DISCONTINOUS GALERKIN METHODS FOR COMPUTATIONAL MECHANICS:

PROJECT WORK: ACOUSTIC WAVE EQUATION:

Submitted to

Herr. Dr. Martin Kronbichler

Lehrstuhl für Numerische Mechanik

Technical University of Munich

Submitted by

Ramesh Kandan Murugan

Matrikel Nr: 03681648

M.Sc Computational Science and Engineering.

Faculty of Informatics

Technical University of Munich

TABLE OF CONTENTS:

OBJECTIVE:

To solve the Acoustic wave equation using the method of discontinuous Galerkin and to present the observations along with the computational results and to check their accordance with the true solution.

OBSERVATIONS & INFERENCES:

Task 1:

In order to find out the approximate formula for the maximum time step in terms of the prescribed CFL number, the following analytical treatment of the given equations would have to be considered.

$$\rho\, \partial v/\, \partial t + \nabla p = 0;$$

$$c^2(\partial p\, /\, \partial t) \;+\; \rho \nabla \cdot v \;=\; 0.$$

The above equations can be combined to form a one-dimensional system

$$\frac{\partial}{\partial t}\begin{bmatrix} v \\ p \end{bmatrix} + \begin{bmatrix} 0 & \frac{1}{\rho} \\ \frac{\rho}{c^2} & 0 \end{bmatrix} \frac{\partial}{\partial x}\begin{bmatrix} v \\ p \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad (1)$$

Which is analogous to the linear advection equation, $\frac{\partial u}{\partial t} + A\frac{\partial u}{\partial x} = 0$. Hence the **eigenvalues of the matrix** A gives the corresponding **wave speeds**. On solving, we find the eigenvalues ($\boldsymbol{\lambda}$) to be $\pm \boldsymbol{c}$**.** This implies that the system of equation is the **superposition of two linear advection equation with c and –c wave velocities.**

The maximum time step formula for this system would be given by:

$$\Delta \mathbf{t} \leq \frac{\mathbf{Cr}}{\mathbf{max}(\boldsymbol{\lambda})}\, \Delta \boldsymbol{x}_{\boldsymbol{min}}^{\boldsymbol{k}} = \frac{\mathbf{Cr}}{\boldsymbol{c}}\, \Delta \boldsymbol{x}_{\boldsymbol{min}}^{\boldsymbol{k}}$$

Where Cr = Courant number, c – wave velocity and Δx_{min}^{k} – Distance between the nodes.

The wave velocity is one of the most decisive factors which have to be considered while choosing the time stepping scheme. If in case, we choose a random time step which is not bounded by the systems eigenvalues i.e., greater than h/|c|, then it means that the wave will travel faster than the element size and hence we would not have the right solution being captured.

Task 2:

Before getting into the discussion about convergence plot, it would be quiet interesting to discuss about the flux terms and the code snippets, in order to get better clarity over the program. Reference programs (AdvecDriver1D.m, Advec1D.m and AdvecRHS1D.m) are from the lecture book **"Nodal Discontinuous Galerkin Methods – Hesthaven & Warburton"**.

The equation that we would have after integrating by parts and simplifying the weak form will be

$$\frac{\mathbf{d}\boldsymbol{v}_h^k}{\mathbf{dt}} = \frac{1}{\rho}(\boldsymbol{M}^k)^{-1}\boldsymbol{S}^T\boldsymbol{p}_h^k - \boldsymbol{f}_p^{k,*} \quad \text{-------(2)}$$

$$\frac{\mathbf{d}\boldsymbol{p}_h^k}{\mathbf{dt}} = \rho c^2(\boldsymbol{M}^k)^{-1}(\boldsymbol{S}^k)^T\boldsymbol{v}_h^k - \boldsymbol{f}_v^{k,*} \text{ -----(3)}$$

where $f_v^{k,*}$ and $f_p^{k,*}$ are the flux terms where we make use **of Lax-friedrichs flux and Hybridizable discontinuous Galerkin flux**. (We would resume back to the discussion for the formulation of these terms in a while.). M^k

For the implementation of $(M^k)^{-1}$, we make use of the relation,

$$(M^k)^{-1} = VV^T$$

Where V is the **Vandermonde matrix.** Similarly for the Stiffness matrix the codes of the lecture book implement it as follows:

$$M^k D_r = S^k \Rightarrow (VV^T)^{-1}D_r = S^k \Rightarrow (S^k)^T = (D_r)^T((VV^T)^{-1})^T$$

Where $\boldsymbol{D_r}$ **is the differentiation matrix**. The mass and the stiffness matrix are declared as Global variables in **Global1D.m** and defined in the **Startup1D.m**

```
% M inverse
M_inv = (V*V');

% S transpose
S_trans = Dr'*((V*V')^-1)';
```

Startup1D.m

Now, the right hand side (RHS) of the equations (2) and (3) are defined as

```
rhsv = rx.*(M_inv*S_trans*p_dash) - LIFT*(Fscale.*dp);
rhsp = rx.*(M_inv*S_trans*v_dash) - LIFT*(Fscale.*du);
```

AkuRHS1D.m

where dp and du are the flux terms and p_dash ,dp, v and dv all have the constants $\frac{1}{\rho}$ and ρc^2 incorporated in the following as:

```
p_dash = (1./rho).*p;
v_dash = rho.*(c.^2).*vel;
```

AkuRHS1D.m

Resuming back to the flux terms, we directly write the flux terms as given to us. But at the boundaries, the required boundary conditions have to be applied.

Applying Dirichlet boundary conditions ($p_d = 0$)

$p^+ = -p^- + 2p_d$, we get, $\boldsymbol{p}^+ = -\boldsymbol{p}^-$. Using this and applying a do-nothing condition to the velocity $\boldsymbol{v}^+ = \boldsymbol{v}^-$**,** the given **LF flux** modifies to 4 conditions at the boundaries:

$\left(\frac{p}{\rho}\right)^{*} n^{-} = 0$ ($both\ at\ the\ inlet\ and\ the\ outlet\ bouncndary\ terms$);

$(\rho c^2 v)^{*} n^{-} = (\rho c^2 v^{-}) n^{-} + \max(c^{-}, c^{+}) p^{-}$;

$(\rho c^2 v)^{*} n^{+} = (\rho c^2 v^{+}) n^{+} + \max(c^{-}, c^{+}) p^{+}$;

They are translated into MATLAB code as:

```
%Boundary condition %Dirichlet conditions
 du (mapI) = (v_dash(vmapI))*nx(mapI) + (p_c(vmapI));
 du (mapO) = (v_dash(vmapO))*nx(mapO) + (p_c(vmapI));;

 dp (mapI) = 0;
 dp (mapO) = 0;
```

AkuRHS1D.m

And for the HDG flux, we end up having the same boundary conditions when we solve the flux terms. For 1-D problems, the LF Flux and the HDG flux are the same. In the HDG flux, The only term which is different from that of the LF flux is the first term in pressure flux has $(n^{-}.n^{-})$ which is just always equal to 1. Thus the LF and HDG are completely identical.

This gives us an idea that the **errors and the convergence behavior should be the same.**

To determine the error, we make use of the maximum error i.e., the infinity norm of the error vector. Here, the error is evaluated for the pressure values.

$$||e||_{\infty} = \max|e_i|$$

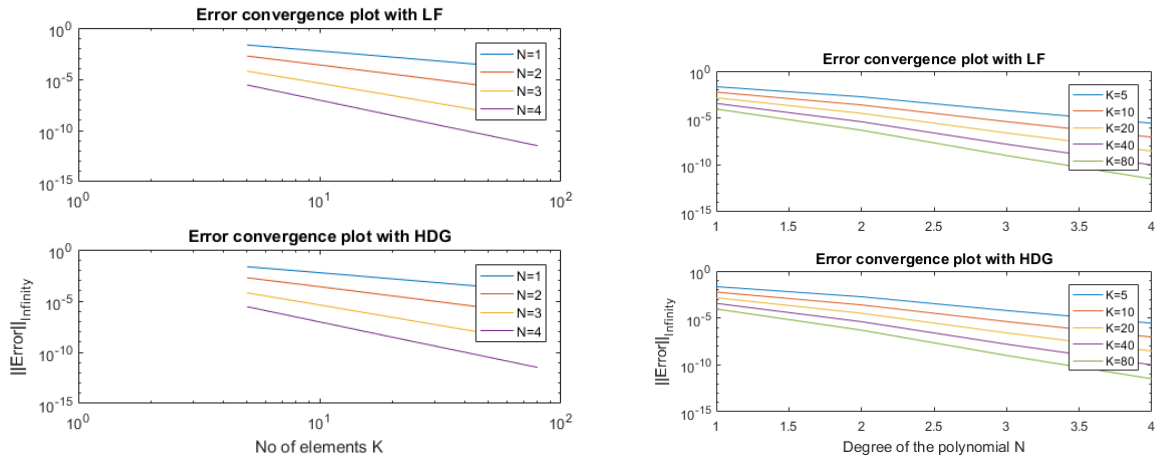
The plots clearly show that the error norm Vs K (no of elements) for varying N follows the convergence order:

$$||p - p_h||_{\infty} \leq Ch^{N+1}|p|\ ,\ \text{where}\ ||p - p_h||_{\infty} = \ ||e||_{\infty}$$

The N+1 convergence rate can be found out by analyzing the convergence rate values. The very same values are observed with both LF flux as well as HDG flux.

No of Elements(K)	N=1		N=2		N=3		N=4	
	Error	Rate	Error	Rate	Error	Rate	Error	Rate
5	0.02204	---	0.001847	---	6.23E-05	---	2.80E-06	---
10	0.005757	1.9367	0.000249	2.8907	4.12E-06	3.9189	9.18E-08	4.9301
20	0.001452	1.9875	3.20E-05	2.9604	2.57E-07	4.002	2.96E-09	4.956
40	0.000364	1.9978	4.03E-06	2.9897	1.60E-08	4.0056	9.97E-11	4.8902
80	9.09E-05	1.9998	5.04E-07	2.9974	1.00E-09	3.9996	3.29E-12	4.9236

Table 1.Error and convergence rate for different values of K and N



Left:Fig 1(a). loglog plot of Error norm Vs. No. of Elements; Right: Fig1 (b). Semi log plot of Error norm Vs. polynomial degree

Whereas if we plot the error values Vs. degree of the polynomials for varying K values, we would observe spectral convergence i.e., a very faster convergence, which is why we make use of a semilog plot for the latter and loglog plot in the former case. This in turn enlightens us with a fact that **in order to have faster(exponential) convergence rate it is better to use higher order polynomial.**

Task 3:

As we discussed earlier in task 1, we would expect two waves travelling with velocities c and –c respectively i.e., two waves travelling in left and right directions. An exponential function is made use for the pressure wave as the initial condition. We makes use of the same HDG Flux and implement absorbing as well as Dirichlet boundary conditions with HDG Flux.

For the absorbing boundary conditions, we have,

$p^+ = -p^- + 2c\rho v^-.n^-$ and $v^+ = v^-$, the given **HDG flux** modifies to 4 conditions at the boundaries:

$$\left(\frac{p}{\rho}\right)^* n^- = (cv^-)n^-.\ n^-\ ;\ \left(\frac{p}{\rho}\right)^* n^+ = (cv^+)n^+.\ n^+$$

$$\left(\rho c^2 v\right)^* n^- = \ (\rho c^2 v^-)n^- + \ (c^- p^- - \ (\rho c^2 v^-)n^-);$$
$$\left(\rho c^2 v\right)^* n^+ = \ (\rho c^2 v^+)n^+ + (c^+ p^+ - \ (\rho c^2 v^+)n^+)\ .$$

```
%Boundary condition %Absorbing boundary conditions)

  dp (mapI) = v_c(vmapI).*nx(mapI).*nx(mapI);
  dp (mapO) = v_c(vmapO).*nx(mapO).*nx(mapO);

  du (mapI) = (v_dash(vmapI)).*nx(mapI) + (p_c(vmapI) - (v_dash(vmapI).*nx(mapI)));
  du (mapO) = (v_dash(vmapO)).*nx(mapO) + (p_c(vmapO) - (v_dash(vmapO).*nx(mapO))) ;
```

AkuRHS1D_HDG_03.m

The next very important discussion is about the justification in choosing the values for N , K and time step size. As discussed in the task 1, the time step size has to be in the order such that it is bounded by the eigenvalue of the system. Here we choose to be 10^{-6}

It is definitely intuitive to select a higher order polynomial here in this case for two good reasons:

(i) Our initial condition for the pressure is a smooth exponential function. Hence it is practical to think of a wave motion which is supposed to be smooth i.e., **choosing a higher order polynomial**. Or else lower degree polynomials give a result like in the following:

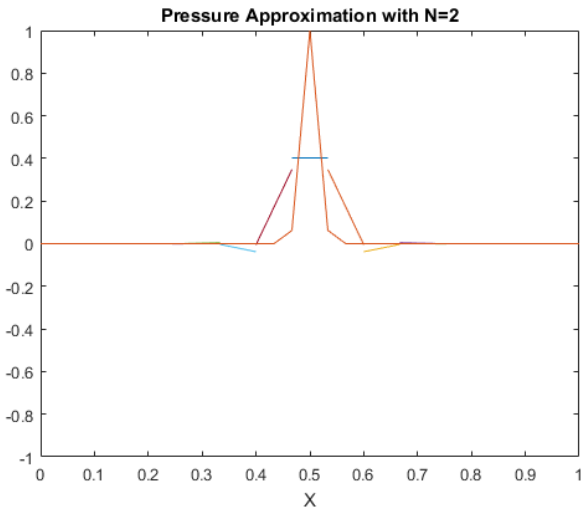


Fig 2. Pressure Vs. X Plot for N =2

(ii) In our task 2, we saw that preferring a higher order polynomial approximation leads us to have a faster convergence.

Hence choosing a higher order polynomial with lower number of elements would be the best choice than lower order polynomials with higher number of elements. But, this leaves us with a challenge that approximating higher order polynomials tend to increase the result in oscillations because of the dispersive property. Since this causes so many higher frequencies to travel at different wave speeds rather than travelling at the same wave speed c. The result for example looks like in the following:

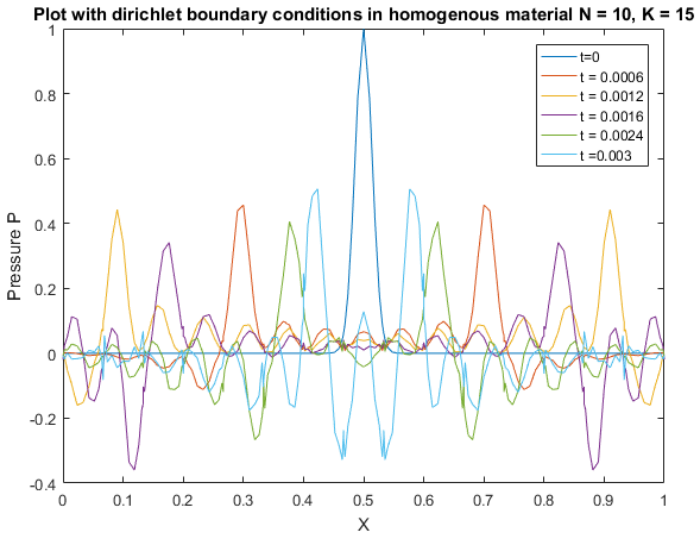


Fig.3 Graph showing pressure wavelets at different time intervals.

These oscillations could be eliminated by the use of filters. Here in the program we make use of an exponential filter code available.

```
F = Filter1D(N,floor(N/2),32);

%Applying filters
 resv = F*resv;
 resp = F*resp;

 vel = F*vel;
 p = F*p;
```

Aku1D_HDG_03.m

Now the result becomes very clear:

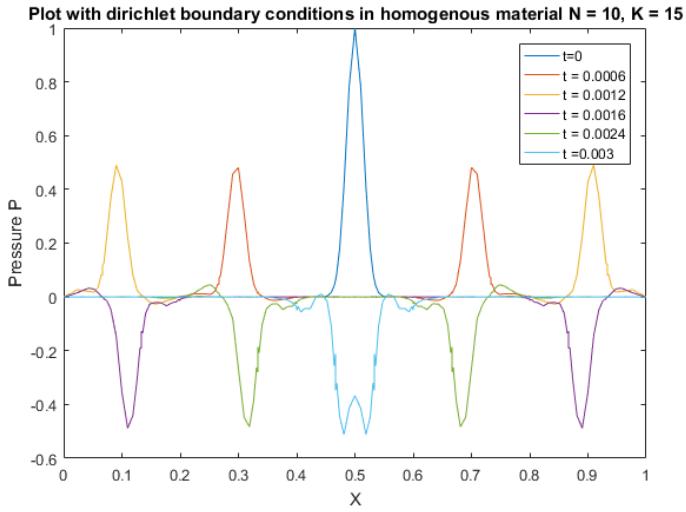


Fig 4. Plot of pressure variation in an homogenous material with c= 340, ρ = 1.2 with filters for dirichlet boundary conditions.

Similarly for absorbing boundary conditions:

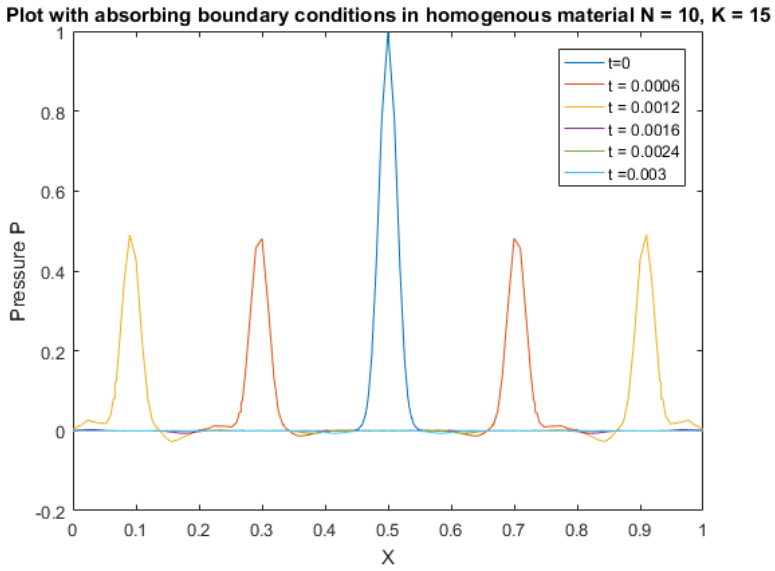


Fig 4. Plot of pressure variation in an homogenous material with c= 340, ρ = 1.2 with filters for absorbing boundary conditions.

From the two graphs, the physical interpretation becomes very clear that the **dirichlet boundary conditions** behave like the waves are fixed at both the ends, Thus they get reflected back into the medium in the opposite direction. This means that they **simulate closed boundaries.** Whereas, in the **absorbing boundary conditions**, the waves get absorbed at the boundaries as though some sinks (sponge like layers) are available at both the boundaries, which absorb all the non-physical reflections. This in turn explains that they simulate **open boundary conditions** where the pressure waves can leave the computational domain without having to get reflected back into the medium.

Task 4:

Now instead of the homogenous material, we are making use of the heterogeneous boundary layers. This explains one of the most important applications of DG-FEM where the properties of the material at the element boundaries could be captured boundaries could be incorporated. In our case,

$c_1 = 340;\ \rho_1 = 1.2\ in\ [\frac{1}{3}\ \frac{2}{3}]\ ; c_2 = 1000;\ \rho_2 = 1.2\ in\ \left[0\ \frac{1}{3}\right] and\ \left[\frac{2}{3}\ 1\right].$

This tells us that the number of elements to be considered must a multiple of 3 so that it could easily be divided into 3 regions.

```
rho1 = [0.16*ones(1,K/3),1.2*ones(1,K/3),0.16*ones(1,K/3)];
 rho = ones(Np,1)* rho1;

 c1 = [1000*ones(1,K/3),340*ones(1,K/3),1000*ones(1,K/3)];
 c = ones(Np,1)* c1;
```

AkuRHS1D_HDG_03.m

We use the same values of N,K and time step size and the initial conditions of task 3. The results are as follows:

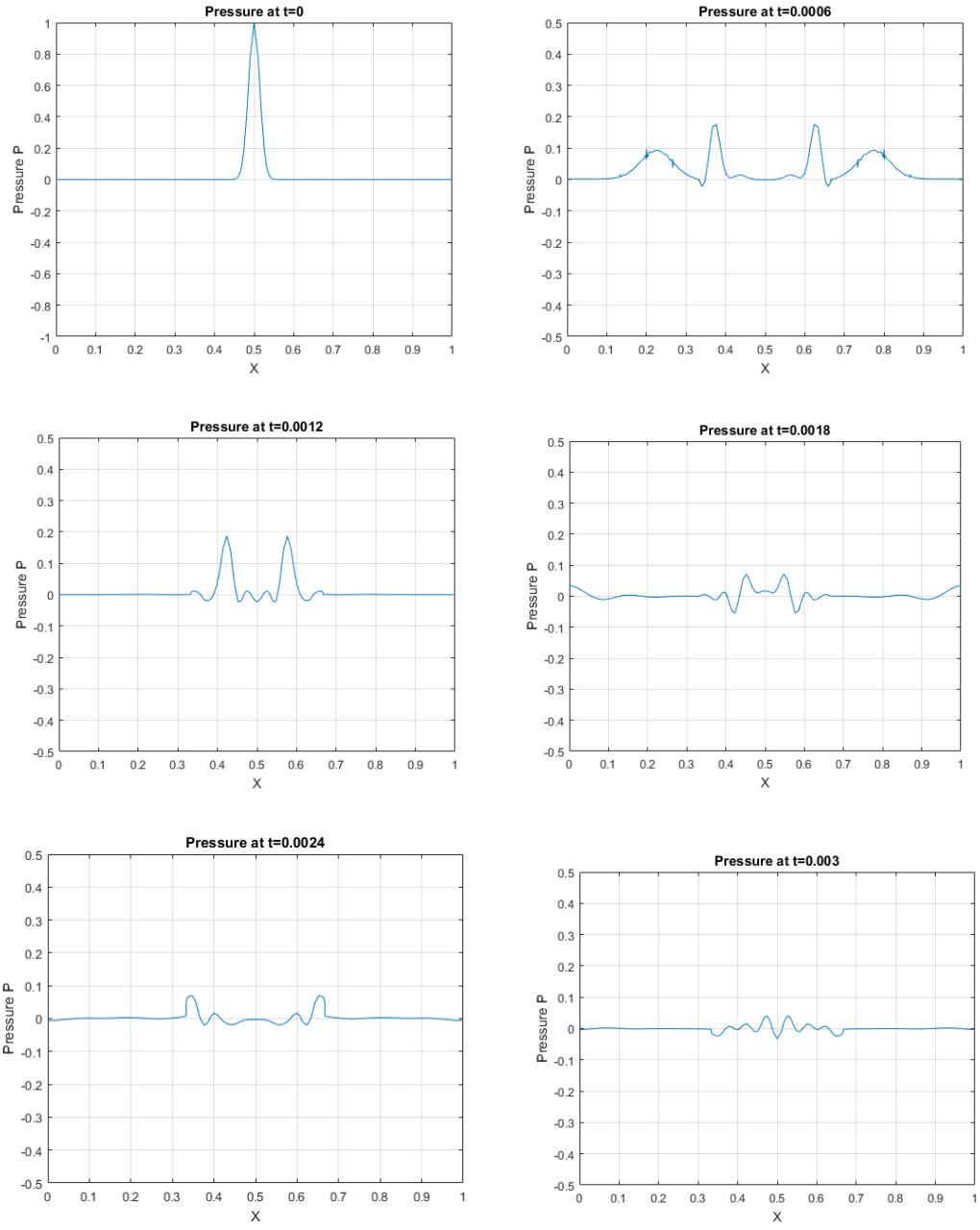


Fig 5. Pressure Vs. X plot at different time intervals showing the wave propagation for absorbing boundary conditions.

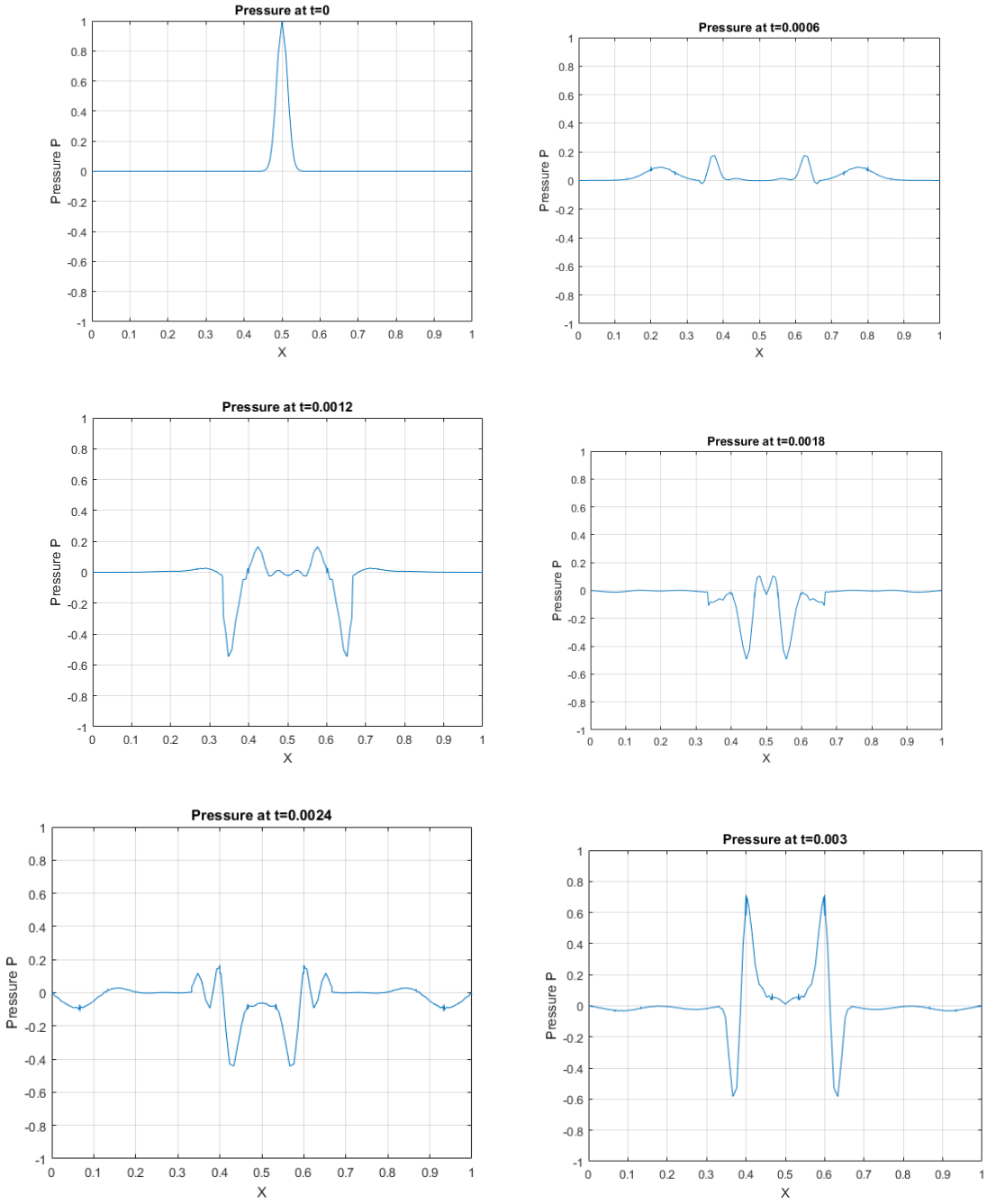


Fig 6. Pressure Vs. X plot at different time intervals showing the wave propagation for dirichlet boundary conditions

As we can clearly see from the graphs, the waves obviously travel faster in the medium where the wave velocity is 1000 i.e., at the ends. In the absorbing boundary conditions, the pressure gets completely absorbed but in the dirichlet boundary conditions they grow gradually as the wave gets reflected in the medium. The boundaries can be easily found out by the pressure Vs time graph at the boundaries. Peaks occur whenever the pressure wave crosses the boundary. The refection graphs for the absorbing boundary conditions are.

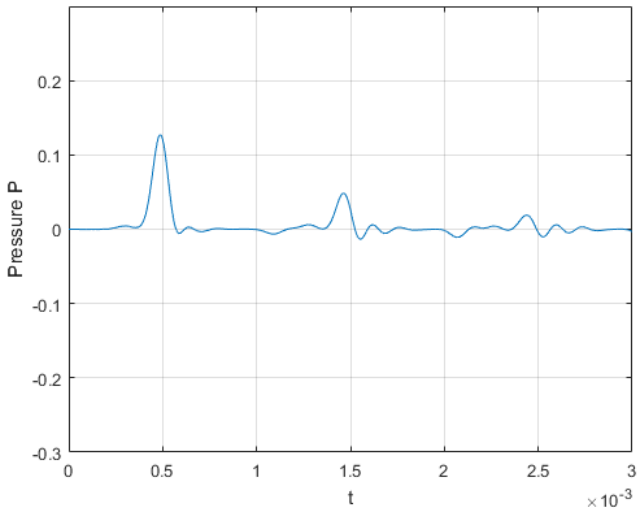


Fig 7. Pressure Vs. Time at X = 0.3333 material transition boundary where the reflections reflect the peak.