POLYTECHNIC UNIVERSITY OF CATALONIA
MASTER OF SCIENCE IN COMPUTATIONAL MECHANICS

Finite Element in Fluids

Student : Chiluba Isaiah Nsofu

Lecture : Dr. Matteo Giacomini

Date: 2nd April, 2018

Contents

1 Propagation of a cosine profile **2**
1.1 Comparison of Explicit methods 2
1.2 Comparison of Implicit methods 3

2 Propagation of a steep front **4**

3 The convection-diffusion case : Gaussian hill problem **6**

1 Propagation of a cosine profile

1.1 Comparison of Explicit methods

TG2 (Consistent mass matrix) formulation results using Courant 0.5 and 0.75

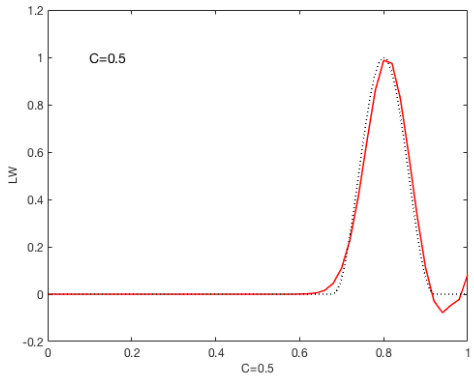

Figure 1.1: TG2 at C=0.5

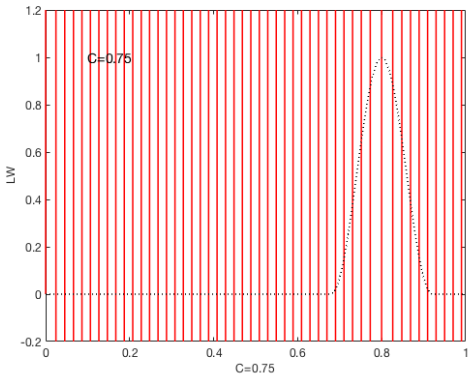

Figure 1.2: TG2 at C=0.75

LW-FD (Diagonal mass matrix) formulation results using Courant 0.5 and 0.75

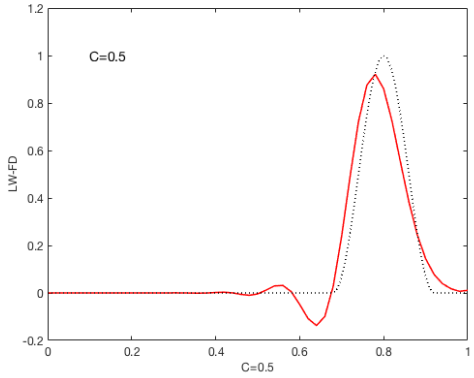

Figure 1.3: LW-FD for CFL=0.5

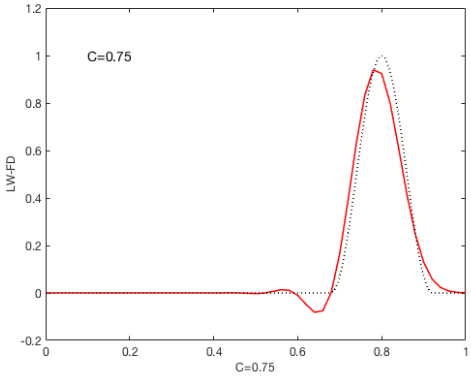

Figure 1.4: LW-FD, CFL=0.75

TG3 formulation results using Courant 0.5 and 0.75

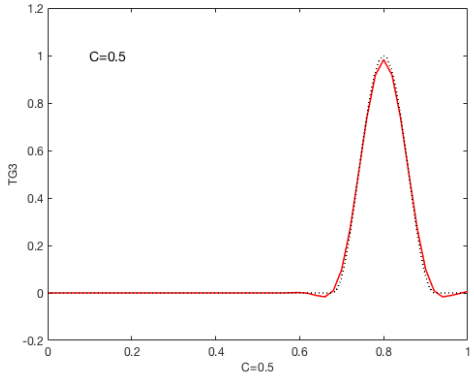

Figure 1.5: TG3 for CFL=0.5

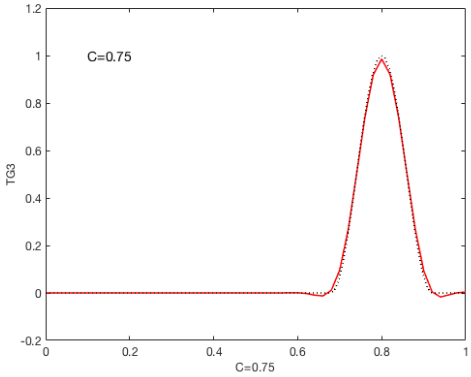

Figure 1.6: TG3, CFL=0.75

In comparison with the exact solution it can be concluded for that, the two explicit methods TG2 and TG3 with the consistent matrix are more accurate than the Lax-wendroff (LW-FD) scheme with a diagonal matrix. However from (1.1) its noted that TG2 cannot be used for C=0.75. In terms of stability we see that TG3 is more stable than TG2 and LW-FD Methods

1.2 Comparison of Implicit methods

CN formulation results using Courant 0.5 and 0.75

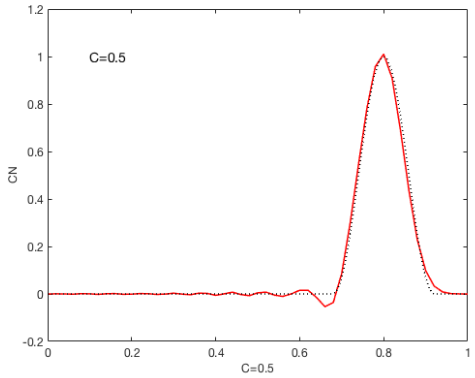

Figure 1.7: CN at C=0.5

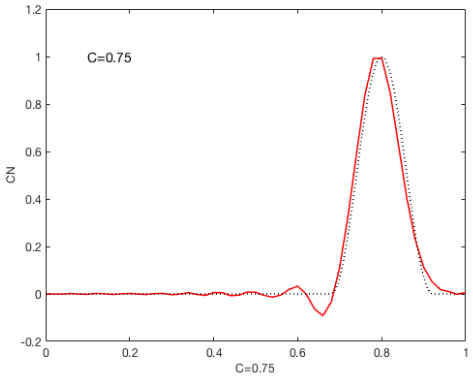

Figure 1.8: CN at C=0.75

CN-FD (Diagonal mass matrix) formulation results using Courant 0.5 and 0.75

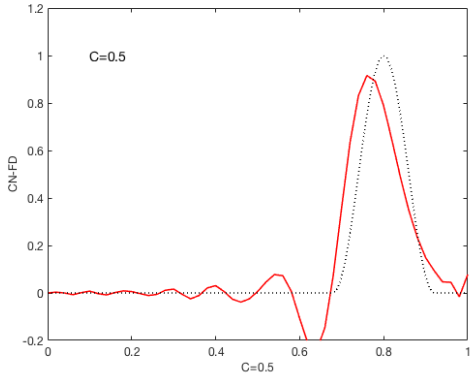

Figure 1.9: CN-FD for CFL=0.5

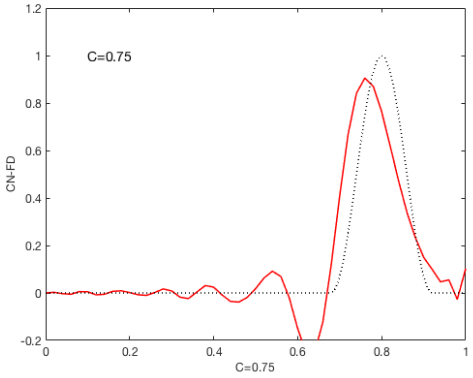

Figure 1.10: CN-FD, CFL=0.75

TG4 formulation results using Courant 0.5 and 0.75

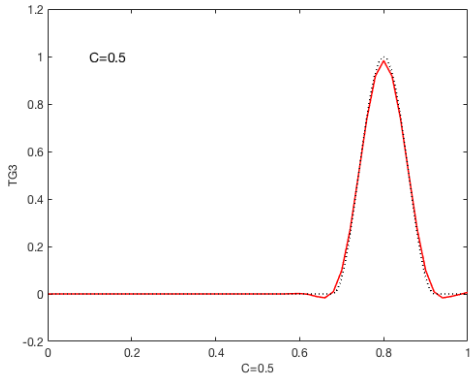

Figure 1.11: TG4 for CFL=0.5

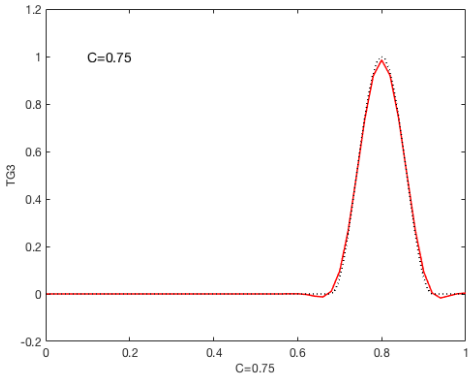

Figure 1.12: TG4, CFL=0.75

Comparing the results in Figure (1.7) (1.8), (1.11) and (1.11) with those in Figure (1.9),(1.10) its evident that implicit methods with the consistent mass matrix are more accurate than the one with the diagonal mass matrix. This behaviour is also similar with the explicit methods. Comparing CN and TG4 we can see that TG4 is more accurate. With regards to stability CN-FD is snot stable as shown in Figure (1.9),(1.10). Therefore of all the three implicit method TG4 is the most accurate and stable method.

2 Propagation of a steep front

1. Calculation of the CFL number

$$\begin{aligned} CFL &= a * \frac{\Delta t}{h} \\ &= 1 * \frac{0.0015}{0.02} \\ &= 0.75 \end{aligned}$$

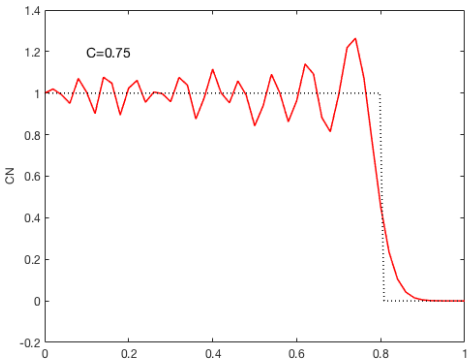

Figure 2.1: CN-G for CFL=0.75

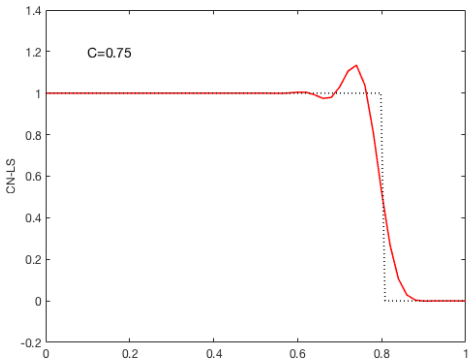

Figure 2.2: CN-LS, CFL=0.75

The propagation of the steep front problem is first solved using the Crank-Nicolson (with the Galerkin in space) and the results are shown in Figure(2.1). From this figure we can see that this formulation procuses a lot oscillations which are caused by the Galerkin method. To overcome this problem we employ another method that uses the Crank NiclosOn in time and the least square method in space. The results for this new method are shown in Figure (2.2). These results shows that this new method is bale to get ricd of the spurious oscillations.

Implementation of the Crank-Nicholson scheme in time and the least-squares formulation in space

The implementation of this method is based on the discrete form of this equation with respect to the convection equation which is given on page 121 of the Lecture book. This discrete form can also be derived based on the θ family of methods but in this work we will just use the final form as follows.

$$\left(1 + \left(\frac{1}{6}\delta^2 - \theta^2 C^2 \delta t^2\right)\right) \Delta u = -\frac{1}{2} C \delta u_j^n + \theta C^2 \delta^2 u_j^n \tag{1}$$

This discrete equation can be written using the spatial operator given the Lecture book on page 100 which yields the following equation.

$$\left((N_i, N_j) + \theta^2 \Delta t (\boldsymbol{a} \cdot \boldsymbol{\nabla} N_i, \boldsymbol{a} \cdot \boldsymbol{\nabla} N_j)\right) \Delta u = \Delta t \left(\boldsymbol{a} \cdot \boldsymbol{\nabla} N_i, N_j\right) u^n - \theta \Delta t^2 (\boldsymbol{a} \cdot \boldsymbol{\nabla} N_i, \boldsymbol{a} \cdot \boldsymbol{\nabla} N_j) u^n \tag{2}$$

Using the above equation we can say that for $\theta = \frac{1}{2}$ we are employing the Crack Nicolson

method. Hence this was the method that is implemented in the code as shown below.

$$A(isp,isp) = A(isp,isp) + w_ig * (N' * N + 0.25 * dt_2 * (a * Nx)' * (a * Nx))$$
$$B(isp,isp) = B(isp,isp) + w_ig * (dt * (a * Nx)' * N - (dt^2/2) * (a * Nx)' * (a * Nx))$$
$$f(isp) = f(isp) + w_ig * (dt * (N') * SourceTerm(x) + (dt^2/2) * (a * Nx)' * SourceTerm(x));$$

Solution using second-order Lax-Wendroff method

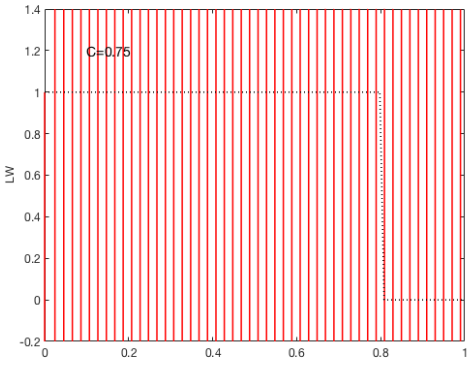

Figure 2.3: TG2 for CFL=0.75

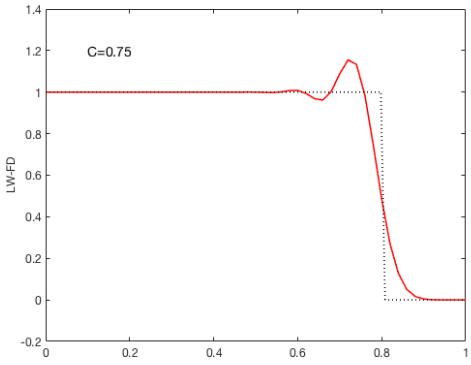

Figure 2.4: LW-FD, CFL=0.75

As it is evidently seen in Figure 2.3 using the second order Lax-Wendroff method with a consistent mass matrix produces unstable solution at CFL = 0.75 . In order to overcome this problem we need to use the lumped mass matrix hence the result in (2.4) shows that when the lumped mass is used the solution ia stable and accurate.

Implementation of the second-order two-step Lax-Wendroff method

This method is given on page 141 of the lecture book as

$$u^{n+1/2} = u^n + \frac{\Delta t}{2} u_t^n$$
$$U^{n+1} = u^n + \Delta t u_t^{n+1/2}$$

When we carry out the necessary representation of introducing the test function and performing integration by parts then we can implement the method in the code as follows.

$$A(isp,isp) = A(isp,isp) + w_ig * N' * N$$
$$B(isp,isp) = B(isp,isp) + w_ig * dt * (a * Nx)' * N$$
$$f(isp) = f(isp) + w_ig * dt_2 * (N') * SourceTerm(x)$$

At the same time we can implement the same method with the diagonal mass matrix as follows.

$$A(isp,isp) = A(isp,isp) + w_ig * diag((N' * N) * unos)$$
$$B(isp,isp) = B(isp,isp) + w_ig * dt * (a * Nx)' * N$$
$$f(isp) = f(isp) + w_ig * dt_2 * (N') * SourceTerm(x)$$

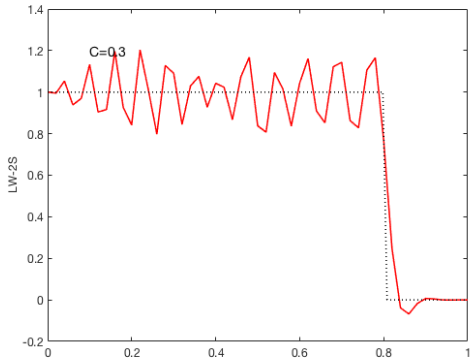

Figure 2.5: LW-2S for CFL=0.3

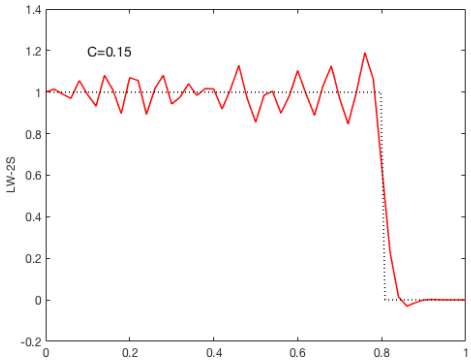

Figure 2.6: LW-2S for CFL=0.15

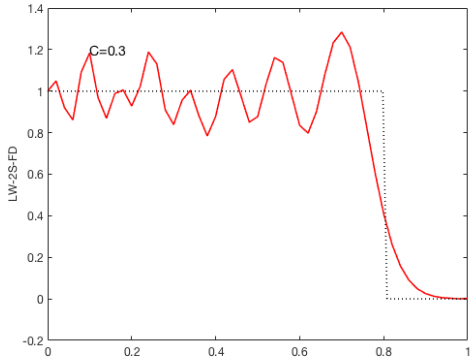

Figure 2.7: LW-FD-2S -Diag for CFL=0.3

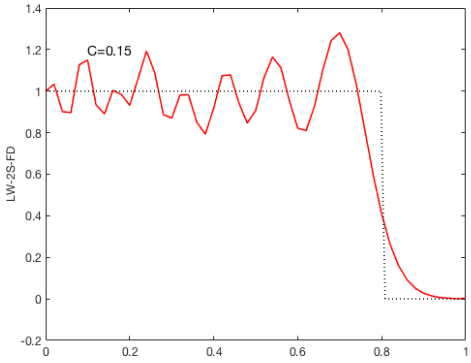

Figure 2.8: LW-FD-2S-Diag for CFL=0.15

As we can see in figure (2.5) , (2.6) , (2.7) and (2.8) the problem was solved using lower values of C= 0.3 and C =0.15. From the results we can see that both methods are not accurate and stable enough for this problem. Although it is stated from the literature in the lecture book that the method should be stable for C= 0.3 the results presented here states otherwise. From the author's point of view its either the formulation is not as stable as it is predicted in the lecture notes or there is something wrong with authors implementation of the method.

3 The convection-diffusion case : Gaussian hill problem

Crack-Nicolson in time + Galerkin in space solution

- Viscosity = 0.0033

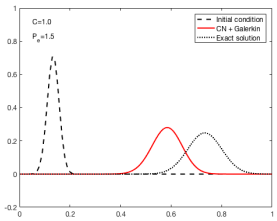

Figure 3.1: For pe=1.5

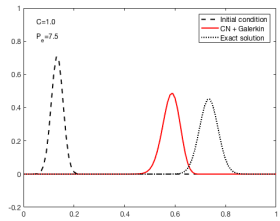

Figure 3.2: For pe= 7.5

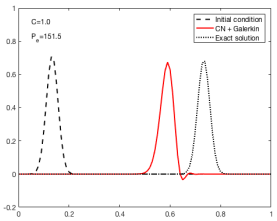

Figure 3.3: For pe=151

Figures (3.1), (3.2) and (3.3) shows the results for different values of the viscosity term. From these results we can see that as we decrease the value of viscosity the problem becomes highly convective which is a problem when using the Galerkin method in space. Hence from this figures we see that with an increase in the *pe* number the solution displays a lot of phase

errors. This leads to conclusion that the standard Galerkin method is not suitable to be used together the Crank - Nicolson method.

Comparison of Adam-Bashforth method and R22 in time methods

Adam-Bashforth in time + Galerkin in space results

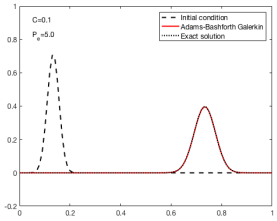

Figure 3.4: C=0.1 pe=5

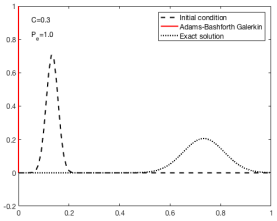

Figure 3.5: C=0.3 pe=1

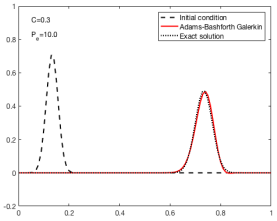

Figure 3.6: C=0.3 pe=10

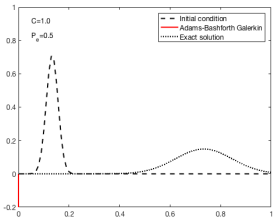

Figure 3.7: C=0.1 pe=0.5

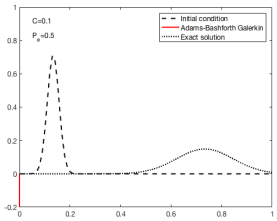

Figure 3.8: C=0.1 pe=0.5

R 22 in time + Galerkin in space

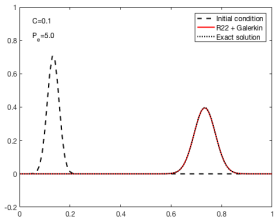

Figure 3.9: C=0.1 pe=5

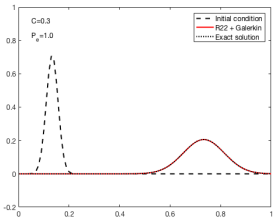

Figure 3.10: C=1 pe=0.5

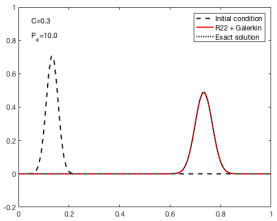

Figure 3.11: C=0.1 pe=0.5

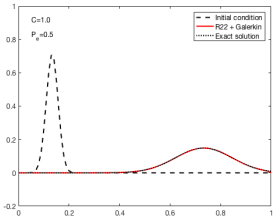

Figure 3.12: C=1 pe=0.5

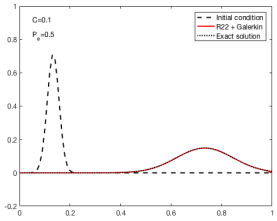

Figure 3.13: C=0.1 pe=0.5

The results shown a comparison of both methods for different values of Courant and pe numbers. From the results in Figures (3.4) (3.5) we see that if the value of the peclet number is very high then the Adam Bashforth method becomes accurate and stable . It should also be noted that this happens when we take a very small time step. The moment we decrease the value of Pe the Adam Bashforth method becomes unstable and inaccurate as it shown in Figures (3.5) (3.7) and (3.8). From all the results of this method the value of C seems to to have the same impact as the Pe value. On the other When using R22 in time we see that this method

combined with the Galerkin in space is very stable and accurate for that cases that have been considered.

Important points on the implementation od the Adams Bashforth method

This is a second order accuareate explicit method an dthe derivation of this method was done in class using the forward Taylor series

$$u(t^{n+1}) = u(t^n) + \Delta t u_t(t^n) + \frac{\Delta t^2}{2} u_{tt}(t^n) + ...$$

in which the second order term is approximated as

$$u_{tt}(t^n) = \frac{u_t(t^n) - u_t(t^{n-1})}{\Delta t} + ...$$

which gives the two level second order explict method

$$u^{n+1} = u^n + + \frac{\Delta t}{2} 3u_t^n - u_t^{n-1} + ...$$

We can replace u_t in the above method with our equation and use the forward Euler method for the first time step in the implementation of this method. interms of stability Stability of the second-order Adams-Bashforth method is also governed the same type as the Euler method, but its stability range is only half that for the Euler method. This is because in convection-diffusion, the stability condition depends on the value of Pe.

Time-discontinuous Galerkin formulation for the convection-diffusion equation

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

When we use linear finite element approximations in both space and time, it gives a third-order accurate and unconditionally stable method. Following the developments in Section 3.10.1 for the pure convection problem we can do the same for the to the convection diffusion case. Through this process the following partitioned matrix system is obtained for the nodal unknowns $\mathbf{u}^{n+1} - u^{n^+}$

$$\left(\mathbf{M} + \frac{2}{3}\Delta t \mathbf{C} + \frac{2}{3}\nu \Delta t \mathbf{K} \right) \mathbf{u}^{n+1} - \left(\mathbf{M} - \frac{1}{3}\Delta t \mathbf{C} - \frac{1}{3}\nu \Delta t \mathbf{K} \right) u^{n^+} = 0 \quad (3)$$

$$\left(\mathbf{M} + \frac{1}{3}\Delta t \mathbf{C} + \frac{1}{3}\nu \Delta t \mathbf{K} \right) \mathbf{u}^{n+1} - + \left(\mathbf{M} + \frac{2}{3}\Delta t \mathbf{C} + \frac{2}{3}\nu \Delta t \mathbf{K} \right) u^{n^+} = 2\ \mathbf{M} u^{n^-} \quad (4)$$

Where the matrices $\mathbf{M}$ (consistent mass), $\mathbf{C}$(convection), and $\mathbf{K}$ (diffusion) are defined as the general finite element matrices. Note that here the conditions $u^{n+1}(1) = u^{n^+}(1) = 0$ are enforced to satisfy the inlet condition. Also it should be noted that this is a third order accurate and unconditionally stable method which requires the solution of an algebraic system double the size of usual time-stepping algorithms.