HW2 - REPORT

Unsteady convection problems

Finite Element in Fluids

Álvaro Rodríguez Luis

1. Introduction

The equation we want to solve to solve is:

$$\begin{cases} u_t + a u_x = 0 & x \in (0,1),\ t \in (0, 0.6] \\ u(x,0) = u_0(x) & x \in (0,1) \\ u(0,t) = 1 & t \in (0, 0.6] \end{cases}$$

$$u_0(x) = \begin{cases} 1 & \text{if } x \leq 0.2, \\ 0 & \text{otherwise} \end{cases}$$

$$a = 1, \Delta x = 2 \cdot 10^{-2}, \Delta t = 1.5 \cdot 10^{-2}$$

We have to solve it with linear finite element for the Galerkin scheme in space and the following methods in time:

1) Crank-Nicholson scheme. (CN)
2) Second-order Lax-Wendorff method. (TG2)
3) Third-order explicit Taylor-Galerkin method. (TG3)

2. Code modifications

In order to follow the instructions of the report, presented in the introduction, some modifications where made to the provided code, in the function **System.m**, and are shown in the Annex.

In this function we need to provide matrices A and B to solve the finite element method system:

$$\mathrm{A} \cdot \Delta \mathrm{u} \;=\; \mathrm{f} \;+\; \mathrm{B} \cdot \mathrm{u}$$

Using the FEM matrices M, C and K:

$$M_{\mathrm{ab}} \;=\; \int N_a N_b dx\,, \qquad \mathrm{C}_{ab} \;=\; \int N_a \frac{\partial N_b}{\partial \mathrm{x}} dx\,, \qquad \mathrm{K}_{ab} \;=\; \int \frac{\partial N_{\mathrm{a}}}{\partial \mathrm{x}} \frac{\partial N_b}{\partial \mathrm{x}} dx$$

This is already done for the CN method in the class notes and it was provided in the given function. CN scheme reads:

$$\frac{\Delta \mathrm{u}}{\Delta \mathrm{t}} + \frac{1}{2}(a \cdot \nabla)\Delta u = \frac{1}{2}(s^{n+1} + s^n) - a \cdot \nabla u^n$$

Where s is the source term, zero in this case. Using weighted residual method, with Galerkin formulation:

$$\left(\omega, \frac{\Delta u}{\Delta t}\right) + \frac{1}{2}(\omega, (a \cdot \nabla)\Delta u) = \frac{1}{2}(\omega, s^{n+1} + s^n) - (\omega, a \cdot \nabla u^n)$$

The FEM system would read:

$$\left(\frac{1}{\Delta t} M + \frac{1}{2} \cdot \mathrm{a} \cdot C\right) \Delta u = f - \mathrm{a} \cdot C u^n$$

And then $\mathrm{A} \;= \frac{1}{\Delta t} M + \frac{1}{2} \cdot \mathrm{a} \cdot C$ and $\mathrm{B} \;= -\mathrm{a} \cdot C$. To modify the code, I proceeded in a similar way for TG2 and TG3.

TG2 method reads:

$$\frac{\Delta \mathrm{u}}{\Delta \mathrm{t}} = s^n + \frac{\Delta t}{2}(s_t^n - a \cdot \nabla s^n) - a \cdot \nabla u^n + \frac{\Delta t}{2}(a \cdot \nabla)^2 u^n$$

Using weighted residual method, with Galerkin formulation, integration by parts and the boundary conditions:

$$\left(\omega, \frac{\Delta u}{\Delta t}\right) = \left(\omega, s^n \;+ \frac{\Delta t}{2}(s_t^n - a \cdot \nabla s^n)\right) - (\omega, a \cdot \nabla u^n) \text{ - } \frac{\Delta t}{2} \cdot (a \cdot \nabla \omega, (a \cdot \nabla) u^n\;)$$

The FEM system would read:

$$\frac{1}{\Delta t} M \Delta u = f - a \cdot C u^n - a^2 \cdot \frac{\Delta t}{2} \cdot K u^n$$

And then $\mathrm{A} \;= \frac{1}{\Delta t} M$ and $\mathrm{B} \;= -\mathrm{a} \cdot C - a^2 \cdot \frac{\Delta t}{2} \cdot K$.

TG3 reads:

$$\left[1-\frac{\Delta t^2}{6}(a\cdot\nabla)^2\right]\frac{\Delta \mathrm{u}}{\Delta \mathrm{t}}$$
$$=s^n+\frac{\Delta t}{2}(s_t^n-a\cdot\nabla s^n)+\frac{\Delta t^2}{6}(s_{\mathrm{tt}}^n-a\cdot\nabla {s_t}^n)-a\cdot\nabla u$$
$$+\frac{\Delta t}{2}(a\cdot\nabla)^2u^n$$

Using weighted residual method, with Galerkin formulation, integration by parts and the boundary conditions:

$$\left(\omega,\frac{\Delta u}{\Delta t}\right)-\frac{\Delta t^2}{6}\cdot\left(a\cdot\nabla\omega,(a\cdot\nabla)\frac{\Delta u}{\Delta t}\right)$$
$$=\left(\omega,s^n+\frac{\Delta t}{2}(s_t^n-a\cdot\nabla s^n)+\frac{\Delta t^2}{6}(s_{\mathrm{tt}}^n-a\cdot\nabla {s_t}^n)\right)-(\omega,a\cdot\nabla u^n)$$
$$-\frac{\Delta t}{2}\cdot(a\cdot\nabla\omega,(a\cdot\nabla)u^n)$$

The FEM system would read:

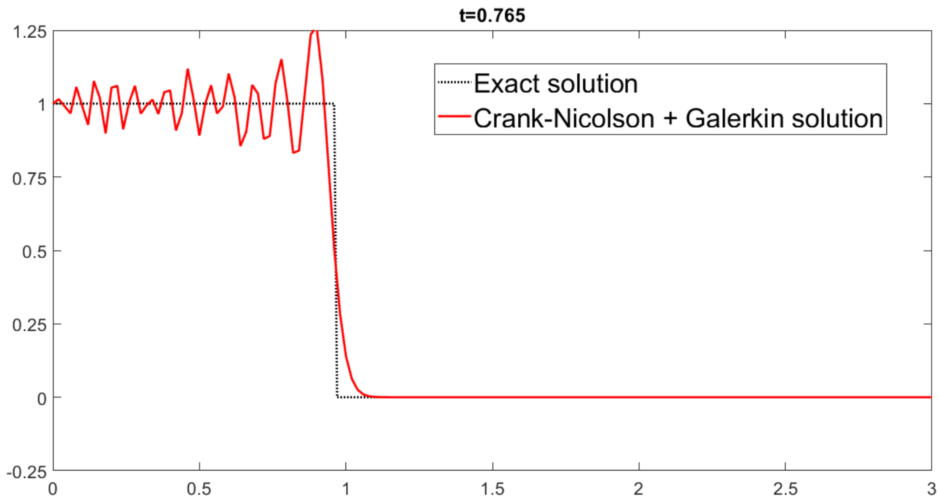
$$\left(\frac{1}{\Delta t}M+\frac{\Delta t^2}{6}\cdot a^2\cdot K\right)\Delta u=f-a\cdot Cu^n-a^2\cdot\frac{\Delta t}{2}\cdot Ku^n$$

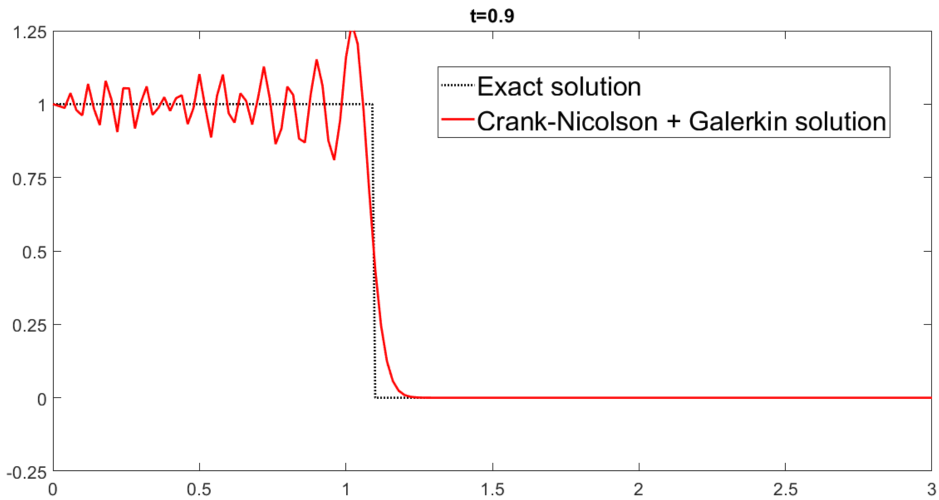
And then $\mathrm{A}=\frac{1}{\Delta t}M+\frac{\Delta t^2}{6}\cdot a^2\cdot K$ and $\mathrm{B}=-\mathrm{a}\cdot C-a^2\cdot\frac{\Delta t}{2}\cdot K$.

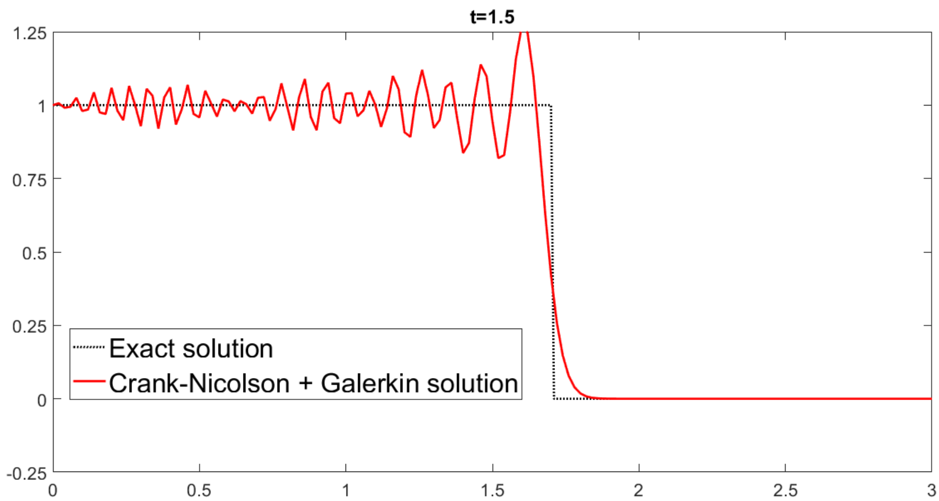
In the implementation, both sides of the equations are multiplied by Δt, obtaining the formulas seen in the Annex.

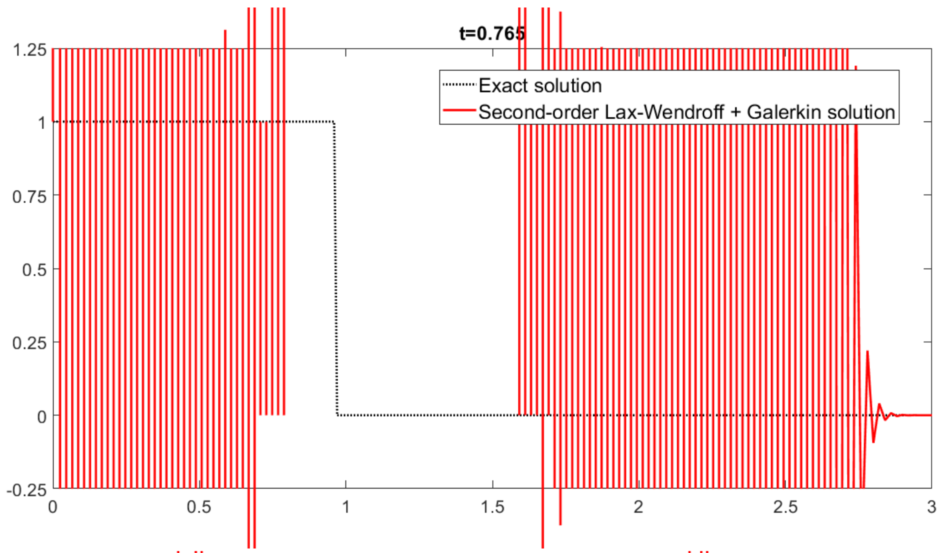
3. Results

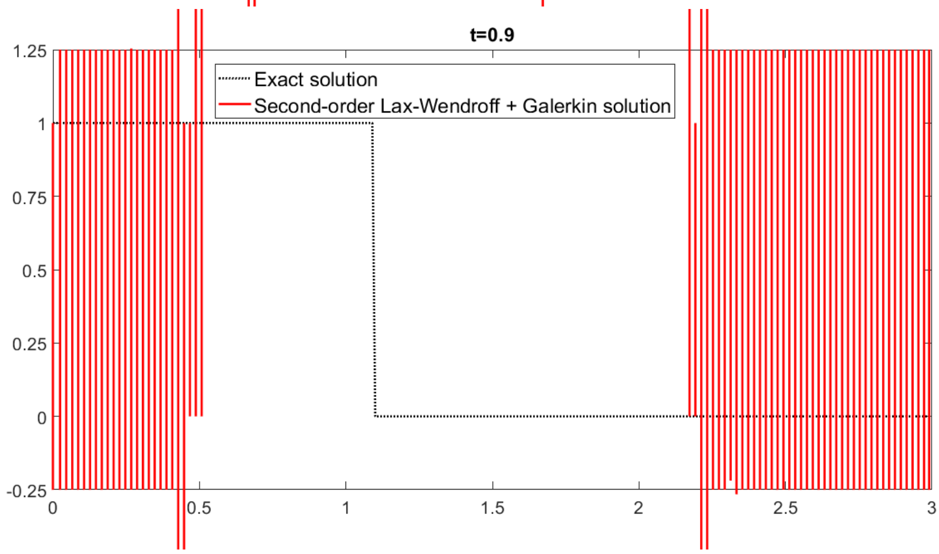
For the proposed problem, the Courant number was $C=a\cdot\Delta t/\Delta x=1.5/2=0.75$, and then $C^2=0.5625$. From the class notes, we know that CN is unconditionally stable, that TG2 stability limit is $C^2=1/3$ and that TG3 stability limit is $C^2=1$. We can predict that with the Courant number of the proposed number, both CN and TG3 methods will be stable, but TG2 will be unstable. For TG2 to be stable we would need a smaller time step or a bigger element size in order to obtain stability. One could also expect the TG3 method to be more precise that the CN method, since the order of convergence in higher. In the following figures, the obtained results are shown. All the results are compatible with what was expected.

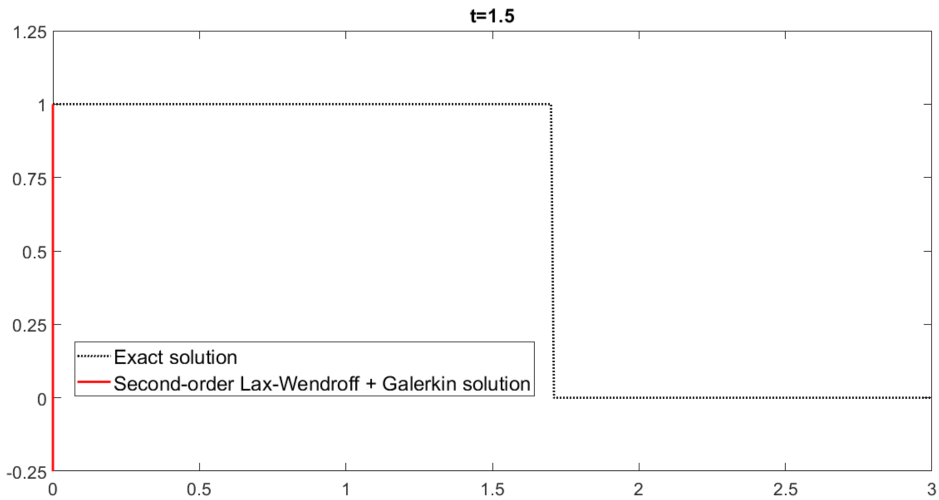

t=0.765
1.25
1
0.75
0.5
0.25
0
-0.25
0
0.5
1
1.5
2
2.5
3
Exact solution
Crank-Nicolson + Galerkin solution

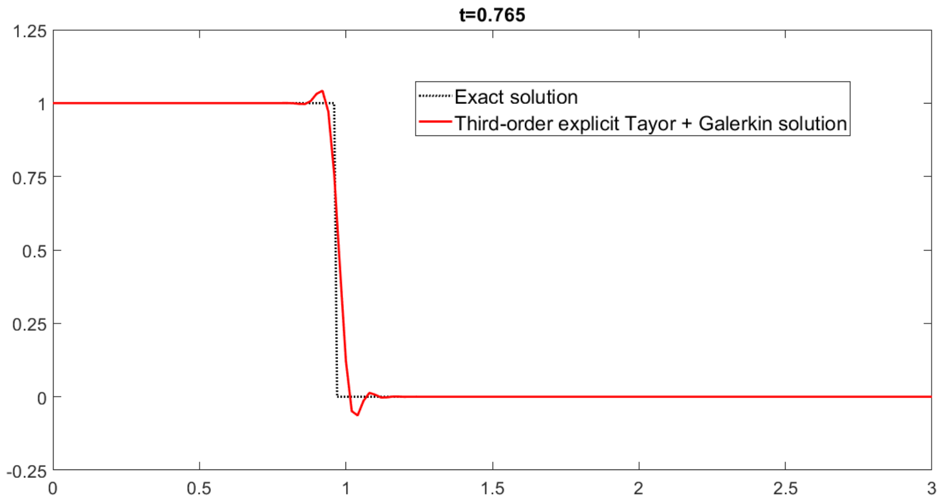

t=0.9
1.25
1
0.75
0.5
0.25
0
-0.25
0
0.5
1
1.5
2
2.5
3
Exact solution
Crank-Nicolson + Galerkin solution

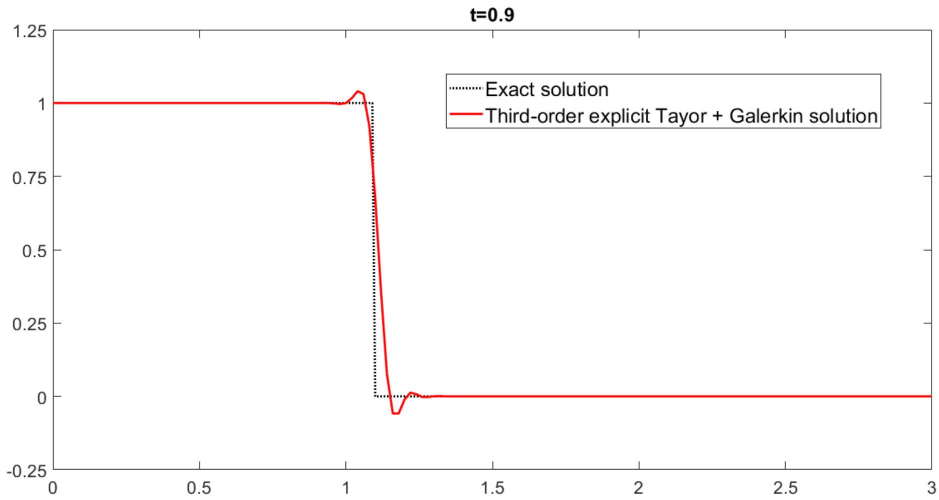

t=1.5
1.25
1
0.75
0.5
0.25
0
-0.25
0
0.5
1
1.5
2
2.5
3
Exact solution
Crank-Nicolson + Galerkin solution

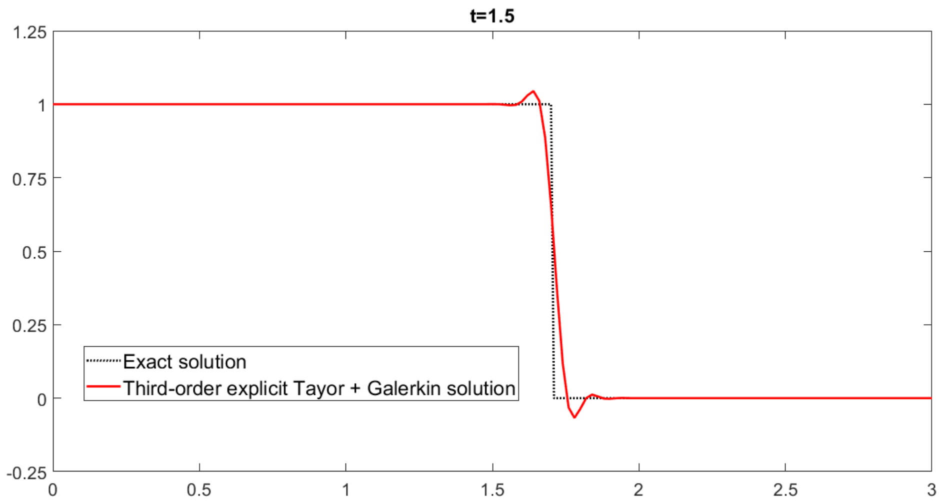

t=0.765
Exact solution
Second-order Lax-Wendroff + Galerkin solution
1.25
1
0.75
0.5
0.25
0
-0.25
0
0.5
1
1.5
2
2.5
3


t=0.9
Exact solution
Second-order Lax-Wendroff + Galerkin solution
1.25
1
0.75
0.5
0.25
0
-0.25
0
0.5
1
1.5
2
2.5
3


t=1.5
Exact solution
Second-order Lax-Wendroff + Galerkin solution
1.25
1
0.75
0.5
0.25
0
-0.25
0
0.5
1
1.5
2
2.5
3


t=0.765
Exact solution
Third-order explicit Tayor + Galerkin solution
1.25
1
0.75
0.5
0.25
0
-0.25
0
0.5
1
1.5
2
2.5
3


t=0.9
Exact solution
Third-order explicit Tayor + Galerkin solution
1.25
1
0.75
0.5
0.25
0
-0.25
0
0.5
1
1.5
2
2.5
3


t=1.5
Exact solution
Third-order explicit Tayor + Galerkin solution
1.25
1
0.75
0.5
0.25
0
-0.25
0
0.5
1
1.5
2
2.5
3

Annex

```
function [A,B,methodName] = System(method,M,K,C,a,dt)
switch method
    case 1 % Crank-Nicolson + Galerkin
        A = M + 1/2*a*dt*C;
        B = -a*dt*C;
        methodName = 'Crank-Nicolson + Galerkin';
    case 2 % Second-order Lax-Wendroff + Galerkin
        A = M;
        B = -a*C*dt - 1/2*a^2*dt^2*K;
        methodName = 'Second-order Lax-Wendroff + Galerkin';
    case 3 % Third-order explicit Tayor-Galerkin
        A = M + 1/6*a^2*dt^2*K;
        B = -a*C*dt - 1/2*a^2*dt^2*K;
        methodName = 'Third-order explicit Tayor + Galerkin';
    otherwise
        error('not available method')
end
```