# FINITE ELEMENTS IN FLUIDS

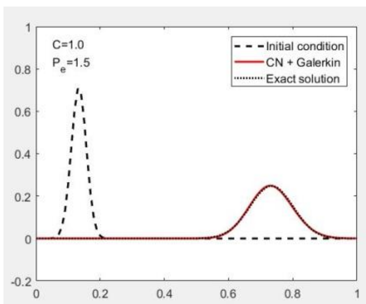
ASSIGNMENT7- GAUSSIAN HILL

-By Anurag Bhattacharjee

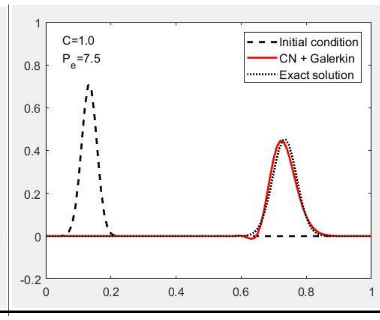
## RESULTS FOR DIFFERENT VALUES OF VISCOSITY

For this comparison default values of the code are taken while changing the values of diffusion coefficient. For spatial discretization Galerkin method has been chosen for it's simplicity. The Courant number is kept constant for this observation.

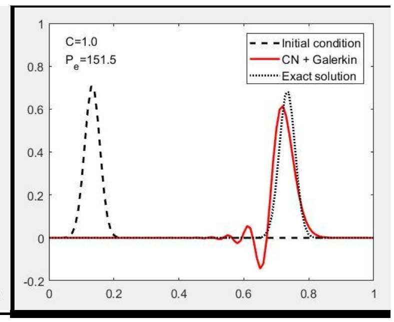
Crank Nicolson + Galerkin




DC=0.0033,P=1.5




DC=0.00067,P=7.5

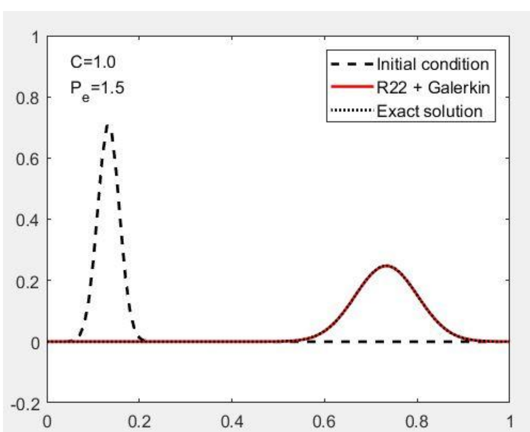



DC=0.000033, P=151.5

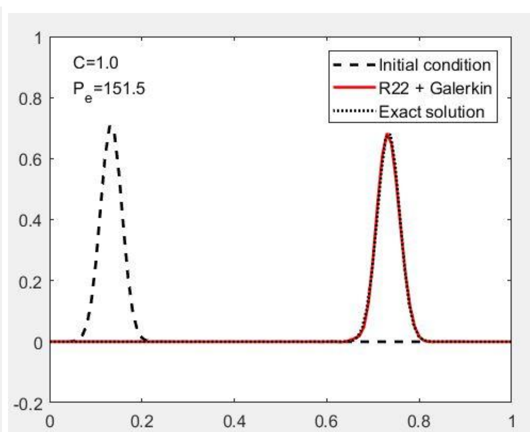
Here we find that with increasing Peclet number the amplitude of the solution at the end of simulation rises. However, we find that there is noticeable oscillations for Pe=151.5 which is to be expected as in this case of a convection dominated problem, a higher value of Pe will result in instabilities. Since we know viscosity is inversely proportional to diffusion, the plots agree with the theory. Increasing viscosity will lead to less diffusivity and will result in less drop in amplitude.

As the Peclet number decreases, diffusion becomes prominent and we see a drop in the amplitude which is in accordance with the theory. The same behavior can be observed for the other methods as well-

R22 + Galerkin

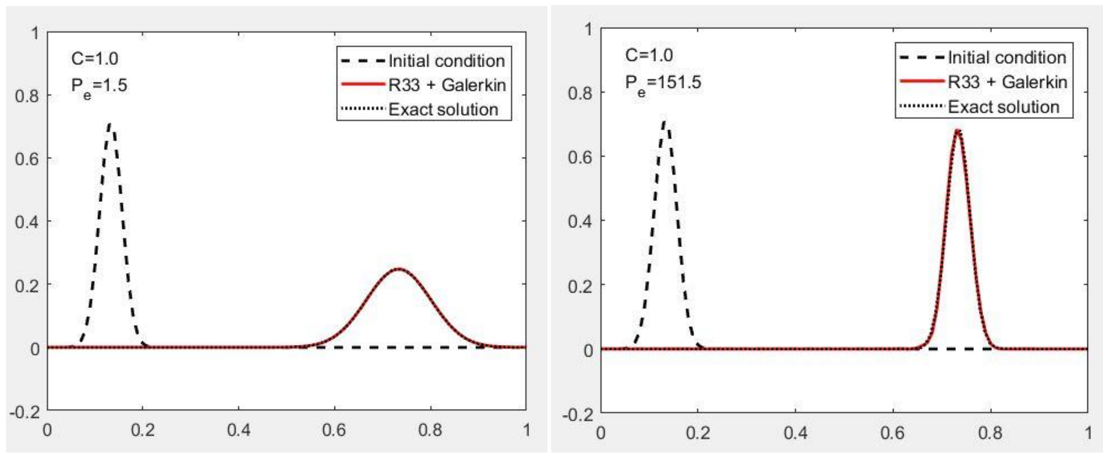



DC=0.0033,P=1.5




DC=0.000033,P=151.5

R33 + Galerkin




DC=0.0033,P=1.5 DC=0.000033,P=151.5

As was the case with Crank Nicolson we find higher amplitudes with lower diffusivity. It can also be seen from the plots that unlike Crank-Nicolson there are no oscillations for R22 and R33 methods for high Peclet number which can be attributed to their higher order of accuracy.

Similar profiles are obtained for other spatial discretizations as well.

## ADAMS- BASHFORTH METHOD

The code is implemented in the initial 1-D pure convection code given in earlier lectures. The entire code was modified to include diffusion term and the initial conditions prescribed in the problem. Galerkin space discretization technique was adopted in the following way-

$$u^{n+1} = u^n + \frac{\Delta t}{2}\left(3u_t^n - u_t^{n-1}\right)$$

$$\Rightarrow u^{n+1} = u^n + \frac{\Delta t}{2}\left(3\left(-au_x^n + \gamma u_{xx}^n\right) - \left(-au_x^{n-1} + \gamma u_{xx}^{n-1}\right)\right)$$

$$\Rightarrow u^{n+1} = u^n + \frac{3\Delta t}{2}\gamma u_{xx}^n - \frac{3a\Delta t}{2}u_x^n + \frac{\cancel{3}a\Delta t}{2}u_x^{n-1} - \frac{\gamma\Delta t}{2}u_{xx}^{n-1}$$

Using galerkin formulation,

$$(\omega, \Delta u^{n+1}) = \frac{3\Delta t\gamma}{2}(\omega, u_{xx}^n) - \frac{3a\Delta t}{2}(\omega, u_x^n) + \frac{a\Delta t}{2}(\omega, u_x^{n-1}) - \frac{\gamma\Delta t}{2}(\omega, u_{xx}^{n-1})$$

Integrating by parts & neglecting boundary terms

$$\Rightarrow (\omega, \Delta u^{n+1}) = -\frac{3\Delta t}{2}\gamma(\omega_x, u_x^n) - \frac{3a\Delta t}{2}(\omega, u_x^n) + \frac{a\Delta t}{2}(\omega, u_x^{n-1}) + \frac{\gamma\Delta t}{2}(\omega_x, u_x^{n-1})$$

Since the method contains u(n-1) the method is not self initializing. So, we implement forward euler method to initialize Adams Bashforth for n=1

For $n=1$,

<u>Forward Euler</u>

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

$$\Rightarrow \Delta u^{n+1} = \Delta t\, u_t^n$$

$$\Rightarrow \Delta u^{n+1} = \Delta t\left(-a u_x^n + \nu u_{xx}^n\right)$$

Galerkin,

$$\Rightarrow (\omega, \Delta u^{n+1}) = -a\Delta t(\omega, u_x^n) + \nu\Delta t(\omega, u_{xx}^n)$$

Integrating by parts & neglecting boundary terms

$$(\omega, \Delta u^{n+1}) = -a\Delta t(\omega, u_x^n) \cancel{+} - \nu\Delta t(\omega_x, u_x^n)$$

The following code was implemented in *system.m*

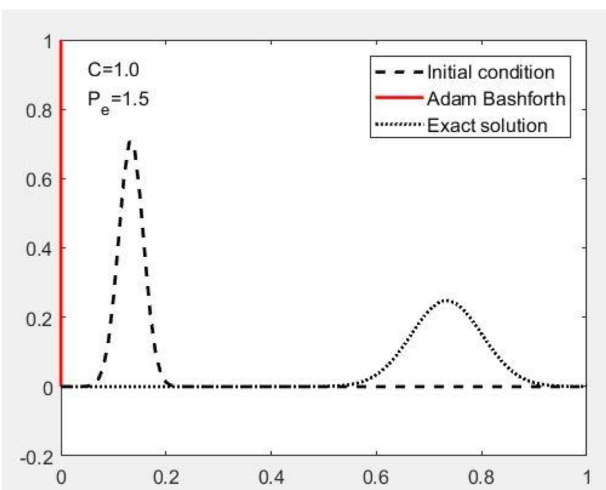
```
switch method
    case 1 % Adam Bashforth + Galerkin
        A= M;
        B= -(3/2)*dt*a*C - (3/2)*nu*dt*K;
        Bb= (1/2)*dt*a*C + (1/2)*nu*dt*K;
        methodName = 'Adam Bashforth';
    case 2 % Forward Euler + Galerkin
        A= M;
        B= -a*dt*C - nu*dt*K;
        Bb=0;
        methodName = 'Forward Euler';
```

And the following changes were made to *main.m*

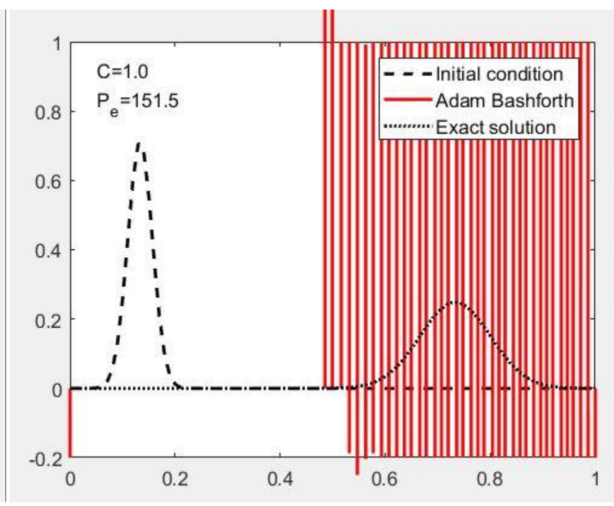
```
    %Forward Euler
[A,B,Bb,methodName] = System(2,M,K,C,a,nu,dt);
  ind_unk = 2:nPt-1;
A = A(ind_unk,ind_unk);
B = B(ind_unk,ind_unk);
Du = A\(B*u(ind_unk,1));
u(ind_unk,2) = u(ind_unk,1) + Du;
clear A,B ;
%Adams-Bashforth
[A,B,Bb,methodName] = System(1,M,K,C,a,nu,dt);
  ind_unk = 2:nPt-1;
A = A(ind_unk,ind_unk);
B = B(ind_unk,ind_unk);
Bb = Bb(ind_unk,ind_unk);

for n = 2:nStep
    Du = A\((B*u(ind_unk,n)) + (Bb*u(ind_unk,n-1)));
    u(ind_unk,n+1) = u(ind_unk,n) + Du;
end
```

<u>Results</u>



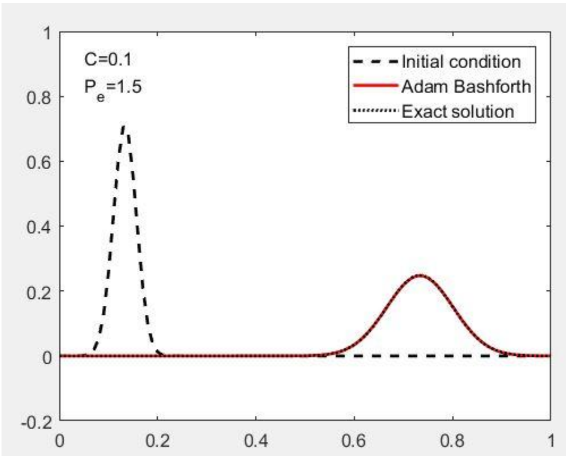
<u>DC=0.0033,P=1.5</u>



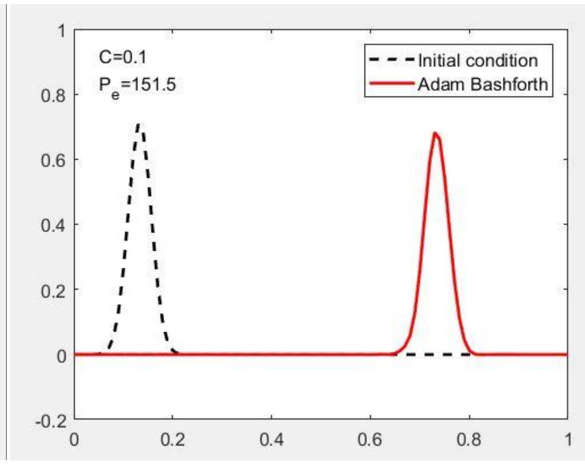
<u>DC=0.000033,P=151.5</u>

We find that for the same values of C (=1) and Pe (=1.5 and 151.5) the method shows extreme inconsistency with the exact solution. Compared to the R22, which is extremely stable this method fails heavily. However, it should be noted that R22 is a 2-step fourth order method and so has a much higher order accuracy than Adams Bashforth.

Since Adams Bashforth is an explicit method it is expected to work for lower values of Courant number which is validated by the following code result for C=0.1 with Pe= 1.5 and 151.5 for which the code previously failed.




DC=0.0033,P=1.5




DC=0.000033,P=151.5

## TIME DISCONTINUOUS GALERKIN FORMULATION

Problem Definition

$u_t + a u_x - \nu u_{xx} = 0$ in $\Omega \times ]0,T[$

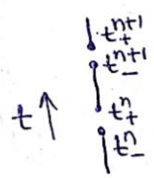
$u(x,0) = u_0(x)$ on $\Omega$ at $t=0$

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

Since we are considering discontinuity at space-time slab interfaces, we consider

$$u^h(t^n_\pm) = \lim_{\varepsilon \to 0^+} u^h(t^n \pm \varepsilon) \quad \text{—(1)}$$

Visually this would mean when moving from one time step to the next, the jump in time discontinuity is considered negligible.



The trial and weighting functions are defined as follows for the dirichlet inlet boundary ~~and so~~

$$u^h|_{Q_e^n} \in P_K(Q_e^n), \; u^h|_{\Gamma_{in}} = u_D$$

$$w^h|_{Q_e^n} \in P_K(Q_e^n), \; w^h|_{\Gamma_{in}} = 0$$

where, $Q_e^n$ defines continuity in space for all elements, &

$P_K$ indicates the space of polynomials.

The weighted residual formulation for a homogenous convection-diffusion equation with dirichlet inlet is as follows:-

$$\iint_{Q^n} \left(u_t^h + a.\nabla u^h - \nabla.(\nu \nabla u^h)\right) d\Omega\, dt + \int_\Omega w^h(t_+^n)\left(u^h(t_+^n) - u^h(t_-^n)\right) d\Omega = 0 \quad \text{--- (2)}$$

with initial condition $u^h(x, t_-^0) = u_0(x)$.

For finite element approximation, we will use the following piecewise functions, polynomial in space & linear in time.

$$u^h(x,t) = \sum_{B=1}^{n_p} N_B(x)\left(\theta_1(t)\, u_B^n + \theta_2(t)\, u_B^{n+1}\right)$$

$N_B(x)$ is the spatial shape function at node B.
$\theta_1(t)$ & $\theta_2(t)$ are linear time interpolation functions -

$$\theta_1(t) = \frac{t^{n+1} - t}{\Delta t}, \quad \Delta t = (t^{n+1} - t^n)$$

$$\theta_2(t) = \frac{t - t^n}{\Delta t}$$

$\Delta t$

For Galerkin formulation, the weighted functions are taken similarly as follows:

$$w^h = N_B \theta_1 \ \& \ N_B \theta_2 \text{ at node } B,\ B = 1, \dots, n_p,$$

$n_p$ = total number of nodes

From Equation (1), we have

$$\sum_{B=1}^{n_p} \left\{ \iint_{Q^n} N_A \theta_2 \left[ \frac{\partial}{\partial t}\left(N_B\left(\theta_1 \tilde{U}_B^n + \theta_2 U_B^{n+1}\right)\right) + a.\nabla\left(N_B\theta_1\tilde{U}_B^n + N_B\theta_2 U_B^{n+1}\right) - \nabla.\left(\nu \nabla\left(N_B\theta_1\tilde{U}_B^n + N_B\theta_2 U_B^{n+1}\right)\right)\right] d\Omega\, dt \right.$$
$$+ \int_\Omega N_A \sum_{B=1}^{n_p} N_B\left(\tilde{U}_B^n - U_B^n\right) d\Omega = 0 \quad \text{--- (3)}$$

[From (1)], where, $\tilde{U}_B^n$ & $U_B^n$ are the nodal values of $u^h$ at $t_+^n$ & $t_-^n$. & $N_A$ is the nodal value of $w$ at $t_+^n$. As $w(t_+^n) = N_A \theta_1 = N_A \dfrac{t^{n+1} - t_+^n}{\Delta t}$

$$\approx N_A \frac{\cancel{\Delta t}}{\cancel{\Delta t}}$$
$$= N_A$$

From (3),

$$\sum_{B=1}^{n_p} \left\{ \iint_{Q^n} N_A \theta_2 \left[ N_B \frac{U_B^{n+1} - \tilde{U}_B^n}{\Delta t} + \left(\theta_1 \tilde{U}_B^n + \theta_2 U_B^{n+1}\right)(a.\nabla) N_B - \left(\theta_1 \tilde{U}_B^n + \theta_2 U_B^{n+1}\right)(\nabla.\nu\nabla) N_B \right] d\Omega\, dt \right.$$
$$+ \int_\Omega N_A \sum_{B=1}^{n_p} N_B \left(\tilde{U}_B^n - U_B^n\right) d\Omega = 0$$

This is the required formulation.