1D Unsteady Transport
KIMEY WAZARE

Ques. Pure Transport Equation.

$$u_t + (a.\nabla)u = s \qquad \text{in } \Omega \text{ x}]0, T[$$

$$u\,(x, 0) = u_0(x) \qquad \text{on } \Omega \text{ at } t = 0,$$

$$u = u_D \qquad \text{on } \Gamma_D^{in} \text{ x}]0, T[$$

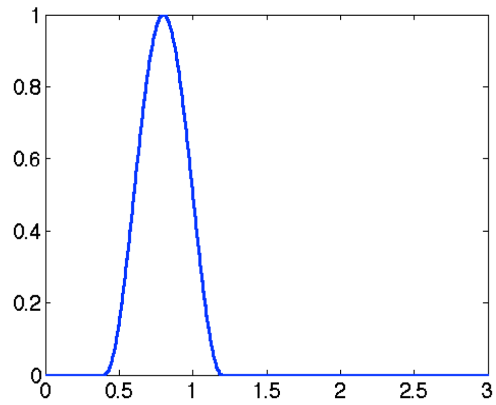
$$-a u.n = h \qquad \text{on } \Gamma_N^{in} \text{ x}]0, T[.$$

$$\Gamma^{in} = \{x \,\epsilon\, \Gamma \mid a.n < 0\}$$

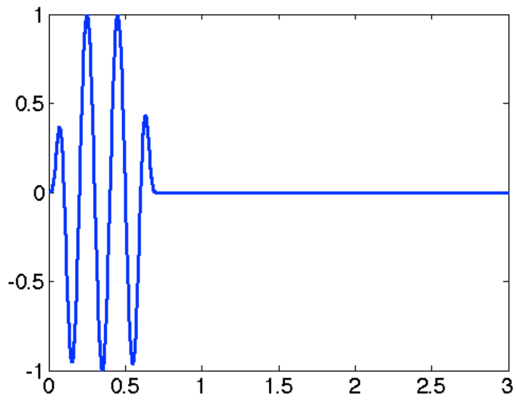
1. Zero source term,
2. Dirichlet boundary conditions on the inflow boundary.

Solution: Unsteady Convention-Diffusion Problem.

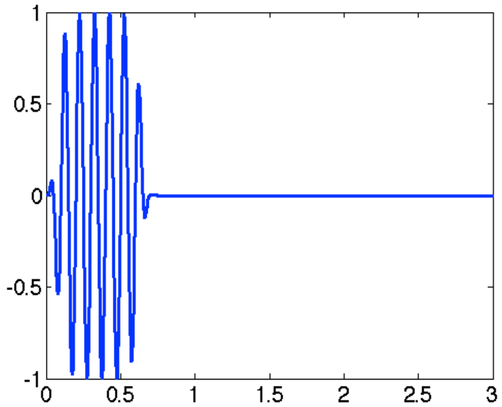
Define Problem: This graphs are the problem.



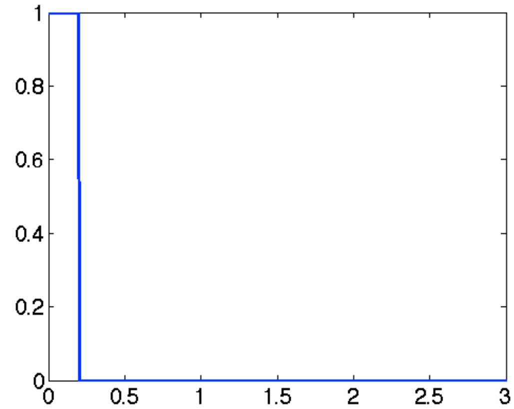
Problem 1



Problem 2



Problem 3



Problem 4

Introduction:

To solve the defined problem using Time Discretization method such as Lax-Wendroff, Leap-Frog, Taylor-Galerkin & Crank-Nicolson methods.

```
% Method used for solving the problem
disp(' ')
disp ('The problem can be solved using one of the following methods: ');
disp ('     [1] Lax-Wendroff + Galerkin')
disp ('     [2] Lax-Wendroff with lumped mass matrix + Galerkin')
disp ('     [3] Crank-Nicolson + Galerkin')
disp ('     [4] Crank-Nicolson with lumped mass matrix + Galerkin')
disp ('     [5] Third order Taylor-Galerkin + Galerkin')
disp ('     [6] Leap Frog + Galerkin')
disp ('     [7] Two Step Third Order with alpha = 1/9 Taylor-Galerkin + Galerkin')
```

Code:

The Matlab code for Lax-Wendroff and Crank-Nicolson were already given and some changes are made in the code for Leap-Frog, Third order & 2 step third order Taylor-Galerkin method. The changes are as follow,

1. Leap-Frog Method:

```
case 6 % Leap Frog + Galerkin
    A = M;
    B = -2*a*dt*C;
    methodName = 'LF';
```

2. Third order Taylor-Galerkin Method

```
case 5 % Third order Taylor-Galerkin + Galerkin
    A = M + dt^3/6*a^2*K;
    B = -dt*a*C-dt^2/2*a*K;
    methodName = 'TG3';
```

3. 2 Step Third Order Taylor-Galerkin Method

```
case 7 % 2 step TG3-I
    A = M;
    B = -(1/3)*dt*a*C-(1/9)*dt^2*a^2*K;
    methodName = 'TG3';
case 8 % 2 step TG3-II
    A = M;
    B = -a*dt*C;
    methodName = 'TG3';
```

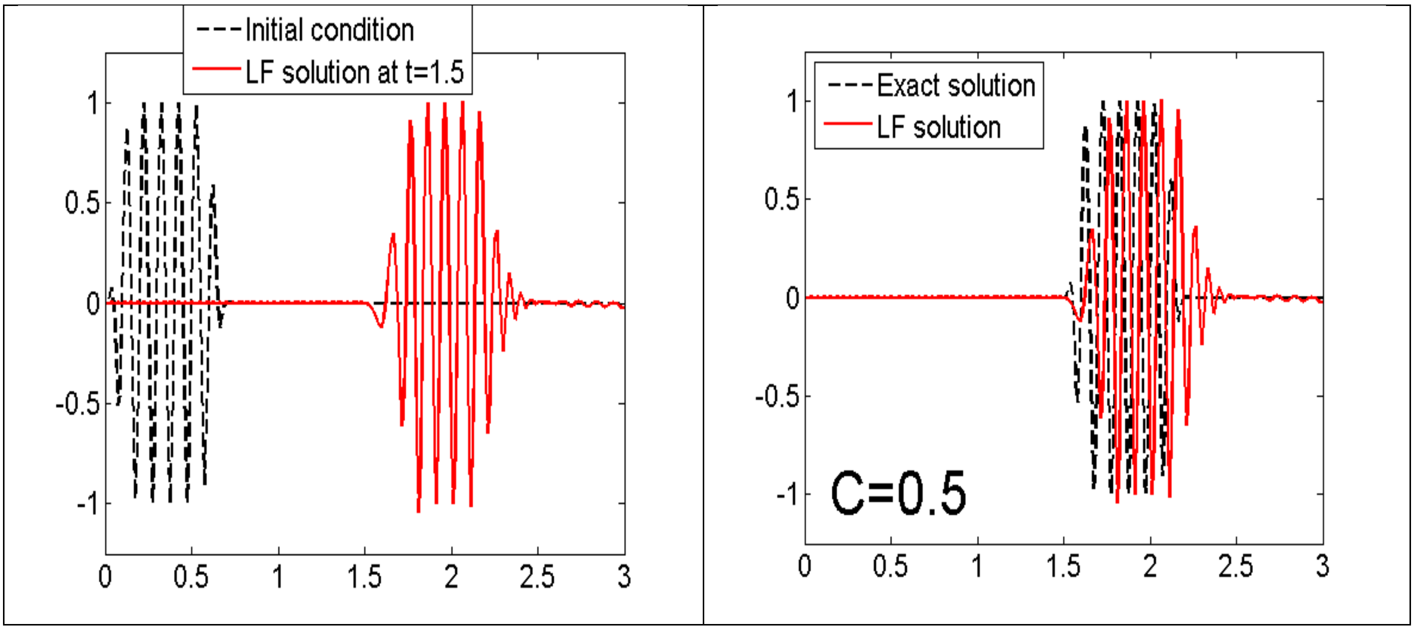
4. Changes in file ‘main.m’

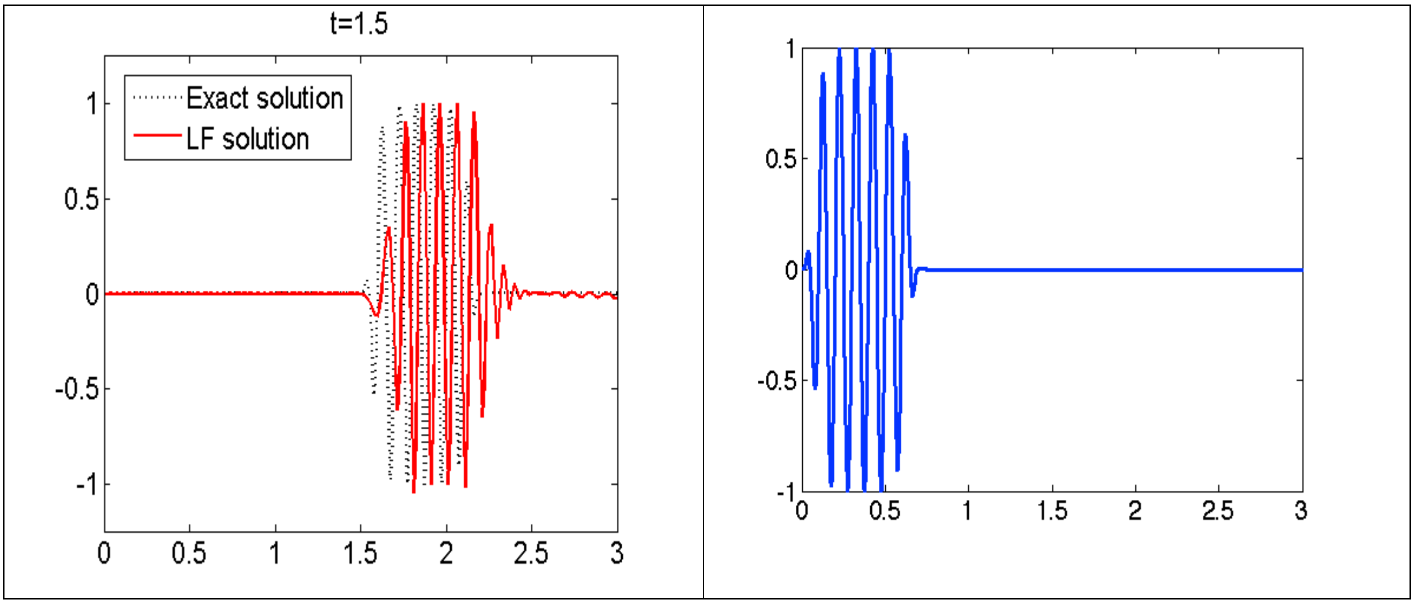
```
for n = 1:nStep
if (method ==1 || method ==2 || method ==3 || method ==4 || method ==5)
    Du = A\(B*u(ind_unk,n) + f);
    u(ind_unk,n+1) = u(ind_unk,n) + Du;
    elseif (method ==6)
    if(n==1)
        Du = A\(B*u(ind_unk,n) + f);
        u(ind_unk,n+1) = u(ind_unk,n) + Du;
        clear A,B;
        [A,B,methodName] = System(method,M,K,C,a,dt);
        A = A(ind_unk,ind_unk);
        B = B(ind_unk,ind_unk);
    else
        u(ind_unk,n+1) = A\(A*u(ind_unk,n-1)+B*u(ind_unk,n) + f);
    end
    elseif (method==7)

        [A,B,methodName]= System(7,M,K,C,a,dt);
        A = A(ind_unk,ind_unk);
        B = B(ind_unk,ind_unk);
        Du = A\(B*u(ind_unk,n) + f);
          u_bar= u(ind_unk,n) + Du;
        clear A,B;
        [A,B,methodName]= System(8,M,K,C,a,dt);
        A = A(ind_unk,ind_unk);
        B = B(ind_unk,ind_unk);
        K1 = K(ind_unk,ind_unk);
        Du = A\(B*u(ind_unk,n)- .5*a^2*dt^2*K1*u_bar + f);
        u(ind_unk,n+1) = u(ind_unk,n) + Du;
end
end
```

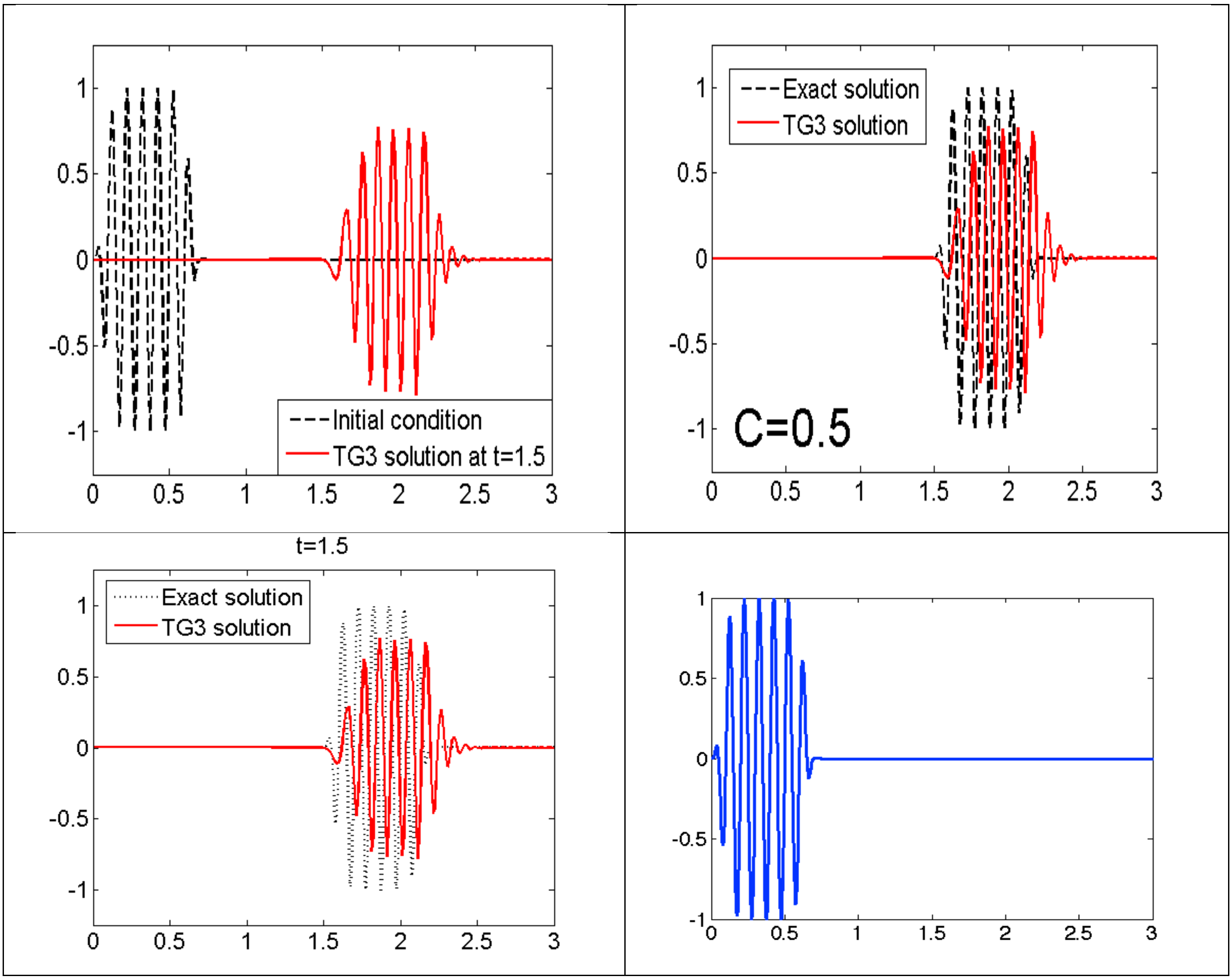
Graphs: Graphs are generated for the above mentioned problem 3 for all different formulation.

1. Leaf-Frog Method

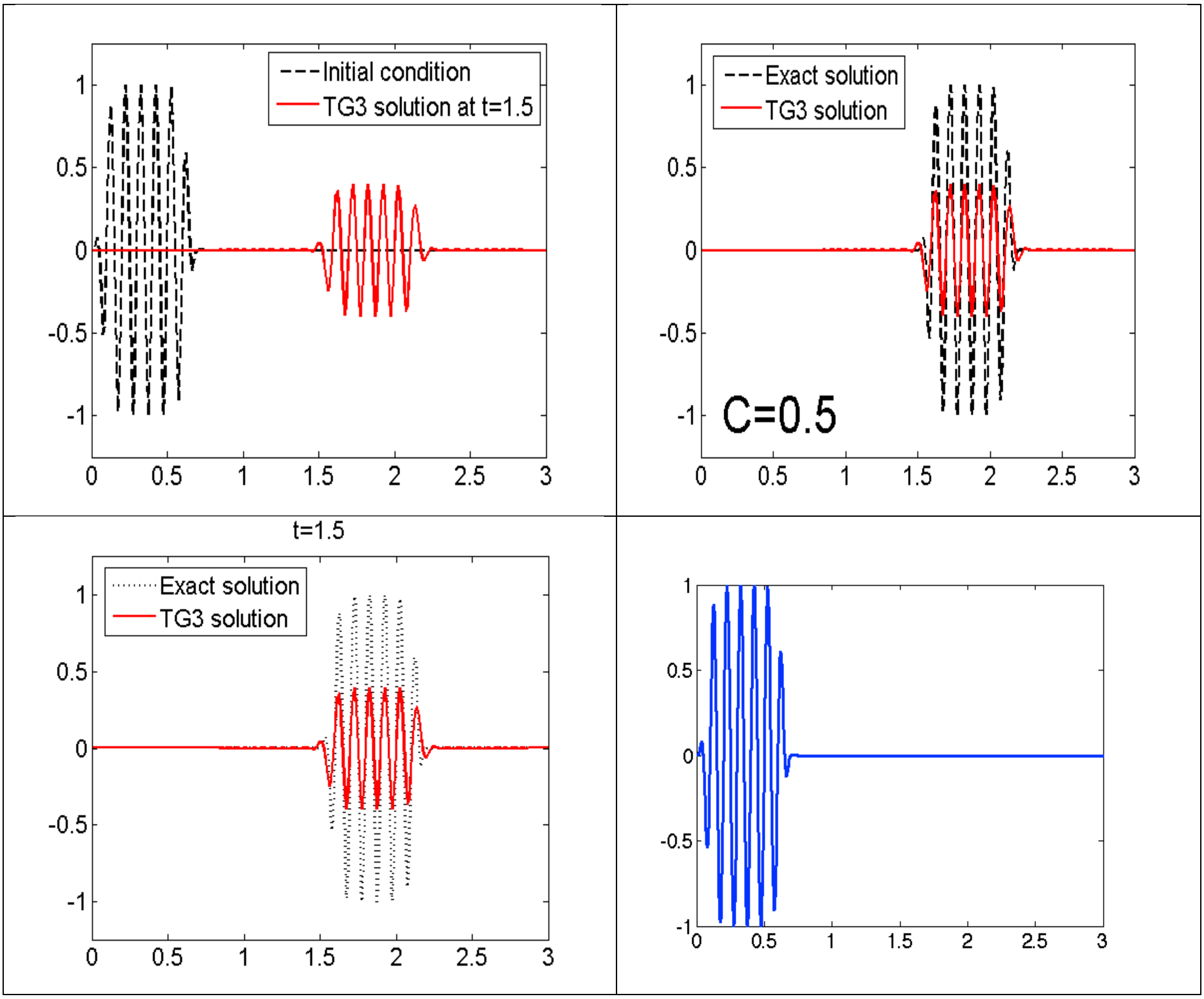




2. Third order Taylor-Galerkin Method



3. 2 Step Third Order Taylor-Galerkin Method



Conclusion: It can be concluded that Leaf frog deviates more compare too Third Order Galerkin Method and Two Step Third Order Galerkin method.