Universitat Politècnica de Catalunya
Master of Science in Computational Mechanics
Finite Element in Fluids
FEF
Spring Semester 2017/2018

MatLab Session 4
Usteady Advection Equation and
Unsteady Advection-Diffusion Equation

Luan Malikoski Vieira

April 02, 2018

1 Introduction

This report will cover the 1D *usteady state advection* equation and 1D *usteady state advection-diffusion* equation, seen in (1) and (2) respectively. The first will be studied by solving the propagation of a *steep front* problem which different time and space discretization methods, as requested. The second will be studied by solving the *Gaussian Hill* problem with different time discretization methods.

$$u_t + au_x = s \tag{1}$$

$$u_t + au_x - \nu u_{xx} = s \tag{2}$$

2 Unsteady Advection Equation

2.1 Changes in Matlab Routines

This subsection will briefly cover the changes in the MatLab routines in order to implement each time/space discretization method.

2.1.1 Implementation of Crank-Nicholson + Least Square

The least square WRM formulation of the Crank-Nicholson method in scalar reduced form, applied to equation (1), can be written as:

$$\left(w + \frac{\Delta t}{2} w_x, \Delta u + \frac{a\Delta t}{2}\Delta u_x\right) = -a\Delta t\left(w + \frac{\Delta t}{2} w_x, u_x^n\right) + \Delta t\left(w + \frac{\Delta t}{2} w_x, s^n\right) \tag{3}$$

Applying the Galerkin formulation ($w_i = N_i$ and $u = \sum_i u_j N_j$), we have:

$$\left(N_i + \frac{\Delta t}{2}\frac{dN_i}{dx}, N_j + \frac{a\Delta t}{2}\frac{dN_j}{dx}\right)\Delta u = -a\Delta t\left(N_i + \frac{\Delta t}{2}\frac{dN_i}{dx}, \frac{dN_j}{dx}\right)u^n + \Delta t\left(N_i + \frac{\Delta t}{2}\frac{dN_i}{dx}, s^n\right) \tag{4}$$

The code implementation of the matrices to of the equation $A\Delta u = Bu + f$ becomes:

```
% Matrices assembly
A(isp,isp) = A(isp,isp) + w_ig*(N+dt_2*a*Nx)'*(N+dt_2*a*Nx);
B(isp,isp) = B(isp,isp) -w_ig*(N+dt_2*a*Nx)'*(dt*a*Nx);
f(isp) = f(isp) + w_ig*(N+dt_2*a*Nx)'*dt*SourceTerm(x);
```

2.1.2 Implementation of TG2-2s + Galerkin

The two steps second order Lax-Wendroff method (Richtnger scheme) is defined by the following equations for each step.

$$u^{n+\frac{1}{2}} = u^n + \frac{\Delta t}{2}u_t^n \tag{5}$$

$$u^{n+1} = u^n + \Delta t u_t^{n+\frac{1}{2}} \tag{6}$$

Using the governing equation (1), those steps can be further written as:

$$\Delta\tilde{u} = \frac{\Delta t}{2}(s^n - au_x^n) \tag{7}$$

$$\Delta u = \Delta t(s^{n+\frac{1}{2}} - au_x^{n+\frac{1}{2}}) \tag{8}$$

The application of the WRM followed by integration by parts in previous equations leads to:

$$(w, \Delta\tilde{u}) = \frac{a\Delta t}{2}(w_x, u^n) + \frac{\Delta t}{2}(w, s^n) \tag{9}$$

$$(w, \Delta u) = a\Delta t(w_x, u^{n+\frac{1}{2}}) + \Delta t(w, s^{n+\frac{1}{2}}) \tag{10}$$

Applying the Galerkin formulation ($w_i = N_i$ and $u = \sum_i u_j N_j$), we have:

$$(N_i, N_j)\,\Delta\tilde{u} = \frac{a\Delta t}{2}\left(\frac{dN_i}{dx}, N_j\right)u^n + \frac{\Delta t}{2}(N_i, s^n) \tag{11}$$

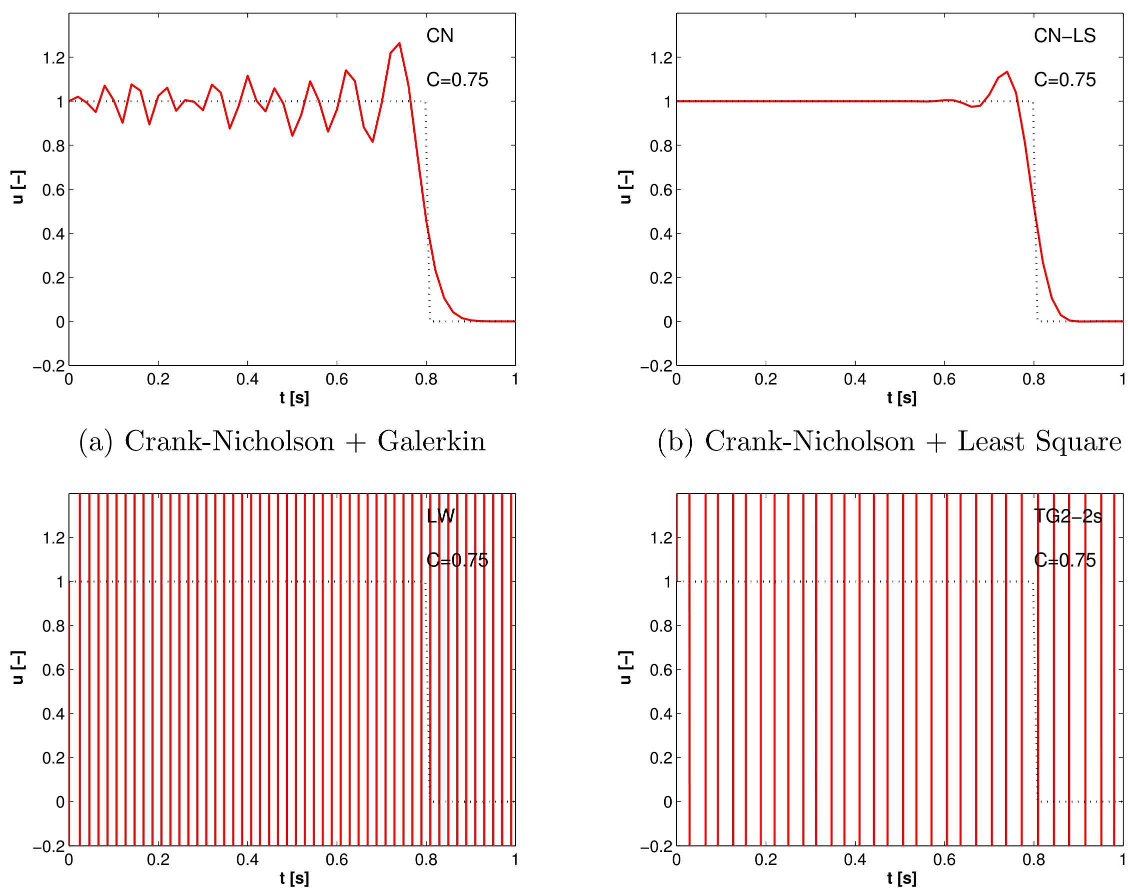
$$(N_i, N_j)\,\Delta u = a\Delta t\left(\frac{dN_i}{dx}, N_j\right)u^{n+\frac{1}{2}} + \Delta t\left(N_i, s^{n+\frac{1}{2}}\right) \tag{12}$$

The code implementation of the matrices of the equation $A\Delta u = Bu + f$, for the first and second steps, becomes:

```
% Matrices assembly
%FIRST STEP
A1(isp,isp) = A(isp,isp) + w_ig*N'*N;
B1(isp,isp) = B(isp,isp) + w_ig*dt_2*(a*Nx)'*N;
f1(isp) = f(isp) + w_ig*(N')*dt_2*SourceTerm(x);
%SECOND STEP
A(isp,isp) = A(isp,isp) + w_ig*N'*N;
B(isp,isp) = B(isp,isp) + w_ig*dt*(a*Nx)'*N;
f(isp) = f(isp) + w_ig*(N')*dt*SourceTerm(x);
```

2.2 Results

The unsteady advection of a steep front is solved with the following setup: $a = 1$, $\Delta x = 2.0 \times 10^{-2}$ and $\Delta t = 1.5 \times 10^{-2}$, which leads to a Courant number $C = 0.75$. The results for the Crank-Nicholson (CN), Crank-Nicholson with least square spatial discretization (CN-LS), Lax-Wendroff (LW) and two steps Lax-Wendroff(LW-2s) at the last time step ($t = 0.6s$) and C = 0.75 are shown in Figure (1).



(a) Crank-Nicholson + Galerkin

(b) Crank-Nicholson + Least Square

(c) Lax-Wendroff

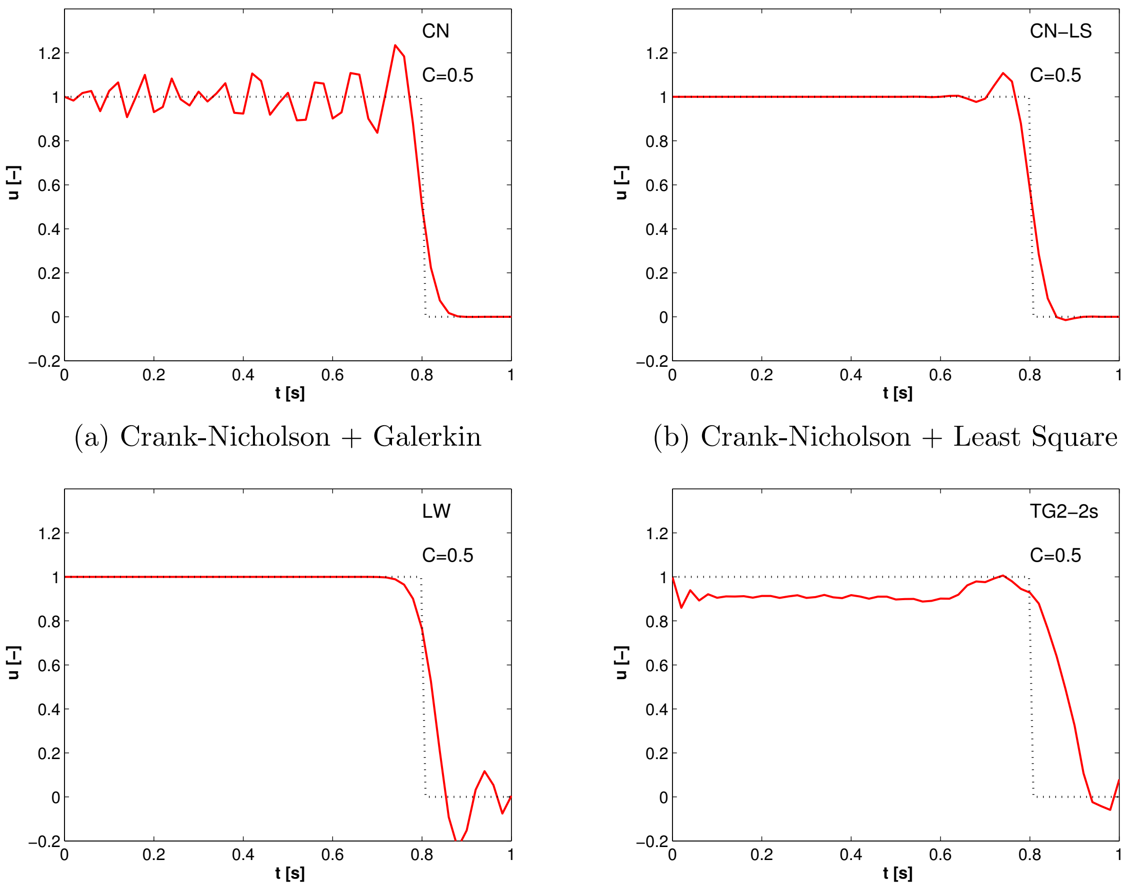
(d) Lax-Wendroff two steps

Figure 1: Steep front propagation problem: t =0.6 s and C = 0.75.

From the previous results it can be seen that for $C = 0.75$ the stability limit of LW and TG2-2s methods are reached. Also, the least-square spatial integration combined with the CN time integration successfully removes the spurious oscillations in the Galerkin formulation of the CN method.

The results for the Crank-Nicholson (CN), Crank-Nicholson with least square spatial discretization (CN-LS), Lax-Wendroff (LW) and two steps Lax-Wendroff(LW-2s) at the last time step ($t = 0.6s$) and C = 0.75 are shown in Figure (2).

When C is reduced, all four methos behave in a stable way. Same concluions can be drawn for CN and CN-LS formulatins. When comparing LW with its two steps counterpart (TG2-2s), its superior accuracy for this Courant number value is noticeable. As Courant is reduced the two-step formulation is expected to improved its amplification response



(a) Crank-Nicholson + Galerkin

(b) Crank-Nicholson + Least Square

(c) Lax-Wendroff

(d) Lax-Wendroff two steps

Figure 2: Steep front propagation problem: t =0.6 s and C = 0.50.

3 Unsteady Advection-Diffusion Equation

3.1 Implementation of Adam-Bashforth Method

The Adam-Bashforth is not an auto starting method, thus the first order Euler method is used in first step. Its WRM formulation, applied in the governing equation (2), has the following form:

$$(w, \Delta u) = -\Delta t \left[(w, au_x^n) + (w_x, \nu u_x^n)\right] + \Delta t(w, s^n) \tag{13}$$

After the first time-step, the Adam-Bashforth method starts. ts WRM formulation, applied in the governing equation (2), has the following form:

$$(w, \Delta u) = -\frac{3\Delta t}{2}\left[(w, au_x^n) + (w_x, \nu u_x^n)\right] + \frac{\Delta t}{2}\left[(w, au_x^{n-1}) + (w_x, \nu u_x^{n-1})\right] + \Delta t(w, s^n) \tag{14}$$

Applying the Galerkin formulation ($w_i = N_i$ and $u = \sum_i u_j N_j$) in both equations we have:

$$(N_i, N_j)\Delta u = -\Delta t\left[a(N_i, \frac{dN_j}{dx}) + \nu(\frac{dN_i}{dx}, \frac{dN_j}{dx})\right] u^n + \Delta t(N_i, s^n) \tag{15}$$

$$\begin{aligned}(N_i, N_j)\Delta u = &-\frac{3\Delta t}{2}\left[a(N_i, \frac{dN_j}{dx}) + \nu(\frac{dN_i}{dx}, \frac{dN_j}{dx})\right] u^n + \\ &+\frac{\Delta t}{2}\left[a(N_i, \frac{dN_j}{dx}) + \nu(\frac{dN_i}{dx}, \frac{dN_j}{dx})\right] u^{n-1} + \Delta t(N_i, s^n)\end{aligned} \tag{16}$$

The code implementation of the matrices of the equation $A\Delta u = Bu + f$ using Lagrange multipliers becomes:

```
if d_temp == 3
Kt  = a*G + nu*K;
A = M;
Mf = M*f;
nccd = size(Accd1,1);
Atot = [A Accd1'; Accd1 zeros(nccd)];
% Factorization of matrix Atot
[L,U] = lu(Atot);
Sol = c;
% Loop to compute the transient solution
for i=1:nstep
    %FIRST STEP - EULLER
    if i == 1
    aux = dt*( -Kt*c + Mf);
    F = [aux;bccd1*0];
    dc  = U\(L\F);
    c = c + dc(1:npoin);
    Sol = [Sol c];
    else
    c1 = Sol(:,i-1);
    aux = dt*(-(3/2)*Kt*c + (1/2)*Kt*c1  + Mf);
    F = [aux;bccd1*0];
    dc  = U\(L\F);
    c = c + dc(1:npoin);
    Sol = [Sol c];
    end
end
```

3.2 Time Discontinuous Galerkin Formulation

The WR formulation of the homogeneous linear advection-diffusion equation can be written as:

$$\int\int_{Q^n} w^h(u_t^h + a\cdot\nabla u^h - \nu\nabla^2 u^h)d\Omega dt + \int_\Omega w^h(t_+^n)(u^h(t_+^n) - u^h(t_-^n))d\Omega = 0 \tag{17}$$

The integration by parts of the second order derivative term leads to:

$$\begin{aligned}\int\int_{Q^n} w^h(u_t^h + a\cdot\nabla u^h)d\Omega dt + \int\int_{Q^n} \nu\nabla w^h\cdot\nabla u^h d\Omega dt +\\ -\int_t\int_\Gamma \nu w^h\nabla u^h\cdot n_e d\Gamma dt + \int_\Omega w^h(t_+^n)(u^h(t_+^n) - u^h(t_-^n))d\Omega = 0\end{aligned} \tag{18}$$

Assuming an space-time interpolation such that: $u^h(x,t) = \sum_B^{n_{np}} N_B(x)[\Theta_1(t)\tilde{u}_B^n + \Theta_2(t)u_B^{n+1}]$ and $w^h = \sum_A^{n_{np}} N_A[\Theta_1(t) + \Theta_2(t)]$. With linear time interpolation such that $\Theta_1 = (t^{n+1} - t)/\Delta t$ and $\Theta_2 = (t - t^n)/\Delta t$. We can write equation (18) for each time slab as:

$$\begin{aligned}\sum_B^{n_{np}}(\int\int_{Q^n} N_A\Theta_1\left[N_B\frac{u_B^{n+1} - \tilde{u}_B^n}{\Delta t} + (\Theta_1\tilde{u}_B^n + \Theta_2 u_B^{n+1})(a\cdot\nabla N_B)\right]d\Omega dt +\\ +\int\int_{Q^n} \nu\nabla N_A\Theta_1(\Theta_1\tilde{u}_B^n + \Theta_2 u_B^{n+1})\nabla N_B d\Omega dt + \int_t\int_\Gamma \nu N_B\Theta_1(\Theta_1\tilde{u}_B^n + \Theta_2 u_B^{n+1})\nabla N_B\cdot n_e d\Gamma dt) = 0\end{aligned} \tag{19}$$

$$\sum_{B}^{n_{np}}(\int\int_{Q^n} N_A\Theta_2\left[N_B\frac{u_B^{n+1}-\tilde{u}_B^n}{\Delta t}+(\Theta_1\tilde{u}_B^n+\Theta_2 u_B^{n+1})(a\cdot\nabla N_B)\right]d\Omega dt+$$
$$+\int\int_{Q^n}\nu\nabla N_A\Theta_2(\Theta_1\tilde{u}_B^n+\Theta_2 u_B^{n+1})\nabla N_B d\Omega dt+\int_t\int_\Gamma \nu N_B\Theta_2(\Theta_1\tilde{u}_B^n+\Theta_2 u_B^{n+1})\nabla N_B\cdot n_e d\Gamma dt)+ \tag{20}$$
$$+\int_\Omega N_A\sum_{B}^{n_{np}} N_B(\tilde{u}_B^n-u_B^n)d\Omega=0$$

3.3 Results

The unsteady advection-diffusion of a Gaussian hill is solved with the following setup: $a = 1$, $\Delta x = 1/150$. The time step $\Delta t = [1/150 \quad 1/500]$, which leads to a Courant number $C = [1 \quad 0.3]$. The diffusion coefficient $\nu = (1/3)\cdot[10\times10^{-3} \quad 20\times10^{-4} \quad 10\times10^{-5}]$, which leads to Pe $= [1 \quad 5 \quad 100]$. The results for the Adam-Bashforth method and Padé R22 at the last time-step (t = 6 s) are compared below.

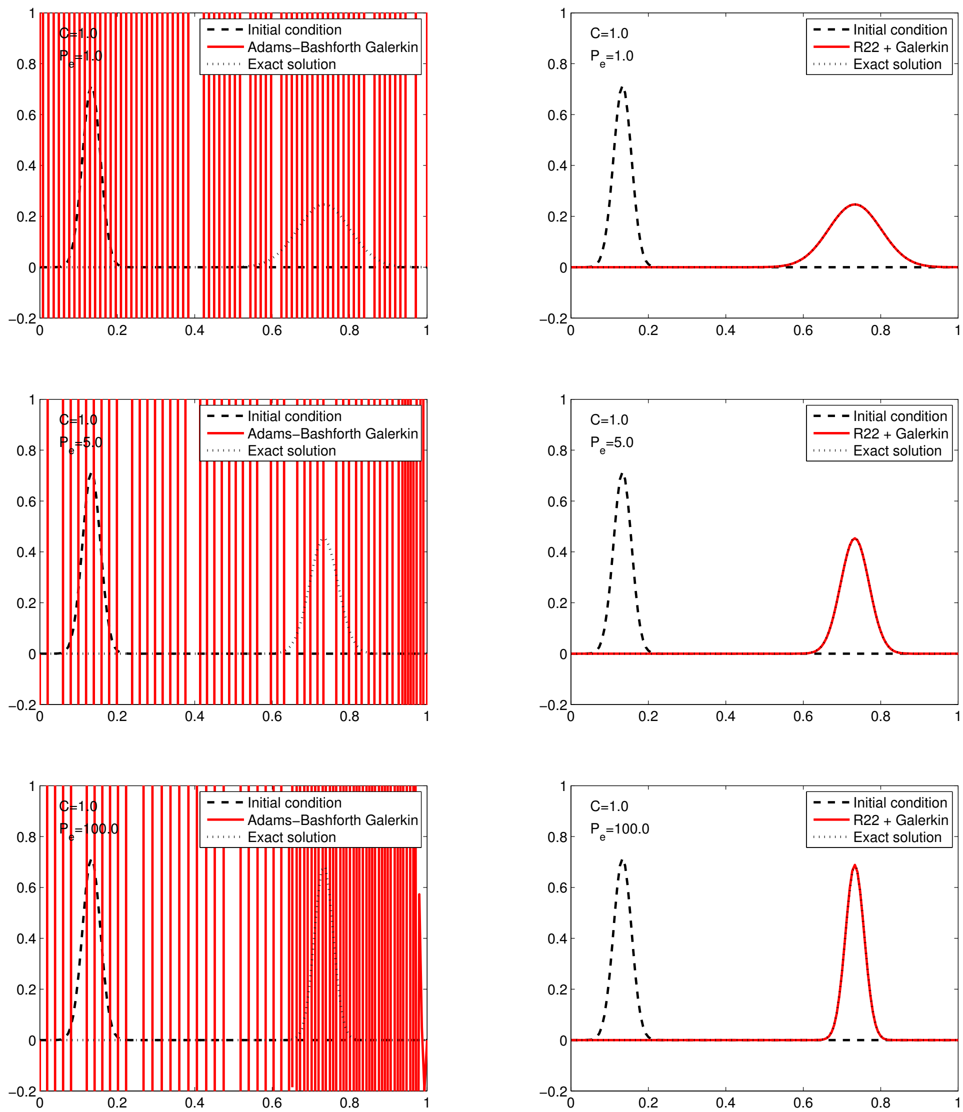


Figure 3: Gaussian Hill problem: t = 0.6 s and C = 1.

As it can be seen, Adam-Bashforth method, being a second order explicit time integration scheme, is unstable for all Pe numbers, while R22, which is a 4th order implicit scheme has an accurate and stable behavior for all Pe numbers.

The results for both methods with $C = 1$ and different Pe values is shown in Figure (4). As Courant number is reduced, the 2nd order explicit scheme starts to behave stable for Pe higher than 5.The 4th implicit scheme is even more accurate for lower Courant number.

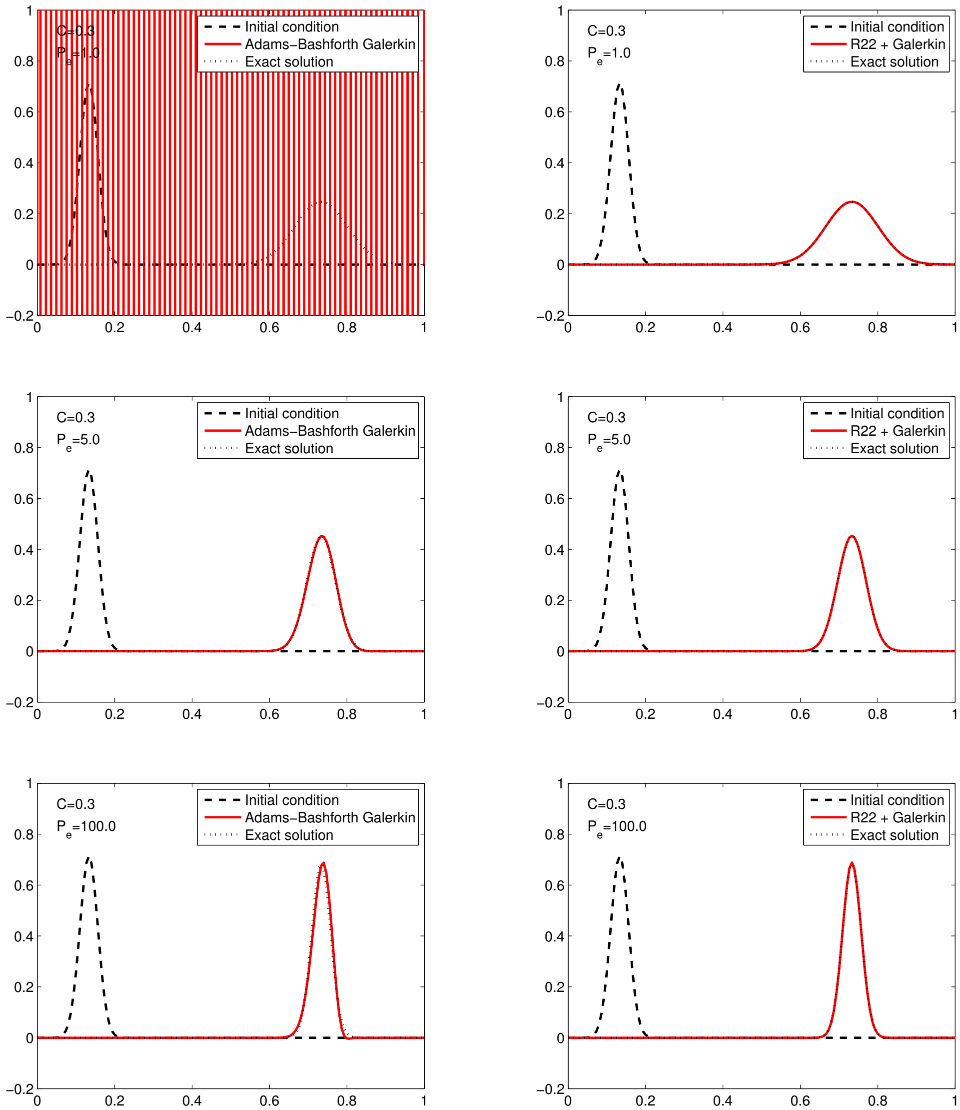


Figure 4: Gaussian Hill problem: t = 0.6 s and C = 0.3.