponding code framework is ex9_CavityNavierStokes. Don't forget to the modify the file exercises/CMakeLists.txt accordingly, see the first exercise sheet for details.

Submission: until 09.07.2021, 11 am, moodle upload