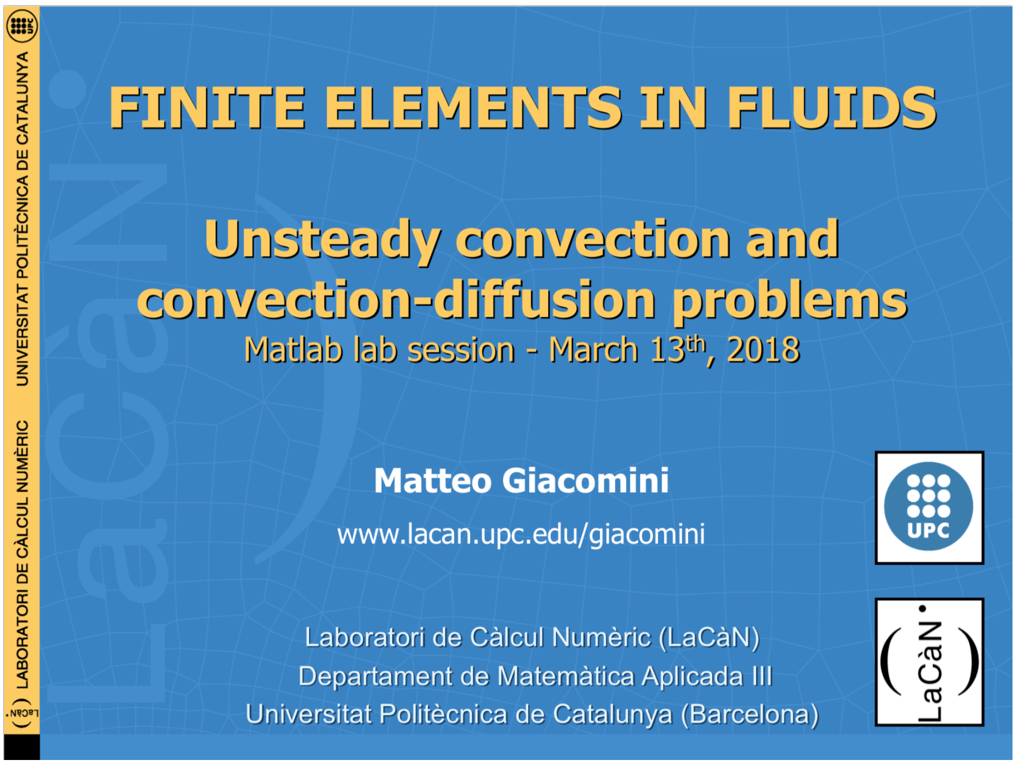


Unsteady Convection & Diffusion

NUMERICAL EXAMPLES

Iberico Leonardo, Juan Diego | Finite Elements in Fluids | 02/04/2018

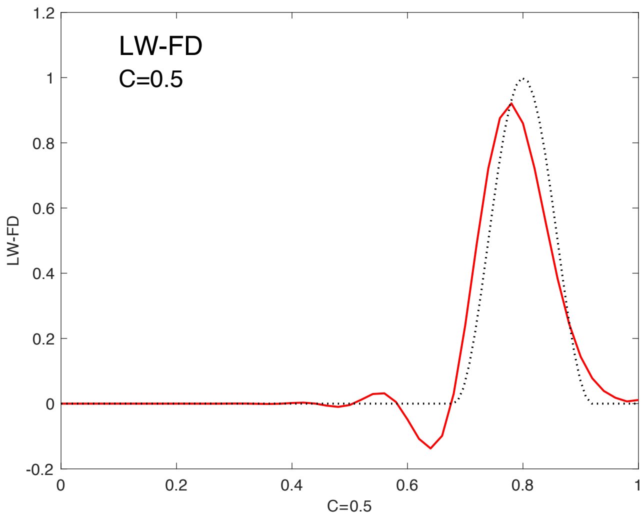
Example: Pure convection case

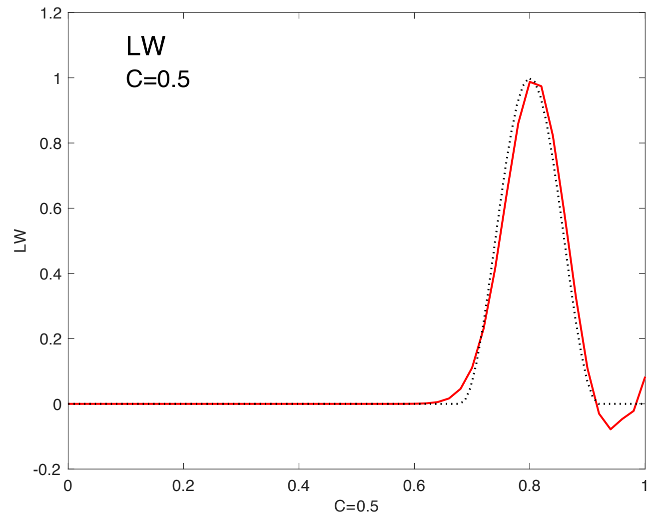
The following report it is focus unsteady convection problem. The aim of the document it is to show the different behavior of the available schemes (time and space integrators), to solve a pure transport problem.

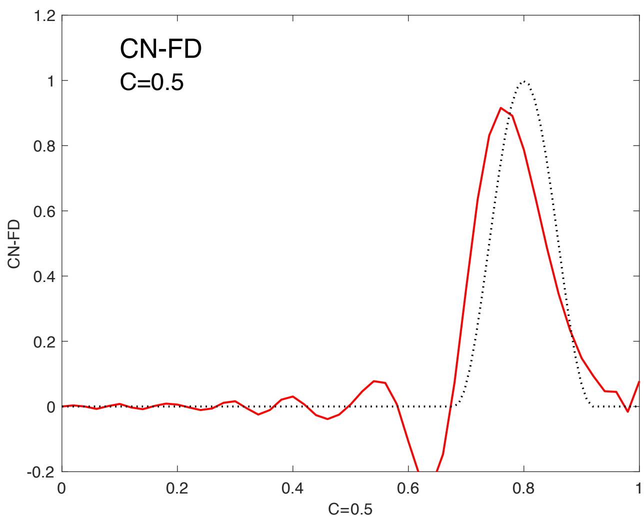
The solution is for 1D dimension, where the initial condition is propagation of a cosine profile.

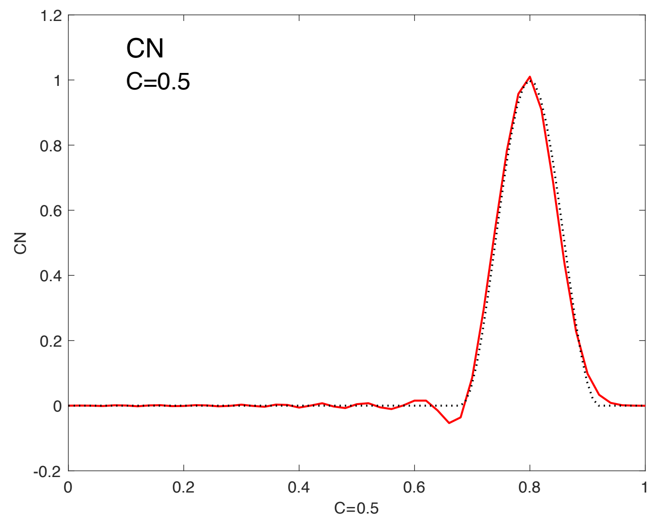
EXERCISE FOR UNSTEADY CONVECTION

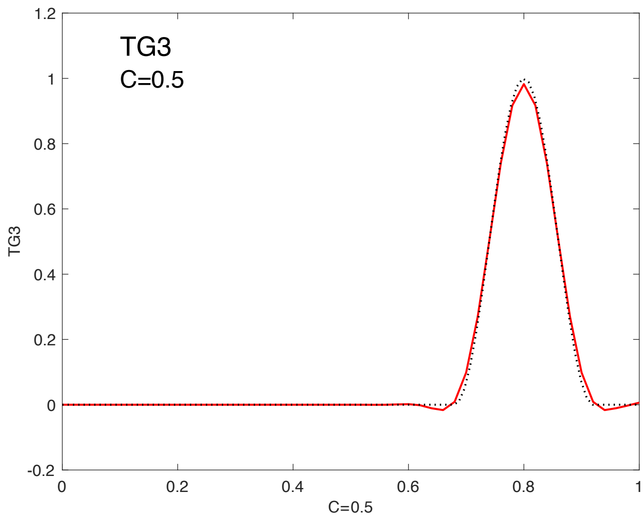
- Comment the numerical experiments (stability and accuracy).

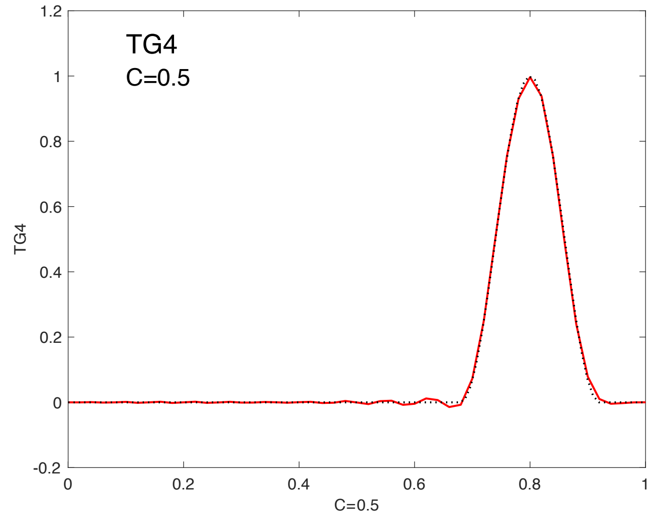




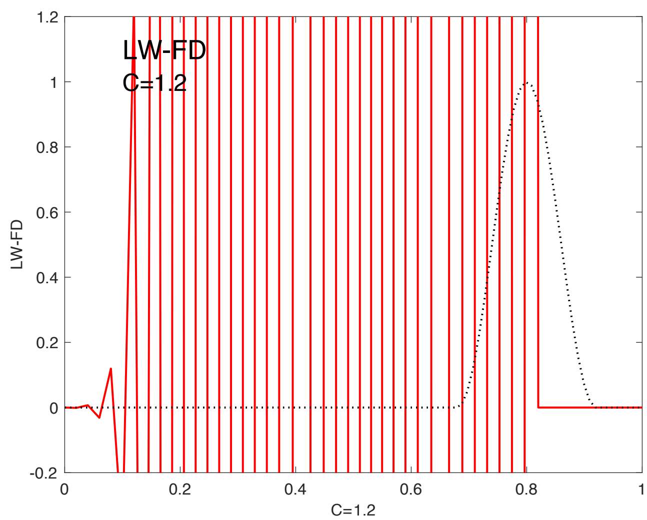


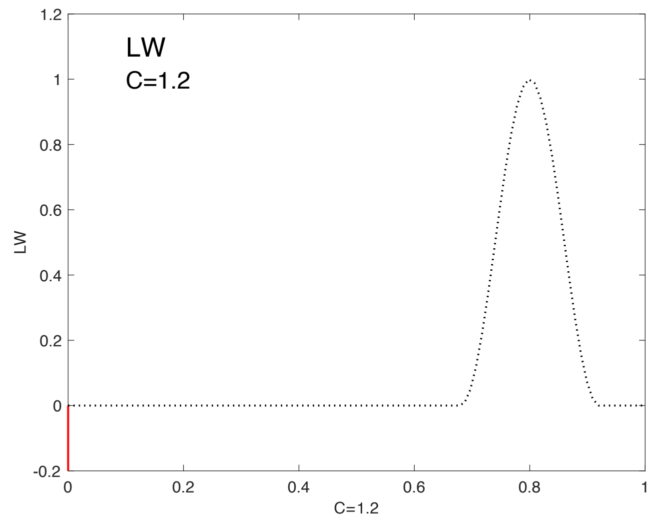


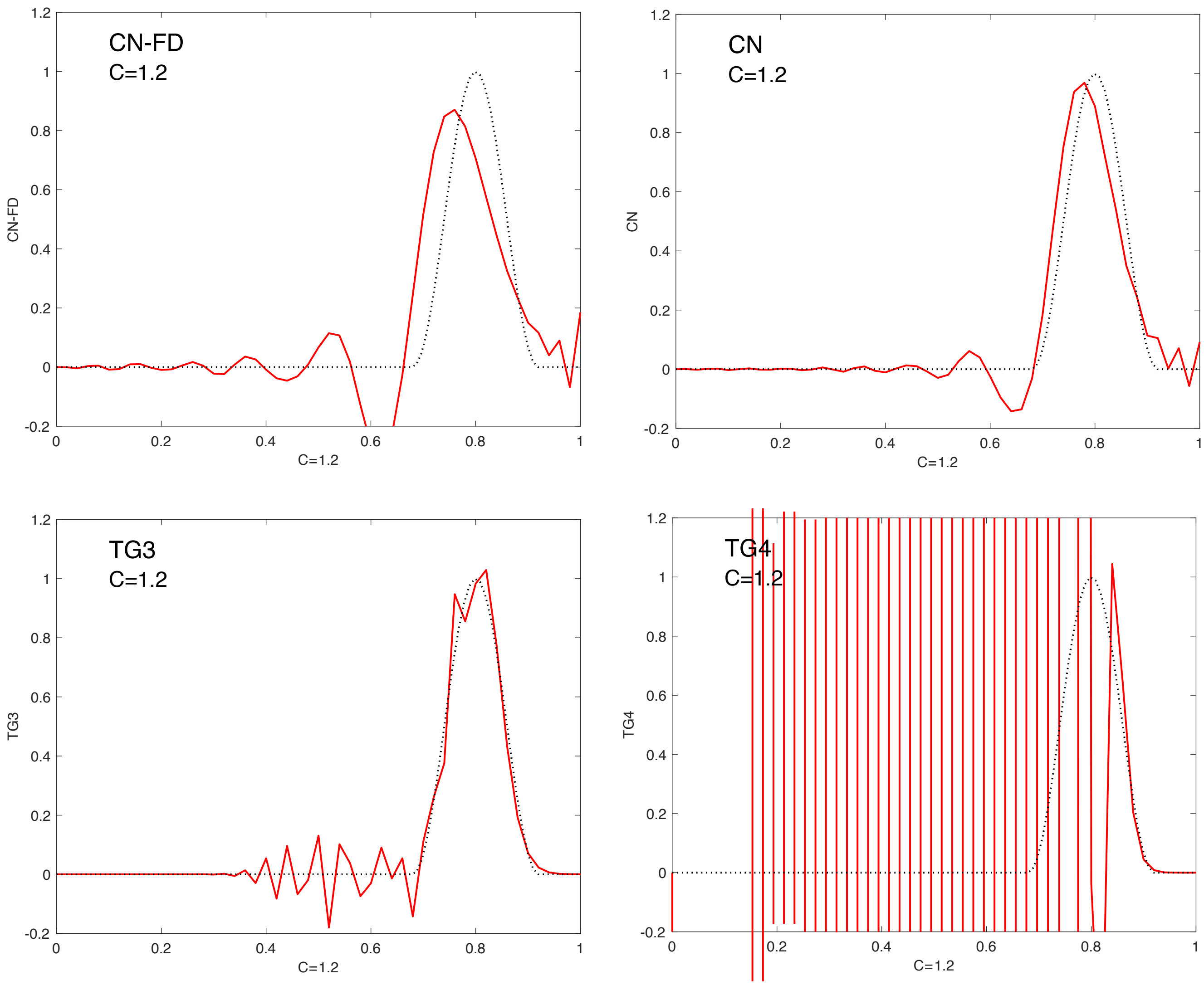




All the time integrators schemes already implemented were tested with courant number 0.5. As it can be seen the methods where the lumped mass matrix introduces some numerical error in order to increase the range of stability. On the other side, the higher order accuracy shows an admissible result for the pure transport equation.







The same problem it was tested, but with a higher courant number equal to 1.2. In this case, as it is known, for Lax-Wendroff schemes we are out of the stability range. On the other side, the Crank-Nicolson is able to manage the stability for high values of courant numbers although the accuracy on the solution decrease. As a remark, Crank-Nicolson with lumped mass, introduce some numerical errors compared with CN keeping as an advance the computational cost.

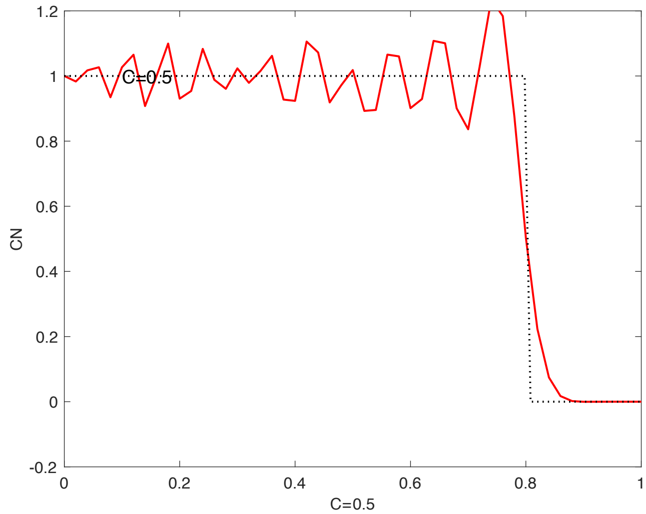
Propagation of a step front:

The implementation of the new initial condition as also the evolution of the exact solution had been done on the main.m file. On the following lines it is presented the implementation for the propagation of a step front.

```matlab
% INITIAL CONDITION FOR THE TRANSIENT ANALYSIS
% Cosine profile
u = zeros(numnp, nstep+1);
x0 = 0.2;
sigma = 0.12;
for i=1:numnp
    %dist = xnode(i)-x0;
    if xnode(i) <= x0
        u(i,1) = 1;
    end
end
```

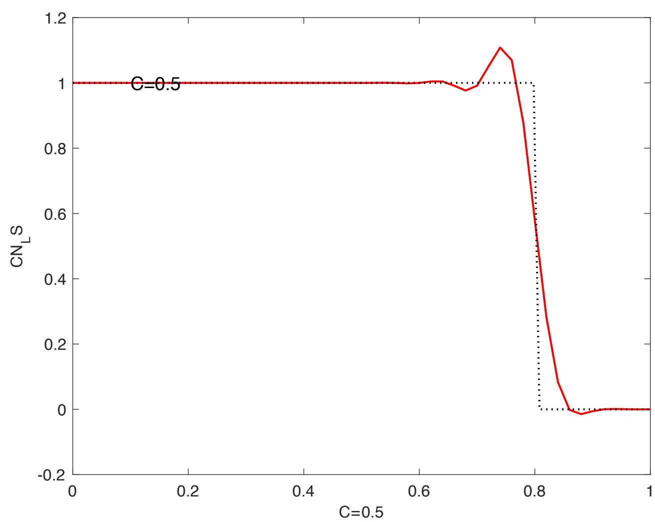
```matlab
% POSTPROCESS
% Exact solucion at time t = t_end;
x = linspace(0,1);
u_ex = zeros(1,100);
xe = x0 + a*t_end;
for i=1:100
    %dist = x(i)-xe;
    if x(i) <= xe
        u_ex(i) = 1;
    end
end
```

- Solve the problem using the Crank-Nicholson scheme in time and linear finite element for the Galerkin scheme in space. Is the solution accurate?



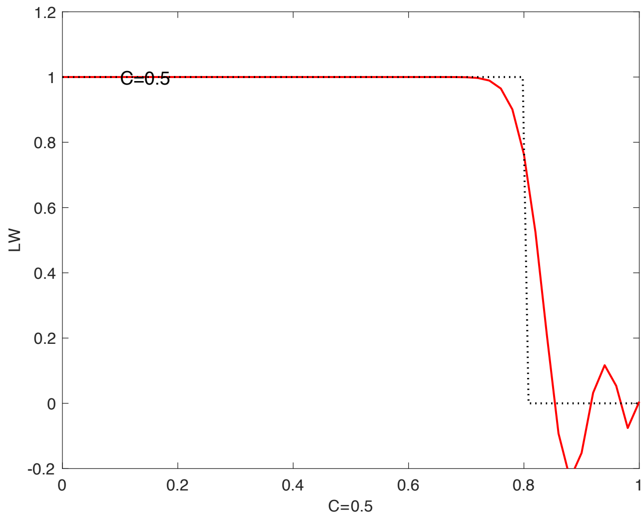
As it can be seen the accuracy of Crank Nicholson scheme has decreased, meanwhile it is keeping stable. That behavior is due to the step exact solution.

- Implement the Crank Nicholson scheme in time and the least-squares formulation in space. Comment the results.



Now, due to the change of the integration on the space discretization from Galerkin to Least Squares we reach a better accuracy with the time integrator Crank-Nicholson.

- Solve the problem using second-order Lax-Wendroff method. Can we expect the solution to be accurate? If not, what changes are necessary? Comment the results.

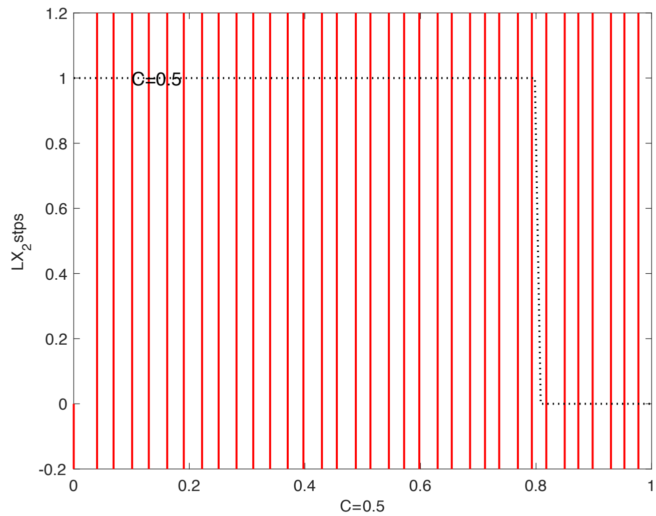


The solution from Lax-Wendroff we can say it has better accuracy than CN with Galerkin spatial discretization. But, compared with the Least Squares, the accuracy of Lax-Wendroff is worst on the down of the step front,

- Implement the second-order two-step Lax-Wendroff method. Comment the results.

```
    % ----------- Defining the step u_(n+0.5) ------------------------
    A(isp,isp) = A(isp,isp) + w_ig*(N'*N);
    B(isp,isp) = B(isp,isp) + w_ig*(N'*N - dt_2*a*(N)'*Nx);
    % ----------- Defining the step u_(n+1) --------------------------
    A_2(isp,isp) = A_2(isp,isp) + w_ig*(N)'*N;
    B_2(isp,isp) = B_2(isp,isp) + w_ig*(-dt*a*(Nx)'*N);
    f(isp) = f(isp) + w_ig*(N')*SourceTerm(x);
end

    if meth == 8 % Dedice the method
     % SOLUTION AT EACH TIME STEP
        for n = 1:nstep
           btot = [B*u(:,n)+f; bccd];
           u_hlfstp  = U\(L\btot);
           b_2tot = [A_2*u(:,n) + B_2*u_hlfstp(1:numnp)+f; bccd];
           aux = Atot\b_2tot;
           u(:,n+1) = aux(1:numnp);
        end
```



According to the implementation done and the supplied code, it couldn't be reached the correct solution.