Dynamic Analysis of Unbounded Domain Problems by Applying the Scaled Boundary Finite-Element Method

A Thesis
Submitted to the
Faculty of Engineering Shoubra, Benha University
In Partial Fulfillment of the Requirements for the
Degree of Master of Science
in
Engineering Mathematics

by

Mohamed Ahmed Kamal Abd-EL-Khalik Ahmed Soliman

Demonstrator in Engineering Mathematics and Physics Department
Faculty of Engineering at Shoubra, Benha University

Faculty of Engineering-Shobra
Benha University
2015

Dynamic Analysis of Unbounded Domain Problems by Applying the Scaled Boundary Finite-Element Method

by

Mohamed Ahmed Kamal Abd-EL-Khalik Ahmed Soliman

Demonstrator in Engineering Mathematics and Physics Department
Faculty of Engineering at Shoubra, Benha University

A thesis submitted in partial fulfillment of the requirements for the degree of
Master of Science in Engineering Mathematics

Under supervision of

Prof. Dr. Abd-Elrahman Ali Saad

Professor of Applied Mathematics
Eng.Mathematics and Physics Dept.
Faculty of Engineering – Shoubra
Benha University

Dr. Emad Hassan Zahran

Lecturer of Applied Mathematics
Eng.Mathematics and Physics Dept.
Faculty of Engineering – Shoubra
Benha University

Faculty of Engineering-Shobra
Benha University
2015

APPROVAL COMMITTEE

The Committee on Final Examination recommends that the thesis by

Mohamed Ahmed Kamal Abd-El-Khalik Ahmed Soliman

entitled

Dynamic Analysis of Unbounded Domain Problems by applying the Scaled Boundary Finite-Element Method

be accepted in partial fulfillment of the requirements for the degree of
Master of Science in Engineering Mathematics

EXAMINERS COMMITTEE:

Prof. Dr. Adel Abd-Elrahman Megahed ………………………

Professor of Applied Mathematics — External Examiner
Eng. Mathematics and Physics Departement
Faculty of Engineering
Cairo University

Prof. Dr. Youssef Fawzy Rashed …………………………

Professor of Structural Analysis and Mechanics — External Examiner
Structural Eng. Department
Faculty of Engineering
Cairo University

Prof. Dr. Abd-Elrahman Ali Saad …………………………

Professor of Applied Mathematics — Supervisor
Eng. Mathematics and Physics Departement
Faculty of Engineering - Shoubra
Benha University

Date: ………………………

DEDICATION

To my father,

my mother

and my Brother,

ACKNOWLEDGEMENT

First of all due thanks go to **God** the most merciful and most graceful. Who without his guidance and inspiration nothing could have been accomplished.

I would like also to thank my professors in my college for the advices and support. I would like to express my gratitude to everyone who contributed, in different ways, to completion of this work. Inevitably some names will be missing here.

I wish to express my great appreciation and thanks to **Prof. Dr. Abd-Elrahman Saad**, Professor of Applied Mathematics, Engineering Mathematics and Physics Departement, Faculty of Engineering Shoubra, Benha University, for his kind guidance, valuable advice, sincere fatherhood, and continuous caring during this research.

I also wish to express my deep indebtedness to **Prof. Dr. Nayer Ahmed EL-Esnawy,** Professor of Structural Analysis and Mechanics, Structural Engineering Department, Faculty of Engineering, Cairo University, for his generous guidance and encouraging, sincere help, valuable suggestions, and precise advice through all stages of this research work.

Sincere thanks go **Dr. Emad H. Zahran**, Lecturer of Applied Mathematics, Engineering Mathematics and Physics Departement , Faculty of Engineering - Shoubra, Benha University, for his guidance and advice in some of the mathematical problems in my research.

I want to thank here Dr. Tianyun Liu from the Department of Hydraulic Engineering, Tsinghua University, Beijing, China, who provided me with a FORTRAN code [25] which gives the analytical solution of Lamb`s problem

Dynamic Analysis of Unbounded Domain Problems by applying the Scaled Boundary Finite-Element Method

by

Mohamed Ahmed Kamal Abd-EL-Khalik Ahmed Soliman

An Abstract of

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science in Engineering Mathematics

Faculty of Engineering -Shoubra

Benha University

This Thesis presents an explanation of the scaled boundary finite element method (SBFEM) as a method to solve unbounded domain problems subjected to dynamic loads with application on dynamic soil structure interaction problems in time domain. A complete derivation of the SBFE equations for unit impulse response function $M^{\infty}(t)$ is made; the equations are solved numerically assuming the unit impulse response function as piece wise linear taking the advantage that $M^{\infty}(t)$ becomes nearly linear after a certain time [22], where the time step used in calculating the unit impulse is greater than that used in calculating the response, which reduces the time of analysis greatly. The response of the unbounded domain is calculated, where the force displacement relationship is calculated at each time step using integration by parts [22]. Also coupling between the finite element method (FEM) and the SBFEM is used.

A program made by the author is used to solve the problems using the SBFEM and the coupled FE/SBFE methods using MATLAB. Some benchmark problems are solved as a validation; also some practical problems are solved to show the

efficiency of the method, where the solution is checked using Finite element method. Good agreement is achieved between results in case of benchmark problems and the practical problems. So, the scaled boundary finite element method is an acceptable method in modelling soil structure interaction problems and also it saves a lot of time in comparison to finite element method.

Table of Contents

LIST OF FIGURES

LIST OF SYMBOLS AND ABREVIATIONS

I	$\sqrt{-1}$
$\sigma_{i,j}, \tau_{i,j}, \varepsilon_{i,j}, \gamma_{i,j}$	Normal stress, shear stress, normal strain and shear strain
$[D]$, E, G, ν, ρ	Elasticity matrix, Young`s modulus, shear modulus, Poisson`s ratio and mass density
ω and t	Circular frequency and time
$\hat{x}, \hat{y}, \hat{z}$ and x, y, z	The Cartesian coordinates of any point in space, and the Cartesian coordinates of any point on the structure medium interface
ξ, η, ζ	The curvilinear coordinates
R	Characteristic length
$[M]$, $[C]$ and $[K]$	The mass, damping and stiffness matrices
$[S^{\infty}(\omega, r)]$ and $[M^{\infty}(t, r)]$	The dynamic stiffness for displacement and the unit impulse response function for acceleration
$[C_{\infty}]$, $[K_{\infty}]$ and $[K^{\infty}]$	Damping, static stiffness at $t \to 0$ and static stiffness at $t \to \infty$
$\{R(t, r)\}$	Reaction on the unbounded domain
a, c_s and c_p	The dimensionless frequency, shear (transverse) wave velocity, and longitudinal (primary) wave velocity
$[E_o]$, $[E_1]$, $[E_2]$ and $[M_o]$	The scaled boundary finite element constants
FEM	Finite element method
SBFEM	Scale boundary finite element method
Dof	Degree of freedom

Chapter 1: Introduction and Background

1.1 Problem statement and background

Dynamic analysis of unbounded domain-structure interaction occurs in many fields of Engineering and physical science such as *Soil structure interaction*, *Fluid structure interaction*, *Diffusion*, *Electromagnetism*, etc. In *Soil structure interaction* (SSI) as an example the effect of soil on the dynamic behaviour of the structure is important especially when there is problems in the soil itself (Liquefaction for example), or the structure is subjected to special loads as for Nuclear power stations and bridges. This problem can be expressed using partial differential equations (PDE`s) and then solved to get the needed data, as the analytical solution of these equations is easily obtained only for small scale problems so these equations are solved numerically.

The two main problems concerning dealing with unbounded domain are:

1-The infinite dimensions of the domain and

2- The wave propagation must tend to zero as the wave moves to infinity, where the energy must tends to zero (*radiation condition*).

There are two approaches to solve the problem first is the *direct approach*: where the structure and part of the unbounded domain is modelled and truncated at an arbitrary distance to account for infinity, where at this arbitrary distance boundary conditions (B.C.s) needed to be defined and here the second problem arises, if we define the B.C.s as rigid supports so when the wave propagates it will reflect from this support and return again to the medium causing unrealistic behaviour, which can be illustrated by the following example [19]: suppose the unbounded domain shown in fig(1.1) which represents a semi-infinite rod is subjected to a force R(t) at A as shown (fig(1.1a)) so a wave will propagate through the rod, this wave will continue to infinity as shown by fig(1.1b,c).Now if we try to solve the problem numerically and truncate the rod at a certain distance and define a support (B.C. u(t)=0 at B) as shown in fig(1.2), when the load acts at A, the wave will propagate through the rod the same as in the original problem as shown in fig(1.2b) but when it reaches B where the rod is truncated the wave will reflects and return through the rod to A (fig(1.2c)) making a error in the response. So the response at A for example will be correct in the time before the reflected wave return to it.

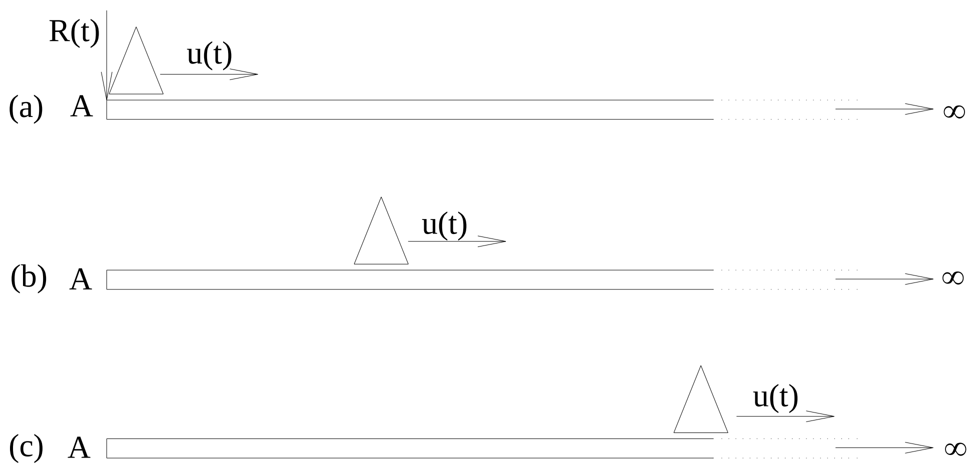


Fig.(1.1): Propagation of wave through an unbounded domain

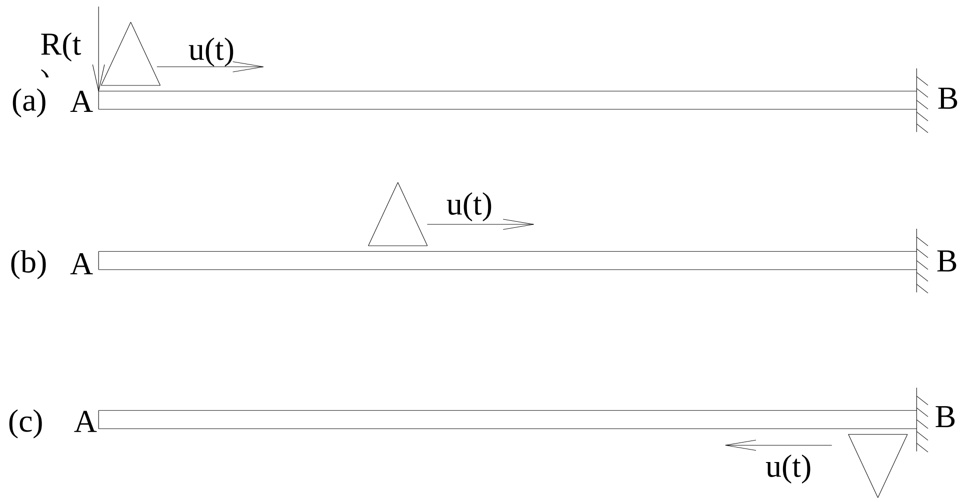


Fig.(1.2): Propagation of wave through a truncated domain

So to solve this problem: 1- absorbent (viscous) boundary conditions may be defined where they absorb the waves so as to minimizes the reflected wave [1], or 2- the domain can be truncated at a distance so that the waves do not reach the part under consideration at time of study, so at the time of study no reflections occur fig(1.3), where the boundaries of the problem are supported on rigid supports.

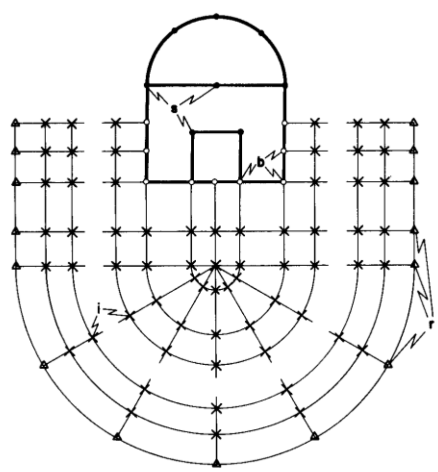


Fig.(1.3): A structure supported on an unbounded domain modeled with finite element and truncated at a certain distance where there is supports

The second approach is the *sub-structuring approach* fig(1.4) where the structure and a small part of the domain is modelled (the part of interest) which is known as the *near field* and the effect of the unbounded domain-which is known as the *far field*-on the problem is taken using the force-acceleration relationship at the boundary (structure medium (domain) interface or simply interface) which is represented by the unit impulse response function in time domain analysis or by dynamic stiffness in frequency domain analysis.

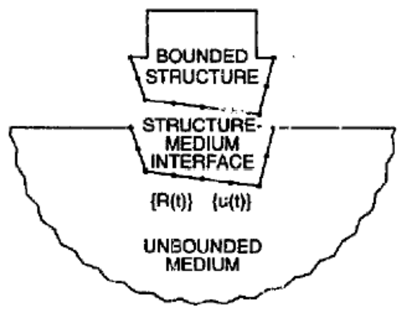


Fig.(1.4): The bounded and the unbounded substructures and the interface at which the force is defined

The governing equation of motion for the structure can be written in time domain as:

$$[M]\{\ddot{u}(t)\}+[C]\{\dot{u}(t)\}+[K]\{u(t)\}=\{P(t)\}-\{R(t)\} \tag{1.1}$$

or in frequency domain as:

$$\left((i\omega)^2[M]+(i\omega)[C]+[K]\right)\{u(\omega)\}=\{P(\omega)\}-\{R(\omega)\} \tag{1.2}$$

where, $[M],[C]$ and $[K]$are the mass, damping and stiffness matrices of the bounded (structure) domain, $\{\ddot{u}(t)\},\{\dot{u}(t)\}$ and $\{u(t)\}$are the acceleration, velocity and displacement vectors, $\{P(t)\}$ and $\{R(t)\}$are the applied force and the reaction of the unbounded domain, respectively. The same definitions hold for the frequency domain equation but the vectors will be for the amplitudes of the acceleration, velocity, displacement and force. The vector $\{R(t)\}$can be expressed-as said above-in terms of the unit impulse function for acceleration $\left[M^{\infty}(\tau)\right]$ [7] as;

$$\{R(t)\} = \int_0^t [M^{\infty}(\tau)]\{\ddot{u}(t-\tau)\}d\tau \tag{1.3}$$

Also, $\{R(\omega)\}$ (ω is the circular frequency) can be expressed in terms of the dynamic stiffness for acceleration [7] as:

$$\{R(\omega)\} = (i\omega)^2 [M^{\infty}(\omega)]\{u(\omega)\} \tag{1.4}$$

where, the superscript ∞ stands for the unbounded domain.

In the above equations the independent variable that concerns for location with respect to coordinates are removed for simplicity.

The near field (bounded domain) is usually solved with finite element method (FEM)and the effect of the far field (unbounded domain) is taken using boundary element or scaled boundary finite element (SBFEM),for example, where the unit impulse or the dynamic stiffness is calculated then using the above equations we get the response of the system.

The *Scaled boundary finite element method* (SBFEM) [8] and [12] is a semi-analytical method made by Prof. J.P. Wolf and Prof. Chongmin Song in 1995 under the name *Consistent infinitesimal finite element cell method* ([5]. [6] and [7]), the name SBFEM appears first in 1996 [8]. In this method the boundary (interface) is discretized with surface elements only without any discretization inside the domain, a curvilinear set of coordinates is defined on the boundary (ξ (radial direction),(η,ζ) (circumferential direction)) fig.(1.5, 1.6) where the governing equations are weakened in two directions (η,ζ) by assuming shape functions in these directions, so the method remains exact in the radial direction and approximate in the circumferential direction.

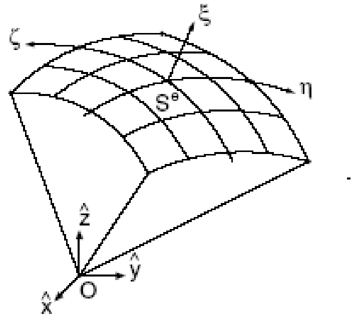


Fig.(1.5): Curvilinear and Cartesian coordinates with surface elements

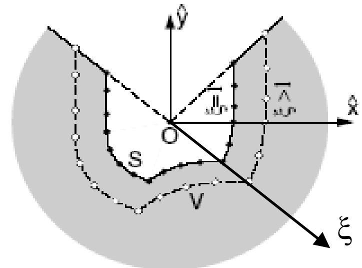


Fig.(1.6): The boundary ξ=1 and the similar boundaries ξ>1

Therefore, the problem of extension to infinity is solved. The radiation condition is satisfied by taking the damping as +ve value so that the rate of change of energy with respect to time is minimized as the wave moves towards infinity. We can get the solution at any similar surface other than the boundary by scaling with ξ fig.(1.6).The SBFEM gives two equations in frequency one for the displacement which is a second order linear ordinary differential equation, the other is for dynamic stiffness which is a first order non-linear ordinary differential equation.

In the SBFEM there is a scaling center at which the boundary is scaled with ξ to get the other points in the domain. The side faces which passes through the scaling center is by definition need not be discretized.

The SBFEM solves the problem of unbounded domain in time domain, as an example, in [9] the unit impulse response function for velocity is calculated, in [18] and [19] the unit impulse response function for acceleration is calculated for non-homogeneous domain and the reduced set of base functions are used to reduce the number of degrees of freedom, in [16] the unit impulse response function for acceleration is calculated only for the non-linear part and the values of the linear part of the function is calculated using extrapolation, in [22] the same is done as [16] but the unit impulse assumed to be piece-wise linear not constant as before and it is calculated at larger time steps than the displacement which reduces time also the force displacement relationship (equation (1.3)) is calculated using integration by parts in each time step, in [23] the unit impulse for displacement is calculated where it is calculated only till the time at which the regular part tends to zero, also the rate of change of the regular part of the unit impulse for displacement is assumed to be piece-wise linear.

The SBFEM solves the problem of unbounded domain also in frequency domain, as an example, in [10] the SBFE equations are solved analytically in the frequency domain where a first order differential equation in displacement and internal force is formulated using the SBFE equation in displacement and the relation between the internal force and the displacement, in [17] and [19] the SBFE equation for dynamic stiffness for displacement is solved numerically (for non-homogeneous domain) using Bulirsch-Stoer method and the starting value is obtained using the asymptotic expansion for high frequency.

Here is a table to see the advantage of the SBFEM over the FEM.

Points of comparison	Finite element method	Scaled boundary finite element method
Reduction of spatial dimension by one	---------------	√
No fundamental solution required	√	√
Suitable for anisotropic materials	√	√
Radiation condition satisfied	√	√
No discretization of side faces passing through the scaling center	---------------	√

1.2 Thesis objectives

The objective of this thesis is to show the SBFEM as a method to solve dynamic soil structure interaction problems in time domain, using the approximations in [22] (for homogeneous, isotropic and linearly elastic domains) and apply it to some practical problems.

1.3 Thesis organization

This thesis consists of five chapters after this chapter. These chapters contain the followings:

Chapter 2: The explanation of the SBFEM and the derivation of its equations and how it is coupled with the FEM.

Chapter 3: The program made by the author is explained.

Chapter 4: Some benchmark problems are solved as a validation of the program.

Chapter 5: Some practical problems are solved using coupled FE/SBFE.

Chapter 6: Summary and conclusion are stated.

Chapter 2: Scaled boundary finite element method for unbounded domain

2.1. Introduction

In this chapter, the derivation of the scaled boundary finite element (SBFE) equations for both displacements and dynamic stiffness for elastodynamics problems are stated in frequency and time domain [8], [11] (and [14]), also the numerical solution is stated [22]. The domain here is assumed isotropic, homogeneous and linearly elastic.

In this method the structure medium interface is discretized only. Shape functions are assumed in direction of the structure medium interface so the governing partial differential equations are weakened in these directions only (circumferential) but in the inner direction (radial) the solution is exact.

2.2. The governing equations for elastodynamics

The governing equations will be written in Cartesian coordinates $(\hat{x}, \hat{y}, \hat{z})$, where the circumflex indicates that the coordinates are for any point in space but if we write (x, y, z) it indicates the points on the structure medium interface

The governing equations for elastodynamics in frequency domain expressed in the displacement amplitude vector

$$\{u\} = \{u(\hat{x}, \hat{y}, \hat{z}, \omega)\} = \begin{bmatrix} u_x & u_y & u_z \end{bmatrix}^T \tag{2.1}$$

are:

$$[L]^T \{\sigma(\hat{x}, \hat{y}, \hat{z}, \omega)\} + \{p(\hat{x}, \hat{y}, \hat{z}, \omega)\} + \omega^2 \rho \{u(\hat{x}, \hat{y}, \hat{z}, \omega)\} = \{0\} \tag{2.2}$$

where, $\{p\}$ is the body force, ρ is the medium density, ω is the frequency, $\{\sigma\}$is the stress vector which can be written from Hook`s law as:

$$\{\sigma(\hat{x}, \hat{y}, \hat{z}, \omega)\} = [D]\{\varepsilon(\hat{x}, \hat{y}, \hat{z}, \omega)\} = \begin{bmatrix} \sigma_x & \sigma_y & \sigma_z & \tau_{yz} & \tau_{xz} & \tau_{xy} \end{bmatrix}^T \tag{2.3}$$

where, $[D]$ is the elasticity matrix (see [13] appendix A)and $\{\varepsilon\}$ is the strain amplitude vector

$$\{\varepsilon(\hat{x},\hat{y},\hat{z},\omega)\} = [L]\{u(\hat{x},\hat{y},\hat{z},\omega)\} = \begin{bmatrix}\varepsilon_x & \varepsilon_y & \varepsilon_z & \gamma_{yz} & \gamma_{xz} & \gamma_{xy}\end{bmatrix}^T \tag{2.4}$$

$(\hat{x},\hat{y},\hat{z},\omega)$ will be remove for conciseness.

$$\text{where } [L] = \begin{bmatrix} \partial/\partial\hat{x} & 0 & 0 \\ 0 & \partial/\partial\hat{y} & 0 \\ 0 & 0 & \partial/\partial\hat{z} \\ 0 & \partial/\partial\hat{z} & \partial/\partial\hat{y} \\ \partial/\partial\hat{z} & 0 & \partial/\partial\hat{x} \\ \partial/\partial\hat{y} & \partial/\partial\hat{x} & 0 \end{bmatrix} \tag{2.5}$$

2.3. The scaled boundary transformation

In order to simplify the governing equations, the structure medium interface is discretized with surface elements, where the points of the elements are related to each other with assumed shape functions. Curvilinear coordinate (ξ,η,ζ) are introduced at the interface, where, the coordinates (η,ζ) are parallel to the interface (which is named as the circumferential direction) and ξ (which is named as the radial direction) is from the origin to any other point in space see fig (2.1, 2.2), so the dicretization is done for the interface only which make the solution exact in the ξ direction and approximate in the circumferential direction ($\xi \geq 1$, $\xi=1$ at the interface)

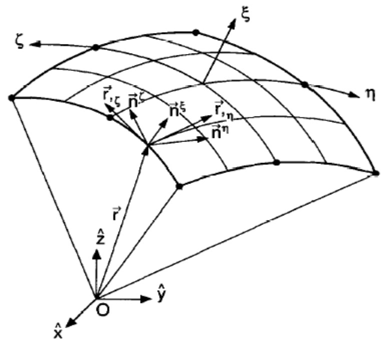


Fig.(2.1): Curvilinear and Cartesian coordinates with surface elements

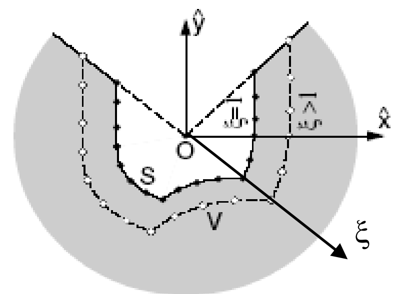


Fig.(2.2): The boundary ξ=1 and the similar boundaries ξ>1

The coordinates on the medium interface can be written as follows:

$$\begin{aligned} x(1,\eta,\zeta) &= [N(\eta,\zeta)]\{x\} \\ y(1,\eta,\zeta) &= [N(\eta,\zeta)]\{y\} \\ z(1,\eta,\zeta) &= [N(\eta,\zeta)]\{z\} \end{aligned} \qquad (2.6)$$

$[N(\eta,\zeta)]$ is the assumed shape function, $\{x\}$, $\{y\}$ and $\{z\}$ are the coordinates of the element nodes (For the assumed shape functions see chapter 3).

The coordinates at any other point in space can be obtained by scaling using ξ

$$\begin{aligned} \hat{x}(\xi,\eta,\zeta) &= \xi[N(\eta,\zeta)]\{x\} = \xi x(\xi,\eta,\zeta) \\ \hat{y}(\xi,\eta,\zeta) &= \xi[N(\eta,\zeta)]\{y\} = \xi y(\xi,\eta,\zeta) \\ \hat{z}(\xi,\eta,\zeta) &= \xi[N(\eta,\zeta)]\{z\} = \xi z(\xi,\eta,\zeta) \end{aligned} \qquad (2.7)$$

So at any certain ξ there will be only two coordinate (η,ζ)

Also, it will be assumed that the displacement at any point is related to that at the nodal points with shape functions the same as for coordinates, therefore the displacement amplitude is written as:

$$\{u(\xi,\eta,\zeta,\omega)\} = \left[u_x(\xi,\eta,\zeta,\omega) \quad u_y(\xi,\eta,\zeta,\omega) \quad u_z(\xi,\eta,\zeta,\omega)\right]^T = [N(\eta,\zeta)]\{u(\xi,\omega)\},$$

where $\{u(\xi,\omega)\}$ is the nodal displacement amplitude, so the displacement is approximated in the circumferential direction and is exact in the radial one.

We will now change the differential operator in the governing equation (2.2) from differentiating with respect to (x, y, z) to (ξ,η,ζ).

First get the Jacobian:

$$[J(\xi,\eta,\zeta)]=[\hat{J}]=\begin{bmatrix}\hat{x}_{,\xi} & \hat{y}_{,\xi} & \hat{z}_{,\xi}\\ \hat{x}_{,\eta} & \hat{y}_{,\eta} & \hat{z}_{,\eta}\\ \hat{x}_{,\zeta} & \hat{y}_{,\zeta} & \hat{z}_{,\zeta}\end{bmatrix}$$

Using equations (2.7)

$$[J(\xi,\eta,\zeta)]=[\hat{J}]=\begin{bmatrix}x & y & z\\ \xi x_{,\eta} & \xi y_{,\eta} & \xi z_{,\eta}\\ \xi x_{,\zeta} & \xi y_{,\zeta} & \xi z_{,\zeta}\end{bmatrix}=\begin{bmatrix}1 & 0 & 0\\ 0 & \xi & 0\\ 0 & 0 & \xi\end{bmatrix}\begin{bmatrix}x & y & z\\ x_{,\eta} & y_{,\eta} & z_{,\eta}\\ x_{,\zeta} & y_{,\zeta} & z_{,\zeta}\end{bmatrix} \tag{2.8}$$

Let, $[J(\eta,\zeta)]=[J]=\begin{bmatrix}x & y & z\\ x_{,\eta} & y_{,\eta} & z_{,\eta}\\ x_{,\zeta} & y_{,\zeta} & z_{,\zeta}\end{bmatrix}$ (2.9)

$$[\hat{J}]^{-1}=[J]^{-1}\begin{bmatrix}1 & 0 & 0\\ 0 & 1/\xi & 0\\ 0 & 0 & 1/\xi\end{bmatrix}$$

Therefore, the differentiation with respect to x, y and z can know be written:

$$\begin{Bmatrix}\dfrac{\partial}{\partial\hat{x}}\\ \dfrac{\partial}{\partial\hat{y}}\\ \dfrac{\partial}{\partial\hat{z}}\end{Bmatrix}=[\hat{J}]^{-1}\begin{Bmatrix}\dfrac{\partial}{\partial\xi}\\ \dfrac{\partial}{\partial\eta}\\ \dfrac{\partial}{\partial\zeta}\end{Bmatrix}=[J]^{-1}\begin{Bmatrix}\dfrac{\partial}{\partial\xi}\\ \dfrac{1}{\xi}\dfrac{\partial}{\partial\eta}\\ \dfrac{1}{\xi}\dfrac{\partial}{\partial\zeta}\end{Bmatrix} \tag{2.10}$$

$$[J]^{-1}=\frac{1}{|J|}\begin{bmatrix}y_{,\eta}z_{,\zeta}-y_{,\zeta}z_{,\eta} & y_{,\zeta}z-yz_{,\zeta} & yz_{,\eta}-y_{,\eta}z\\ x_{,\zeta}z_{,\eta}-x_{,\eta}z_{,\zeta} & xz_{,\zeta}-x_{,\zeta}z & x_{,\eta}z-xz_{,\eta}\\ x_{,\eta}y_{,\zeta}-x_{,\zeta}y_{,\eta} & x_{,\zeta}y-xy_{,\zeta} & xy_{,\eta}-x_{,\eta}y\end{bmatrix} \tag{2.11}$$

where, $|J|$ is the determinant of the jacobian

Now we are going to simplify the inverse of the jacobian in order to write the $[L]$ matrix in a simple way, consider a point in space its position vector can be written as follows:

$\vec{\hat{r}} = \hat{x}\vec{i} + \hat{y}\vec{j} + \hat{z}\vec{k}$

Using equation (2.7)

$$\vec{\hat{r}} = \xi(x\vec{i} + y\vec{j} + z\vec{k}) = \xi\vec{r} \quad (2.12)$$

Therefore, $\vec{\hat{r}}_{,\xi} = (x\vec{i} + y\vec{j} + z\vec{k}) = \xi\vec{r}$ (2.13)

and, $\vec{\hat{r}}_{,\eta} = \xi(x_{,\eta}\vec{i} + y_{,\eta}\vec{j} + z_{,\eta}\vec{k}) = \xi\vec{r}_{,\eta}$ (2.14)

and, $\vec{\hat{r}}_{,\zeta} = \xi(x_{,\zeta}\vec{i} + y_{,\zeta}\vec{j} + z_{,\zeta}\vec{k}) = \xi\vec{r}_{,\zeta}$ (2.15)

The vector $\vec{\hat{r}}_{,\eta}$ and $\vec{\hat{r}}_{,\zeta}$ are the tangent to the curves ζ=0 and η=0, respectively, as shown in figure (2.1).

Let, $\vec{g}^{\xi}$, $\vec{g}^{\eta}$ and $\vec{g}^{\zeta}$ be the vectors normal to the plane (η,ζ), (ξ,ζ) and (ξ,η) at constant ξ, η and ζ, respectively.

Therefore,

$$\vec{g}^{\xi} = \vec{r}_{,\eta} \times \vec{r}_{,\zeta} = (y_{,\eta}z_{,\zeta} - y_{,\zeta}z_{,\eta})\vec{i} + (x_{,\zeta}z_{,\eta} - x_{,\eta}z_{,\zeta})\vec{j} + (x_{,\eta}y_{,\zeta} - x_{,\zeta}y_{,\eta})\vec{k} \quad (2.16)$$

$$\vec{g}^{\eta} = \vec{r}_{,\zeta} \times \vec{r} = (y_{,\zeta}z - yz_{,\zeta})\vec{i} + (xz_{,\zeta} - x_{,\zeta}z)\vec{j} + (x_{,\zeta}y - xy_{,\zeta})\vec{k} \quad (2.17)$$

$$\vec{g}^{\zeta} = \vec{r} \times \vec{r}_{,\eta} = (yz_{,\eta} - y_{,\eta}z)\vec{i} + (x_{,\eta}z - xz_{,\eta})\vec{j} + (xy_{,\eta} - x_{,\eta}y)\vec{k} \quad (2.18)$$

Therefore, the unit vectors in their directions are:

$$\vec{n}^{\xi} = \frac{\vec{g}^{\xi}}{|\vec{g}^{\xi}|} = n_x^{\xi}\vec{i} + n_y^{\xi}\vec{j} + n_z^{\xi}\vec{k} \quad \text{normal to plane } (\eta,\zeta) \quad (2.19)$$

$$\vec{n}^{\eta} = \frac{\vec{g}^{\eta}}{|\vec{g}^{\eta}|} = n_x^{\eta}\vec{i} + n_y^{\eta}\vec{j} + n_z^{\eta}\vec{k} \quad \text{normal to plane } (\xi,\zeta) \quad (2.20)$$

$$\vec{n}^{\zeta} = \frac{\vec{g}^{\zeta}}{|\vec{g}^{\zeta}|} = n_x^{\zeta}\vec{i} + n_y^{\zeta}\vec{j} + n_z^{\zeta}\vec{k} \quad \text{normal to plane } (\eta,\xi) \quad (2.21)$$

Substitute by equations (2.16, 2.17, 2.18, 2.19, 2.20, and 2.21) in (2.11)

Therefore,

$$[J]^{-1} = \frac{1}{|J|}\begin{bmatrix} |\bar{g}^{\xi}|n_x^{\xi} & |\bar{g}^{\eta}|n_x^{\eta} & |\bar{g}^{\zeta}|n_x^{\zeta} \\ |\bar{g}^{\xi}|n_y^{\xi} & |\bar{g}^{\eta}|n_y^{\eta} & |\bar{g}^{\zeta}|n_y^{\zeta} \\ |\bar{g}^{\xi}|n_z^{\xi} & |\bar{g}^{\eta}|n_z^{\eta} & |\bar{g}^{\zeta}|n_z^{\zeta} \end{bmatrix} \tag{2.22}$$

Substitute by equation (2.22) in (2.10)

$$\begin{Bmatrix} \frac{\partial}{\partial \hat{x}} \\ \frac{\partial}{\partial \hat{y}} \\ \frac{\partial}{\partial \hat{z}} \end{Bmatrix} = \frac{1}{|J|}\begin{Bmatrix} |\bar{g}^{\xi}|n_x^{\xi}\frac{\partial}{\partial \xi} + \frac{1}{\xi}|\bar{g}^{\eta}|n_x^{\eta}\frac{\partial}{\partial \eta} + \frac{1}{\xi}|\bar{g}^{\zeta}|n_x^{\zeta}\frac{\partial}{\partial \zeta} \\ |\bar{g}^{\xi}|n_y^{\xi}\frac{\partial}{\partial \xi} + \frac{1}{\xi}|\bar{g}^{\eta}|n_y^{\eta}\frac{\partial}{\partial \eta} + \frac{1}{\xi}|\bar{g}^{\zeta}|n_y^{\zeta}\frac{\partial}{\partial \zeta} \\ |\bar{g}^{\xi}|n_z^{\xi}\frac{\partial}{\partial \xi} + \frac{1}{\xi}|\bar{g}^{\eta}|n_z^{\eta}\frac{\partial}{\partial \eta} + \frac{1}{\xi}|\bar{g}^{\zeta}|n_z^{\zeta}\frac{\partial}{\partial \zeta} \end{Bmatrix} \tag{2.23}$$

Substitute by equation (2.23) in (2.5)

$$[L] = \underbrace{\frac{|\bar{g}^{\xi}|}{|J|}\begin{bmatrix} n_x^{\xi} & 0 & 0 \\ 0 & n_y^{\xi} & 0 \\ 0 & 0 & n_z^{\xi} \\ 0 & n_z^{\xi} & n_y^{\xi} \\ n_z^{\xi} & 0 & n_x^{\xi} \\ n_y^{\xi} & n_x^{\xi} & 0 \end{bmatrix}}_{[b_1(\eta,\zeta)]}\frac{\partial}{\partial \xi} + \frac{1}{\xi}\underbrace{\frac{|\bar{g}^{\eta}|}{|J|}\begin{bmatrix} n_x^{\eta} & 0 & 0 \\ 0 & n_y^{\eta} & 0 \\ 0 & 0 & n_z^{\eta} \\ 0 & n_z^{\eta} & n_y^{\eta} \\ n_z^{\eta} & 0 & n_x^{\eta} \\ n_y^{\eta} & n_x^{\eta} & 0 \end{bmatrix}}_{[b_2(\eta,\zeta)]}\frac{\partial}{\partial \eta} + \frac{1}{\xi}\underbrace{\frac{|\bar{g}^{\zeta}|}{|J|}\begin{bmatrix} n_x^{\zeta} & 0 & 0 \\ 0 & n_y^{\zeta} & 0 \\ 0 & 0 & n_z^{\zeta} \\ 0 & n_z^{\zeta} & n_y^{\zeta} \\ n_z^{\zeta} & 0 & n_x^{\zeta} \\ n_y^{\zeta} & n_x^{\zeta} & 0 \end{bmatrix}}_{[b_3(\eta,\zeta)]}\frac{\partial}{\partial \zeta} \tag{2.24}$$

Therefore,

$$[L(\xi,\eta,\zeta)] = [b_1(\eta,\zeta)]\frac{\partial}{\partial \xi} + \frac{1}{\xi}\left([b_2(\eta,\zeta)]\frac{\partial}{\partial \eta} + [b_3(\eta,\zeta)]\frac{\partial}{\partial \zeta}\right) \tag{2.25}$$

For $[L(\xi,\eta,\zeta)]$ (ξ,η,ζ) will be omitted for conciseness and also for $[b_1(\eta,\zeta)]$, $[b_2(\eta,\zeta)]$ and $[b_3(\eta,\zeta)]$ (η,ζ) will be omitted.

Also, we need to discretize the traction vector

$$\{t\} = \begin{Bmatrix} t_x \\ t_y \\ t_z \end{Bmatrix} = \begin{bmatrix} n_x & 0 & 0 & 0 & n_z & n_y \\ 0 & n_y & 0 & n_z & 0 & n_x \\ 0 & 0 & n_z & n_y & n_x & 0 \end{bmatrix}\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{yz} \\ \tau_{xz} \\ \tau_{xy} \end{Bmatrix} \tag{2.26}$$

Comparing equation (2.24) and (2.26)

$$\{t\}^{\xi} = [b_1]^T \frac{|J|}{|\bar{g}^{\xi}|}\{\sigma\} \text{ traction on surface } (\eta,\zeta) \tag{2.27}$$

$$\{t\}^{\eta} = [b_2]^T \frac{|J|}{|\bar{g}^{\eta}|}\{\sigma\} \text{ traction on surface } (\xi,\zeta) \tag{2.28}$$

$$\{t\}^{\zeta} = [b_3]^T \frac{|J|}{|\bar{g}^{\zeta}|}\{\sigma\} \text{ traction on surface } (\xi,\eta) \tag{2.29}$$

2.4. Discretization of equation of motion

Substitute by equation (2.25) in (2.2), but due to the above assumed shape functions the equation will not equal zero as in (2.2), so in order to overcome the error due to approximation the well known *weighted residual technique* (see [14] chapter 1) will be used

Therefore,

$\int_V \{w\}^T \left([b_1]^T \{\sigma_{,\xi}\} + \frac{1}{\xi}\left([b_2]^T\{\sigma_{,\eta}\} + [b_3]^T\{\sigma_{,\zeta}\}\right) + \{p\} + \omega^2 \rho \{u\} \right) dV = \{0\}$ where, $\{w\}$ is an assumed weighting vector.

$$\begin{aligned} &\int_V \{w\}^T [b_1]^T \{\sigma_{,\xi}\} dV + \int_V \frac{1}{\xi}\{w\}^T \left([b_2]^T\{\sigma_{,\eta}\} + [b_3]^T\{\sigma_{,\zeta}\}\right) dV \\ &+ \int_V \{w\}^T \{p\} dV + \int_V \{w\}^T \omega^2 \rho \{u\} dV = \{0\} \end{aligned} \tag{2.30}$$

$$dV = |\hat{J}| d\eta d\zeta d\xi \tag{2.31}$$

From equation (2.8) and (2.9) $dV = \xi^2 |J| d\eta d\zeta d\xi$ (2.32)

For the second integral:

$$I = \int_V \frac{1}{\xi}\{w\}^T \left([b_2]^T\{\sigma_{,\eta}\} + [b_3]^T\{\sigma_{,\zeta}\}\right) dV \tag{2.33}$$

$$I = \int_{\xi}\int_{\zeta}\int_{\eta} \frac{1}{\xi}\{w\}^T \left([b_2]^T\{\sigma_{,\eta}\} + [b_3]^T\{\sigma_{,\zeta}\}\right) \xi^2 |J| d\eta d\zeta d\xi$$

$$I = \int_{\xi} \xi \left(\int_{\zeta}\int_{\eta} \{w\}^T \left([b_2]^T \{\sigma_{,\eta}\} + [b_3]^T \{\sigma_{,\zeta}\} \right) |J| d\eta d\zeta \right) d\xi \tag{2.34}$$

Using integration by parts the first term in equation (2.34) can be as follows:

$$I_2 = \int_{\zeta}\int_{\eta} \{w\}^T [b_2]^T \{\sigma_{,\eta}\} |J| d\eta d\zeta = \int_{\zeta}\int_{\eta} \{w\}^T [b_2]^T |J| d\{\sigma\} d\zeta$$

$$I_2 = \int_{\zeta} \{w\}^T [b_2]^T |J| \{\sigma\} d\zeta - \int_{\zeta}\int_{\eta} \left(\{w\}^T [b_2]^T |J| \right)_{,\eta} \{\sigma\} d\eta d\zeta$$

The same will be for the second part of equation (2.34)

$$I_3 = \int_{\zeta}\int_{\eta} \{w\}^T [b_3]^T \{\sigma_{,\zeta}\} |J| d\eta d\zeta = \int_{\eta} \{w\}^T [b_3]^T |J| \{\sigma\} d\eta - \int_{\zeta}\int_{\eta} \left(\{w\}^T [b_3]^T |J| \right)_{,\zeta} \{\sigma\} d\eta d\zeta$$

Substitute by I_2 and I_3 in equation (2.34)

$$I = \int_{\xi} \xi \left(\int_{\zeta} \{w\}^T [b_2]^T |J| \{\sigma\} d\zeta + \int_{\eta} \{w\}^T [b_3]^T |J| \{\sigma\} d\eta - \int_{\zeta}\int_{\eta} \begin{pmatrix} \left(\{w\}^T [b_2]^T |J| \right)_{,\eta} \{\sigma\} \\ + \left(\{w\}^T [b_3]^T |J| \right)_{,\zeta} \{\sigma\} \end{pmatrix} d\eta d\zeta \right) d\xi$$

From equation (2.28) and (2.29) and since,

$\left(|J| [b_2] \right)_{,\eta} + \left(|J| [b_3] \right)_{,\zeta} = -2 |J| [b_1]$ (For proof see appendix B)

Therefore,

$$I = \int_{\xi} \xi \begin{pmatrix} \int_{\zeta} \{w\}^T \{t^{\eta}\} |\bar{g}^{\eta}| d\zeta + \int_{\eta} \{w\}^T \{t^{\zeta}\} |\bar{g}^{\zeta}| d\eta \\ - \int_{\zeta}\int_{\eta} \left(-2\{w\}^T [b_1]^T + \{w\}^T_{,\eta} [b_2]^T + \{w\}^T_{,\zeta} [b_3]^T \right) \{\sigma\} |J| d\eta d\zeta \end{pmatrix} d\xi \tag{2.35}$$

From (2.35) ,(2.33) and (2.32) substitute in (2.30).

$$\int_{\xi} \begin{pmatrix} \int_{\zeta}\int_{\eta} \{w\}^T [b_1]^T \{\sigma_{,\xi}\} \xi^2 |J| d\eta d\zeta + \xi \begin{pmatrix} \int_{\zeta} \{w\}^T \{t^{\eta}\} |\bar{g}^{\eta}| d\zeta + \int_{\eta} \{w\}^T \{t^{\zeta}\} |\bar{g}^{\zeta}| d\eta \\ - \int_{\zeta}\int_{\eta} \begin{pmatrix} -2\{w\}^T [b_1]^T \\ + \{w\}^T_{,\eta} [b_2]^T \\ + \{w\}^T_{,\zeta} [b_3]^T \end{pmatrix} \{\sigma\} |J| d\eta d\zeta \end{pmatrix} \\ + \int_{\zeta}\int_{\eta} \{w\}^T \{p\} \xi^2 |J| d\eta d\zeta + \int_{\zeta}\int_{\eta} \{w\}^T \omega^2 \rho \{u\} \xi^2 |J| d\eta d\zeta \end{pmatrix} d\xi = \{0\} \tag{2.36}$$

Since the approximation is only in the circumferential direction ((η,ζ) plane) and not the radial one (ξ direction), therefore the integrand in equation (2.36) equals zero and there is no need to make integration in ξ direction.

Therefore,

$$\xi^2 \int_\zeta \int_\eta \{w\}^T [b_1]^T \{\sigma_{,\xi}\} |J| d\eta d\zeta + \xi \left(\begin{array}{l} \int_\zeta \{w\}^T \{t^\eta\} |\bar{g}^\eta| d\zeta + \int_\eta \{w\}^T \{t^\zeta\} |\bar{g}^\zeta| d\eta \\ - \int_\zeta \int_\eta \left(\begin{array}{l} -2\{w\}^T [b_1]^T + \{w\}^T_{,\eta} [b_2]^T \\ + \{w\}^T_{,\zeta} [b_3]^T \end{array} \right) \{\sigma\} |J| d\eta d\zeta \end{array} \right) \tag{2.37}$$

$$+ \xi^2 \int_\zeta \int_\eta \{w\}^T \{p\} |J| d\eta d\zeta + \xi^2 \int_\zeta \int_\eta \{w\}^T \omega^2 \rho \{u\} |J| d\eta d\zeta = \{0\}$$

It is clear that there is no volume integrals in the equation above, they are all surface integrals.

As assumed above:

$$\{u(\xi,\eta,\zeta,\omega)\} = [N(\eta,\zeta)]\{u(\xi,\omega)\} \tag{2.38}$$

For $\{u(\xi,\omega)\}$ ω will be omitted for conciseness

From equation (2.3), (2.4)

$$\{\sigma\} = [D][L]\{u\}$$

From equation (2.25)

$$\{\sigma\} = [D]\left([b_1][N(\eta,\zeta)]\{u(\xi)\}_{,\xi} + \frac{1}{\xi}\left(\begin{array}{l} [b_2][N(\eta,\zeta)]_{,\eta}\{u(\xi)\} \\ + [b_3][N(\eta,\zeta)]_{,\zeta}\{u(\xi)\} \end{array} \right) \right) \tag{2.39}$$

Let, $[B_1(\eta,\zeta)] = [b_1][N(\eta,\zeta)]$ (2.40)

and $[B_2(\eta,\zeta)] = [b_2][N(\eta,\zeta)]_{,\eta} + [b_3][N(\eta,\zeta)]_{,\zeta}$ (2.41)

Therefore,

$$\{\sigma\} = [D]\left([B_1(\eta,\zeta)]\{u(\xi)\}_{,\xi} + \frac{1}{\xi}[B_2(\eta,\zeta)]\{u(\xi)\} \right) \tag{2.42}$$

Let the weighting function be discretized as the displacement i.e.:

$$\{w\} = \{w(\xi,\eta,\zeta,\omega)\} = [N(\eta,\zeta)]\{w(\xi,\omega)\} \tag{2.43}$$

For $\{w(\xi,\omega)\}$ ω will be omitted for conciseness, and for $[N(\eta,\zeta)]$, $[B_1(\eta,\zeta)]$ and $[B_2(\eta,\zeta)]$ (η,ζ) will also be omitted.

Substitute by equations (2.42) and (2.43) in (2.37)

$$\begin{aligned}
&\xi^2\int_\zeta\int_\eta \{w(\xi)\}^T[N]^T[b_1]^T[D]\left([B_1]\{u(\xi)\}_{,\xi\xi}+\frac{1}{\xi}[B_2]\{u(\xi)\}_{,\xi}-\frac{1}{\xi^2}[B_2]\{u(\xi)\}\right)|J|d\eta d\zeta \\
&+\xi\left(\begin{aligned}&\int_\zeta \{w(\xi)\}^T[N]^T\{t^\eta\}|\bar{g}^\eta|d\zeta+\int_\eta\{w(\xi)\}^T[N]^T\{t^\zeta\}|\bar{g}^\zeta|d\eta \\ &-\int_\zeta\int_\eta\begin{pmatrix}-2\{w(\xi)\}^T[N]^T[b_1]^T \\ +\{w(\xi)\}^T[N]^T_{,\eta}[b_2]^T \\ +\{w(\xi)\}^T[N]^T_{,\zeta}[b_3]^T\end{pmatrix}[D]\left([B_1]\{u(\xi)\}_{,\xi}+\frac{1}{\xi}[B_2]\{u(\xi)\}\right)|J|d\eta d\zeta\end{aligned}\right) \\
&+\xi^2\int_\zeta\int_\eta\{w(\xi)\}^T[N]^T\{p\}|J|d\eta d\zeta+\xi^2\int_\zeta\int_\eta\{w(\xi)\}^T[N]^T\omega^2\rho[N]\{u(\xi)\}|J|d\eta d\zeta=\{0\}
\end{aligned} \tag{2.44}$$

Let, $$\{T\}=\int_\zeta[N]^T\{t^\eta\}|\bar{g}^\eta|d\zeta+\int_\eta[N]^T\{t^\zeta\}|\bar{g}^\zeta|d\eta \tag{2.45}$$

And, $$\{P\}=\int_\zeta\int_\eta[N]^T\{p\}|J|d\eta d\zeta \tag{2.46}$$

Since, the area of plane (ξ,ζ) and plane (ξ,η) are given by:

$$dS^\eta=\left|\bar{\hat{r}}_{,\zeta}\times\bar{\hat{r}}_{,\xi}\right|d\zeta d\xi=\left|\xi\bar{r}_{,\zeta}\times\bar{r}\right|d\zeta d\xi=\xi\left|\bar{g}^\eta\right|d\zeta d\xi$$

and $dS^\zeta=\left|\bar{\hat{r}}_{,\xi}\times\bar{\hat{r}}_{,\eta}\right|d\eta d\xi=\left|\bar{r}\times\xi\bar{r}_{,\eta}\right|d\eta d\xi=\xi\left|\bar{g}^\zeta\right|d\eta d\xi$, respectively. (See equations (2.16, 2.17 and 2.18))

Therefore, $\xi\{T\}$ represents the nodal surface traction on the planes (ξ,ζ) and (ξ,η) respectively.

And $\xi^2\{P\}$ represents the nodal body force (see equation (2.32)).

Substitute by equations (2.40), (2.41), (2.45) and (2.46) in (2.44)

Therefore,

$$\{w(\xi)\}^T \begin{pmatrix} \xi^2 \int_\zeta\int_\eta [B_1]^T [D]\left([B_1]\{u(\xi)\}_{,\xi\xi} + \frac{1}{\xi}[B_2]\{u(\xi)\}_{,\xi} - \frac{1}{\xi^2}[B_2]\{u(\xi)\}\right)|J|d\eta d\zeta \\ + \xi\left(\{T\} - \int_\zeta\int_\eta\left(-2[B_1]^T + [B_2]^T\right)[D]\left([B_1]\{u(\xi)\}_{,\xi} + \frac{1}{\xi}[B_2]\{u(\xi)\}\right)|J|d\eta d\zeta\right) \\ + \xi^2\{P\} + \xi^2\int_\zeta\int_\eta [N]^T \rho [N] |J| d\eta d\zeta \omega^2 \{u(\xi)\} \end{pmatrix} = \{0\}$$

$$\xi^2 \int_\zeta\int_\eta [B_1]^T [D]\left([B_1]\{u(\xi)\}_{,\xi\xi} + \frac{1}{\xi}[B_2]\{u(\xi)\}_{,\xi} - \frac{1}{\xi^2}[B_2]\{u(\xi)\}\right)|J|d\eta d\zeta + \xi\{T\}$$

$$+ \xi^2\{P\} - \xi\int_\zeta\int_\eta \begin{pmatrix} -2[B_1]^T[D][B_1]\{u(\xi)\}_{,\xi} - \frac{2}{\xi}[B_1]^T[D][B_2]\{u(\xi)\} \\ + [B_2]^T[D][B_1]\{u(\xi)\}_{,\xi} + \frac{1}{\xi}[B_2]^T[D][B_2]\{u(\xi)\} \end{pmatrix}|J|d\eta d\zeta \tag{2.47}$$

$$+ \xi^2\int_\zeta\int_\eta [N]^T \rho [N] |J| d\eta d\zeta \omega^2 \{u(\xi)\} = \{0\}$$

Let, $[E_o(\eta,\zeta)] = \int_\zeta\int_\eta [B_1]^T[D][B_1]|J|d\eta d\zeta$ (2.48)

$$[E_1(\eta,\zeta)] = \int_\zeta\int_\eta [B_2]^T[D][B_1]|J|d\eta d\zeta \tag{2.49}$$

$$[E_2(\eta,\zeta)] = \int_\zeta\int_\eta [B_2]^T[D][B_2]|J|d\eta d\zeta \tag{2.50}$$

and, $[M_o(\eta,\zeta)] = \int_\zeta\int_\eta [N]^T \rho [N]|J|d\eta d\zeta$ (2.51)

(η,ζ) will be omitted from $[E_o(\eta,\zeta)]$, $[E_1(\eta,\zeta)]$, $[E_2(\eta,\zeta)]$ and $[M_o(\eta,\zeta)]$ for conciseness

Substitute by equations (2.48), (2.49), (2.50), (2.51) in (2.47)

Therefore,

$$\xi^2[E_o]\{u(\xi)\}_{,\xi\xi} + \xi\{T\} + \xi^2\{P\} + \xi[E_1]^T\{u(\xi)\}_{,\xi} - [E_1]^T\{u(\xi)\} + 2\xi[E_o]\{u(\xi)\}_{,\xi} + 2[E_1]^T\{u(\xi)\}$$
$$- \xi[E_1]\{u(\xi)\}_{,\xi} - [E_2]\{u(\xi)\} + \xi^2[M_o]\omega^2\{u(\xi)\} = \{0\}$$

$$\xi^2[E_o]\{u(\xi)\}_{,\xi\xi} + \left(2[E_0]-[E_1]+[E_1]^T\right)\xi\{u(\xi)\}_{,\xi} + \left([E_1]^T - [E_2]\right)\{u(\xi)\}$$
$$+\xi^2[M_o]\omega^2\{u(\xi)\}+\xi\{T\}+\xi^2\{P\}=\{0\} \tag{2.52}$$

Equation (2.52) is the *scaled boundary finite element equation for displacement* for 3D, it is a second order linear ordinary differential equation with variable coefficients, and it can be written in a general case for both 2 and 3 dimensions as follows

$$\xi^2[E_o]\{u(\xi)\}_{,\xi\xi} + \left((s-1)[E_0]-[E_1]+[E_1]^T\right)\xi\{u(\xi)\}_{,\xi} + \left((s-2)[E_1]^T - [E_2]\right)\{u(\xi)\}$$
$$+\xi^2[M_o]\omega^2\{u(\xi)\}+\xi\{T\}+\xi^2\{P\}=\{0\} \tag{2.53}$$

where, s is the spatial dimension =2 or 3 for 2D and 3D problems, respectively.

2.5. The dynamic stiffness

It is defined as the force which produces unit amplitude of displacement. First we will derive a relation between the medium properties and the dynamic stiffness using the dimensional analysis [12], and then an equation to get the dynamic stiffness for a system is derived.

The dynamic stiffness S^∞ (∞ stands for infinity) depends on the following:

r: The characteristic length

G: Shear modulus

ω: Circular frequency

ρ: Medium density

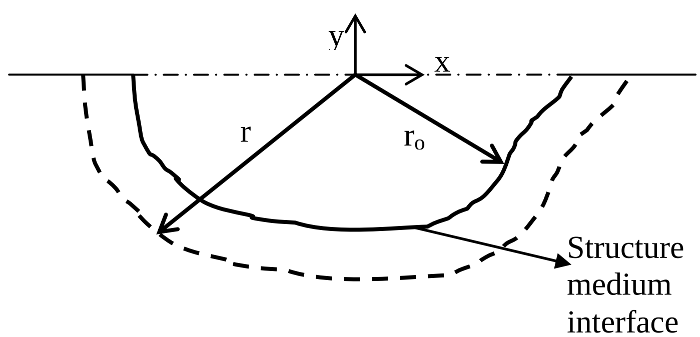


Fig.(2.3): The structure medium interface, the characteristic length and the similar boundaries (in dotted lines)

Using the dimensional analysis:

Physical quantity	Dimensions
S^{∞}	MT^{-2}
R	L
G	$ML^{-1}T^{-2}$
ω	T^{-1}
ρ	ML^{-3}

So if we multiply all the above terms (each raised to a power) the product will be dimensionless.

$$\left(S^{\infty}\right)^{n_1} r^{n_2} G^{n_3} \omega^{n_4} \rho^{n_5} \xrightarrow{\text{dimensions}} M^{n_1} T^{-2n_1} L^{n_2} M^{n_3} L^{-n_3} T^{-2n_3} T^{-n_4} M^{n_5} L^{-3n_5}$$
$$\left(S^{\infty}\right)^{n_1} r^{n_2} G^{n_3} \omega^{n_4} \rho^{n_5} \xrightarrow{\text{dimensions}} M^{n_1+n_3+n_5} L^{n_2-n_3-3n_5} T^{-2n_1-2n_3-n_4} \quad (2.54)$$

Therefore,

$$n_1 + n_3 + n_5 = 0 \quad (2.55)$$

$$n_2 - n_3 - 3n_5 = 0 \quad (2.56)$$

$$-2n_1 - 2n_3 - n_4 = 0 \quad (2.57)$$

In order to solve the three above equations (3 equations in 5 unknowns) an assumption has to be made.

Let, n_1=1 and n_4=0

Therefore, from equation (2.57) n_3=-1

From (2.55) n_5=0

From (2.56) n_2=-1

Substitute by the answers in (2.54)

$\left(S^{\infty}\right)^{1} r^{-1} G^{-1} \omega^{0} \rho^{0} = \bar{S}^{\infty}$ where, $\bar{S}^{\infty}$ is a dimensionless quantity

$$S^{\infty} = \bar{S}^{\infty} r G \quad (2.58)$$

Let, n_1=0 and n_4=1

Therefore, from equation (2.57) n_3=-1/2

From (2.55) n_5=1/2

From (2.56) n_2=1

Substitute by the answers in (2.54)

$\left(S^{\infty}\right)^{0} r^{1} G^{-1/2} \omega^{1} \rho^{1/2} = a$ where, a is a dimensionless quantity

$$a = \omega r \sqrt{\frac{\rho}{G}}$$

Since, $c_s = \sqrt{\frac{G}{\rho}}$ is the shear wave velocity.

Therefore, $a = \frac{\omega r}{c_s}$

a is known as the dimensionless frequency

Since, $r = \xi r_o$

Therefore, $a = \xi \frac{\omega r_o}{c_s} = \xi a_o$ (2.59)

From equations (2.58) and (2.59), the dynamic stiffness is said to be function of the dimensionless frequency a

Therefore, $S^{\infty}(a) = \bar{S}^{\infty}(a) r G = \xi \bar{S}^{\infty}(a) r_o G$ (2.60)

Also it can be written as

$$S^{\infty}(\omega,\xi) = \bar{S}^{\infty}(\omega,\xi) r G = \xi \bar{S}^{\infty}(\omega,\xi) r_o G \quad (2.61)$$

The above 2 equations (2.60) and (2.61) can be written in a more general form for 2 and 3 dimensional problems:

$S^{\infty}(\omega,\xi) = \xi^{s-2} \bar{S}^{\infty}(a) r_o G = \xi^{s-2} \bar{S}^{\infty}(\omega,\xi) r_o G$, where s=2 for 2D and 3 for 3D problems

Now we are going to derive an equation to get the dynamic stiffness

Let, $\{Q(\xi)\}$ be the vector of the nodal internal forces amplitude on surface with constant ξ

Since, $\{t\}^{\xi}$ is the traction on surface (η,ζ), so a relation can be made between $\{Q(\xi)\}$ and $\{t\}^{\xi}$

$$\{w(\xi)\}^T\{Q(\xi)\} = \int_{S^\xi}\{w\}^T\{t\}^\xi\, dS^\xi \tag{2.62}$$

S^ξ is the area of the plane (η,ζ) corresponding to a certain node. The weighting function is used to overcome the error in approximation

Since, $\{w\} = [N(\eta,\zeta)]\{w(\xi)\}$ as said before

Substitute in equation (2.62)

$$\{w(\xi)\}^T\{Q(\xi)\} = \int_{S^\xi}\{w(\xi)\}^T[N(\eta,\zeta)]^T\{t\}^\xi\, dS^\xi \tag{2.63}$$

Substitute by equation (2.27) in (2.63)

$$\{w(\xi)\}^T\{Q(\xi)\} = \{w(\xi)\}^T\int_{S^\xi}[N(\eta,\zeta)]^T[b_1]^T\frac{|J|}{|\bar{g}^\xi|}\{\sigma\}dS^\xi$$

$$\{Q(\xi)\} = \int_{S^\xi}[N(\eta,\zeta)]^T[b_1]^T\frac{|J|}{|\bar{g}^\xi|}\{\sigma\}dS^\xi \tag{2.64}$$

The infinitesimal area dS^ξ can be expressed as follows:

$$dS^\xi = \left|\hat{\bar{r}}_{,\eta}\times\hat{\bar{r}}_{,\zeta}\right|d\eta d\zeta$$

Using equations (2.14), (2.15) and (2.16)

$$dS^\xi = \left|\xi\bar{r}_{,\eta}\times\xi\bar{r}_{,\zeta}\right|d\eta d\zeta = \xi^2\left|\bar{g}^\xi\right|d\eta d\zeta$$

Substitute in equation (2.64)

$$\{Q(\xi)\} = \int_\zeta\int_\eta[N(\eta,\zeta)]^T[b_1]^T|J|\{\sigma\}\xi^2 d\eta d\zeta$$

Substitute in the above equation using (2.40) and (2.42)

$$\{Q(\xi)\} = \xi^2\int_\zeta\int_\eta[B_1]^T|J|[D]\left([B_1]\{u(\xi)\}_{,\xi} + \frac{1}{\xi}[B_2]\{u(\xi)\}\right)d\eta d\zeta$$

Substitute in the above equation using (2.48) and (2.49)

$$\{Q(\xi)\} = \xi^2[E_o]\{u(\xi)\}_{,\xi} + \xi[E_1]^T\{u(\xi)\} \tag{2.65}$$

Let $\{R(\xi)\}$ be the vector of the nodal external forces amplitude, so a relation can be written with respect to the dynamic stiffness as follows

$$\{R(\xi)\} = -\{Q(\xi)\} = \left[S^\infty(\omega,\xi)\right]\{u(\xi)\} - \{R_F(\xi)\} \tag{2.66}$$

The –ve sign in the above equation is for the difference in direction between the internal $\{Q(\xi)\}$ and external forces $\{R(\xi)\}$. $\{R_F(\xi)\}$ is the vector of external body forces and surface traction

Equating both (2.65) and (2.66)

$$\left[S^{\infty}(\omega,\xi)\right]\{u(\xi)\}-\{R_F(\xi)\}=-\xi^2[E_o]\{u(\xi)\}_{,\xi}-\xi[E_1]^T\{u(\xi)\} \quad (2.67)$$

Differentiating the above equation with respect to ξ

$$\left[S^{\infty}(\omega,\xi)\right]_{,\xi}\{u(\xi)\}+\left[S^{\infty}(\omega,\xi)\right]\{u(\xi)\}_{,\xi}=\begin{pmatrix}-2\xi[E_o]\{u(\xi)\}_{,\xi}-\xi^2[E_o]\{u(\xi)\}_{,\xi\xi}-[E_1]^T\{u(\xi)\}\\ -\xi[E_1]^T\{u(\xi)\}_{,\xi}+\{R_F(\xi)\}_{,\xi}\end{pmatrix}$$

$$\begin{aligned}&\left[S^{\infty}(\omega,\xi)\right]_{,\xi}\{u(\xi)\}+\left[S^{\infty}(\omega,\xi)\right]\{u(\xi)\}_{,\xi}+\xi\left(2[E_o]+[E_1]^T\right)\{u(\xi)\}_{,\xi}+\xi^2[E_o]\{u(\xi)\}_{,\xi\xi}\\&+[E_1]^T\{u(\xi)\}-\{R_F(\xi)\}_{,\xi}=\{0\}\end{aligned} \quad (2.68)$$

Subtract equation (2.52) from (2.68)

$$\begin{aligned}&\left[S^{\infty}(\omega,\xi)\right]_{,\xi}\{u(\xi)\}+\left[S^{\infty}(\omega,\xi)\right]\{u(\xi)\}_{,\xi}+\xi[E_1]\{u(\xi)\}_{,\xi}+[E_2]\{u(\xi)\}-\omega^2\xi^2[M_o]\{u(\xi)\}\\&-\xi\{T\}-\xi^2\{P\}-\{R_F(\xi)\}_{,\xi}=\{0\}\end{aligned} \quad (2.69)$$

From equation (2.67)

$$\{u(\xi)\}_{,\xi}=\xi^{-2}[E_o]^{-1}\left(-\left[S^{\infty}(\omega,\xi)\right]\{u(\xi)\}-\xi[E_1]^T\{u(\xi)\}+\{R_F(\xi)\}\right) \quad (2.70)$$

Substitute by equation (2.70) in (2.69)

$$\begin{aligned}&\left[S^{\infty}(\omega,\xi)\right]_{,\xi}\{u(\xi)\}+\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]\right)\xi^{-2}[E_o]^{-1}\left(-\left[S^{\infty}(\omega,\xi)\right]\{u(\xi)\}-\xi[E_1]^T\{u(\xi)\}+\{R_F(\xi)\}\right)\\&+[E_2]\{u(\xi)\}-\omega^2\xi^2[M_o]\{u(\xi)\}-\xi\{T\}-\xi^2\{P\}-\{R_F(\xi)\}_{,\xi}=\{0\}\end{aligned}$$

Therefore,

$$\begin{aligned}&\left(\left[S^{\infty}(\omega,\xi)\right]_{,\xi}-\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]\right)\xi^{-2}[E_o]^{-1}\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]^T\right)+[E_2]-\omega^2\xi^2[M_o]\right)\{u(\xi)\}\\&+\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]\right)\xi^{-2}[E_o]^{-1}\{R_F(\xi)\}-\xi\{T\}-\xi^2\{P\}-\{R_F(\xi)\}_{,\xi}=\{0\}\end{aligned}$$

The above equation can be split into 2 equations

$$\begin{aligned}&\left[S^{\infty}(\omega,\xi)\right]_{,\xi}-\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]\right)\xi^{-2}[E_o]^{-1}\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]^T\right)+[E_2]\\&-\omega^2\xi^2[M_o]=[0]\end{aligned} \quad (2.71)$$

$$\{R_F(\xi)\}_{,\xi}-\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]\right)\xi^{-2}[E_o]^{-1}\{R_F(\xi)\}+\xi\{T\}+\xi^2\{P\}=\{0\} \quad (2.72)$$

So, from equation (2.71) the dynamic stiffness can be calculated and from (2.72) the reaction of the body force and the surface traction can be calculated

Now equation (2.71) will be simplified:

$$\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]\right)\xi^{-1}[E_o]^{-1}\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]^T\right)-\xi[E_2]-\xi\left[S^{\infty}(\omega,\xi)\right]_{,\xi} + \omega^2\xi^3[M_o]=[0] \tag{2.73}$$

From equation (2.61)

$$\xi\left[S^{\infty}(\omega,\xi)\right]_{,\xi}=\xi\left(\left[\bar{S}^{\infty}(\omega,\xi)\right]r_oG+\xi\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,\xi}r_oG\right)$$

From equation (2.61)

$$\xi\left[S^{\infty}(\omega,\xi)\right]_{,\xi}=\left[S^{\infty}(\omega,\xi)\right]+\xi^2\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,\xi}r_oG \tag{2.74}$$

Since, $\xi\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,\xi}=\xi\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,a}\dfrac{\partial a}{\partial \xi}$ and $a=\xi\dfrac{\omega r_o}{c_s}$

Therefore, $\xi\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,\xi}=\xi\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,a}\dfrac{\omega r_o}{c_s}=a\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,a}$ (2.75)

Therefore, $a\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,a}=\omega\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,\omega}$

From equation (2.75): $\xi\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,\xi}=\omega\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,\omega}$ (2.76)

Substitute equation (2.76) in (2.74)

$$\xi\left[S^{\infty}(\omega,\xi)\right]_{,\xi}=\left[S^{\infty}(\omega,\xi)\right]+\xi\omega\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,\omega}r_oG \tag{2.77}$$

From equation (2.61)

$$\left[S^{\infty}(\omega,\xi)\right]_{,\omega}=\xi\left[\bar{S}^{\infty}(\omega,\xi)\right]_{,\omega}r_oG \tag{2.78}$$

Substitute by (2.78) in (2.77)

$$\xi\left[S^{\infty}(\omega,\xi)\right]_{,\xi}=\left[S^{\infty}(\omega,\xi)\right]+\omega\left[S^{\infty}(\omega,\xi)\right]_{,\omega} \tag{2.79}$$

Now substitute by (2.79) in (2.73)

$$\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]\right)\left(\xi[E_o]\right)^{-1}\left(\left[S^{\infty}(\omega,\xi)\right]+\xi[E_1]^T\right)-\xi[E_2]-\left[S^{\infty}(\omega,\xi)\right] -\omega\left[S^{\infty}(\omega,\xi)\right]_{,\omega}+\omega^2\xi^3[M_o]=[0] \tag{2.80}$$

Which is the *scaled boundary finite element equation for dynamic stiffness* for 3D problems, it is a nonlinear first order ordinary differential equation, (The equation is in one independent variable at constant ξ).

The above equation can be written in more general form as:

$$\left(\left[S^{\infty}(\omega,\xi)\right]+\xi^{s-2}[E_1]\right)\left(\xi^{s-2}[E_o]\right)^{-1}\left(\left[S^{\infty}(\omega,\xi)\right]+\xi^{s-2}[E_1]^T\right)-\xi^{s-2}[E_2] -(s-2)\left[S^{\infty}(\omega,\xi)\right]-\omega\left[S^{\infty}(\omega,\xi)\right]_{,\omega}+\omega^2\xi^s[M_o]=[0] \tag{2.81}$$

Where, s=3 for 3D problems and 2 for 2D problems.

2.6. The SBFE equation in time domain

In this part I am going to write equation (2.81) in time domain rather than in frequency domain.

The relationship between the acceleration dynamic stiffness $\left[M^{\infty}(\omega,\xi)\right]$ (which is the force which produces unit amplitude acceleration) and the displacement dynamic stiffness $\left[S^{\infty}(\omega,\xi)\right]$ (see [18] and [19]) is:

$$\left[M^{\infty}(\omega,\xi)\right]=\frac{\left[S^{\infty}(\omega,\xi)\right]}{(i\omega)^{2}} \tag{2.82}$$

Substitute by equation (2.82) in (2.81)

$$\left((i\omega)^{2}\left[M^{\infty}(\omega,\xi)\right]+\xi^{s-2}[E_{1}]\right)\left(\xi^{s-2}[E_{o}]\right)^{-1}\left((i\omega)^{2}\left[M^{\infty}(\omega,\xi)\right]+\xi^{s-2}[E_{1}]^{T}\right)-\xi^{s-2}[E_{2}]-(s-2)(i\omega)^{2}\left[M^{\infty}(\omega,\xi)\right]$$
$$-\omega\left((i\omega)^{2}\left[M^{\infty}(\omega,\xi)\right]\right)_{,\omega}+\omega^{2}\xi^{s}[M_{o}]=[0]$$

After simplification and division by $(i\omega)^{4}$:

$$\xi^{2-s}\left[M^{\infty}(\omega,\xi)\right][E_{o}]^{-1}\left[M^{\infty}(\omega,\xi)\right]+\frac{1}{(i\omega)^{2}}\left[M^{\infty}(\omega,\xi)\right][E_{o}]^{-1}[E_{1}]^{T}+\frac{1}{(i\omega)^{2}}[E_{1}][E_{o}]^{-1}\left[M^{\infty}(\omega,\xi)\right]$$
$$+\frac{\xi^{s-2}}{(i\omega)^{4}}[E_{1}][E_{o}]^{-1}[E_{1}]^{T}-\frac{\xi^{s-2}}{(i\omega)^{4}}[E_{2}]-\frac{1}{(i\omega)^{2}}(s-2)\left[M^{\infty}(\omega,\xi)\right]+\frac{1}{\omega}\left[M^{\infty}(\omega,\xi)\right]_{,\omega}$$
$$-\frac{2}{(i\omega)^{2}}\left[M^{\infty}(\omega,\xi)\right]-\frac{\xi^{s}}{(i\omega)^{2}}[M_{0}]=[0]$$

Applying inverse Fourier transformation to the above equation (see [18] and [19])

$$\xi^{2-s}\int_{0}^{t}\left[M^{\infty}((t-\tau),\xi)\right][E_{o}]^{-1}\left[M^{\infty}(\tau,\xi)\right]d\tau+\int_{0}^{t}\int_{0}^{\tau}\left[M^{\infty}(\tau',\xi)\right]d\tau'd\tau[E_{o}]^{-1}[E_{1}]^{T}$$
$$+[E_{1}][E_{o}]^{-1}\int_{0}^{t}\int_{0}^{\tau}\left[M^{\infty}(\tau',\xi)\right]d\tau'd\tau+\xi^{s-2}\frac{t^{3}}{6}[E_{1}][E_{o}]^{-1}[E_{1}]^{T}-\xi^{s-2}\frac{t^{3}}{6}[E_{2}] \tag{2.83}$$
$$-(s+1)\int_{0}^{t}\int_{0}^{\tau}\left[M^{\infty}(\tau',\xi)\right]d\tau'd\tau+t\int_{0}^{t}\left[M^{\infty}(\tau,\xi)\right]d\tau-\xi^{s}t[M_{0}]=[0]$$

where, $\left[M^{\infty}(\tau,\xi)\right]$ is the unit impulse response function for acceleration (which is the inverse Fourier transform of $\left[M^{\infty}(\omega,\xi)\right]$), and it can be defined as the force which produces 1 unit velocity.

The above equation can be simplified by decomposing $[E_o]$ using Cholesky decomposition

$$[E_o]=[U]^T[U] \tag{2.84}$$

Where, $[U]$ is an upper triangular matrix.

Substitute by equation (2.84) in (2.83), then pre and post multiply equation (2.83) by $([U]^T)^{-1}$ and $[U]^{-1}$, then rearrange.

Therefore,

$$\begin{aligned}&\xi^{2-s}\int_0^t[m^\infty((t-\tau),\xi)][m^\infty(\tau,\xi)]d\tau+\int_0^t\int_0^\tau[m^\infty(\tau',\xi)]d\tau' d\tau\left([e_1]^T-\left(\frac{s+1}{2}\right)[I]\right)\\&+\left([e_1]-\left(\frac{s+1}{2}\right)[I]\right)\int_0^t\int_0^\tau[m^\infty(\tau',\xi)]d\tau' d\tau+\xi^{s-2}\frac{t^3}{6}\left([e_1][e_1]^T-[e_2]\right)+t\int_0^t[m^\infty(\tau,\xi)]d\tau\\&-\xi^s t[m_0]=[0]\end{aligned} \tag{2.85}$$

where,

$$[m^\infty(t,\xi)]=([U]^T)^{-1}[M^\infty(t,\xi)][U]^{-1} \tag{2.86}$$

$[e_1]=([U]^T)^{-1}[E_1][U]^{-1}$

$[e_2]=([U]^T)^{-1}[E_2][U]^{-1}$

$[m_o]=([U]^T)^{-1}[M_o][U]^{-1}$, and $[I]$ is the identity matrix.

For ξ=1, equation (2.85) can be written as follows

$$\begin{aligned}&\int_0^t[m^\infty(t-\tau)][m^\infty(\tau)]d\tau+\int_0^t\int_0^\tau[m^\infty(\tau')]d\tau' d\tau\left([e_1]^T-\left(\frac{s+1}{2}\right)[I]\right)\\&+\left([e_1]-\left(\frac{s+1}{2}\right)[I]\right)\int_0^t\int_0^\tau[m^\infty(\tau')]d\tau' d\tau+\frac{t^3}{6}\left([e_1][e_1]^T-[e_2]\right)+t\int_0^t[m^\infty(\tau)]d\tau\\&-t[m_0]=[0]\end{aligned} \tag{2.87}$$

2.7. Time discretization

Equation (2.87) will be solved numerically, where the unit impulse response function will be discretized into linear function [22], and in order to decrease the time of analysis the time step used in calculating the unit impulse will be greater than that used in calculating the response of the medium (displacement, velocity and acceleration).

The time step in calculating the unit impulse is △t`=N△t, where, △t is the time step used in calculating the response of the medium, and N is the number of steps in time △t` (see fig.(2.4)).

Therefore,

$\left[M_o^{\infty}\right]$ will be at t=0

$\left[M_1^{\infty}\right]$ will be at t=△t

$\left[M_m^{\infty}\right]$ will be at t=m△t

where, m is a counter for the time steps

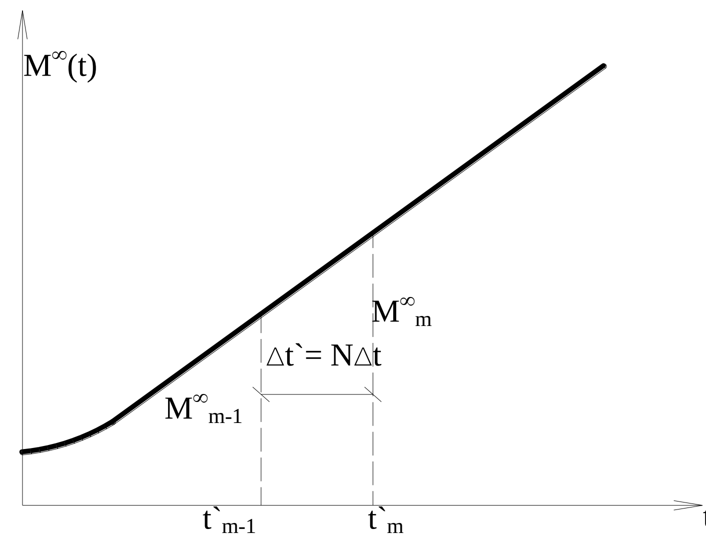
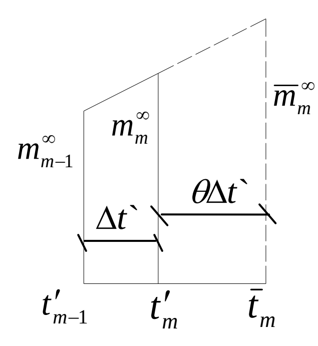


Fig.(2.4): Unit impulse response function

The $M^{\infty}(t)$ drawn in fig.(2.2) equals $\{\phi\}^T\left[M^{\infty}(t)\right]\{\phi\}$ where, $\{\phi\}$ is the vector of spatial motion pattern (The unit vector defining the direction at which $M^{\infty}(t)$ is needed to be calculated).

To improve stability for the numerical solution extrapolation will be used, i.e. $\overline{m}^{\infty}{}_{m}$ will be calculated and then $m^{\infty}{}_{m}$. Where, $\bar{t}_m = t^{`}_{m-1} + \theta\Delta t^{`}$ and



$$\left[m_m^{\infty}\right] = (\theta - 1)\theta^{-1}\left[m_{m-1}^{\infty}\right] + \theta^{-1}\left[\overline{m}_m^{\infty}\right] \qquad (2.88)$$

<u>First get</u> $\left[M_o^{\infty}\right]$:

From [4]

Since $\left[M^{\infty}(t)\right]=C_{\infty}H(t)+K_{\infty}tH(t)+\left[M_r^{\infty}(t)\right]$ (2.89)

Where, $H(t)$ is the Heaviside function, C_{∞} is the radiation damping, K_{∞} is the static stiffness at $t \to 0$, see fig.(2.5)

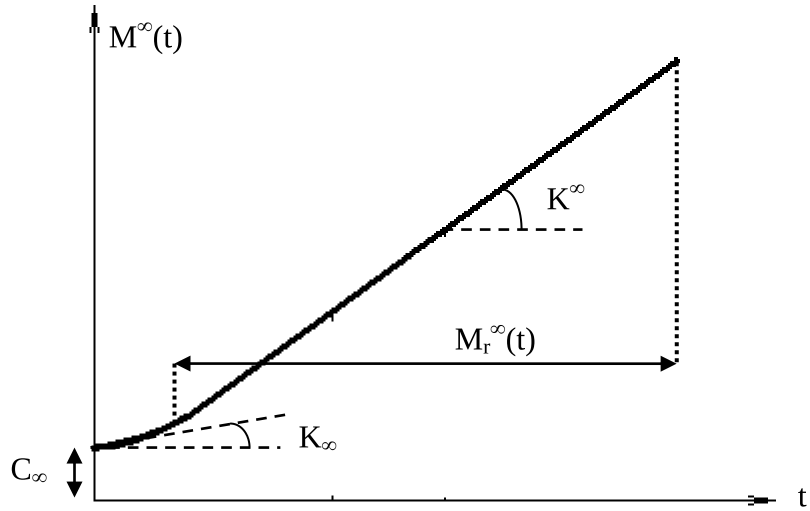


Fig.(2.5): Unit impulse response function showing C_{∞}, K_{∞}, K^{∞} and $M_r^{\infty}(t)$

K_{∞} is the static stiffness at $t \to \infty$

At t=0

$$\left[M^{\infty}(0)\right]=\left[M_o^{\infty}\right]=C_{\infty} \quad (2.90)$$

As said in [4], [9] and [14] C_{∞} can be calculated as follows:

Suppose an unbounded medium where an impact load $r(t)$ per area is just applied to it (normal or tangential to the surface), so the surface of the medium begins to move with the wave velocity c, after a small time t the distance described will be ct, so knowing that the impact is the change in momentum we can get the velocity of the medium after time t, and since the force is just applied to the medium so the calculated force will be equal to the damping force of the medium.

Therefore, $r(t)t = m\dot{u}(t)-0$ where, m is the mass of the medium that moved in time t and $\dot{u}(t)$ is the velocity needed (2.91)

Since, $m = \rho ctS$ where, ρ is the medium density and S is the out of plane area

Therefore, $r(t) = \rho c\dot{u}$ which is as said above equal to the damping force of the medium

Since the damping force $f_D(t) = \overline{C}_\infty \dot{u}(t)$

Therefore, $\overline{C}_\infty = \rho c$

Generally speaking:

$$\{f_D(x,y,z,t)\} = [\overline{C}_\infty]\{\dot{u}(x,y,z,t)\} = \rho[c]\{\dot{u}(x,y,z,t)\} = \rho\begin{bmatrix} c_p & 0 & 0 \\ 0 & c_s & 0 \\ 0 & 0 & c_s \end{bmatrix}\begin{Bmatrix} \dot{u}_n(x,y,z,t) \\ \dot{u}_{t_1}(x,y,z,t) \\ \dot{u}_{t_2}(x,y,z,t) \end{Bmatrix} \quad (2.92)$$

where subscripts n, t_1 and t_2 stands for normal and tangential directions, respectively. c_p and c_s are the longitudinal and shear wave velocities, respectively.

But if the forces applied are not normal and tangential to the surface, the damping force must be resolved.

Since, the nodal force can be written as:

$$\{F_D(t)\} = \int_{S^\xi} [N]^T \begin{Bmatrix} f_{Dx}(x,y,z,t) \\ f_{Dy}(x,y,z,t) \\ f_{Dz}(x,y,z,t) \end{Bmatrix} dS^\xi \quad (2.93)$$

where, $[N]$ is the shape function as described before and $\begin{Bmatrix} f_{Dx}(x,y,z,t) \\ f_{Dy}(x,y,z,t) \\ f_{Dz}(x,y,z,t) \end{Bmatrix}$ is the damping force per unit area in global directions (x,y and z).

$\begin{Bmatrix} f_{Dx}(x,y,z,t) \\ f_{Dy}(x,y,z,t) \\ f_{Dz}(x,y,z,t) \end{Bmatrix} = \begin{bmatrix} n_x^n & n_x^{t_1} & n_x^{t_2} \\ n_y^n & n_y^{t_1} & n_y^{t_2} \\ n_z^n & n_z^{t_1} & n_z^{t_2} \end{bmatrix}\begin{Bmatrix} f_{Dn}(x,y,z,t) \\ f_{Dt_1}(x,y,z,t) \\ f_{Dt_2}(x,y,z,t) \end{Bmatrix}$ where, the 2 dimensional matrix here is the matrix of unit vectors perpendicular to the surface

Let, $[n] = \begin{bmatrix} n_x^n & n_x^{t_1} & n_x^{t_2} \\ n_y^n & n_y^{t_1} & n_y^{t_2} \\ n_z^n & n_z^{t_1} & n_z^{t_2} \end{bmatrix}$

From equation (2.92)

$$\begin{Bmatrix} f_{Dx}(x,y,z,t) \\ f_{Dy}(x,y,z,t) \\ f_{Dz}(x,y,z,t) \end{Bmatrix} = [n][\overline{C}_\infty]\begin{Bmatrix} \dot{u}_n(x,y,z,t) \\ \dot{u}_{t_1}(x,y,z,t) \\ \dot{u}_{t_2}(x,y,z,t) \end{Bmatrix} \tag{2.94}$$

Since, $\begin{Bmatrix} \dot{u}_x(x,y,z,t) \\ \dot{u}_y(x,y,z,t) \\ \dot{u}_z(x,y,z,t) \end{Bmatrix} = [n]\begin{Bmatrix} \dot{u}_n(x,y,z,t) \\ \dot{u}_{t_1}(x,y,z,t) \\ \dot{u}_{t_2}(x,y,z,t) \end{Bmatrix}$

Substituting in equation (2.94)

$$\begin{Bmatrix} f_{Dx}(x,y,z,t) \\ f_{Dy}(x,y,z,t) \\ f_{Dz}(x,y,z,t) \end{Bmatrix} = [n][\overline{C}_\infty][n]^{-1}\begin{Bmatrix} \dot{u}_x(x,y,z,t) \\ \dot{u}_y(x,y,z,t) \\ \dot{u}_z(x,y,z,t) \end{Bmatrix}$$

Since, $\{\dot{u}(x,y,z,t)\} = [N(\xi,\eta,\zeta)]\{\dot{u}(t)\}$

Therefore $\begin{Bmatrix} f_{Dx}(x,y,z,t) \\ f_{Dy}(x,y,z,t) \\ f_{Dz}(x,y,z,t) \end{Bmatrix} = [n][\overline{C}_\infty][n]^{-1}[N]\{\dot{u}(t)\}$ (2.95)

Substitute in equation (2.93)

$$\{F_D(t)\} = \int_{S^\xi} [N]^T [n][\overline{C}_\infty][n]^{-1}[N]\{\dot{u}(t)\} dS^\xi$$

From (2.92), $\{F_D(t)\} = \int_{S^\xi} \rho [N]^T [n][c][n]^{-1}[N] dS^\xi \{\dot{u}(t)\}$

Since, $\{F_D(t)\} = [C_\infty]\{\dot{u}(t)\}$

Therefore, $[C_\infty] = [M_o^\infty] = \int_{S^\xi} \rho [N]^T [n][c][n]^{-1}[N] dS^\xi$ (2.96)

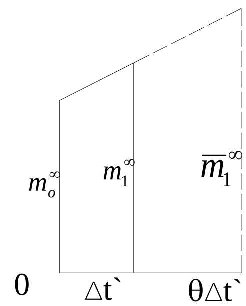
<u>For the first step m=1:</u> $\left[m^{\infty}(\tau)\right]$ at $0 \le t \le \theta\Delta t`$

The assumed equation of the unit impulse will be:

$$\frac{\left[m^{\infty}(t)\right]-\left[m_o^{\infty}\right]}{t}=\frac{\left[\overline{m}_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\theta\Delta t`}$$

Therefore,

$$\left[m^{\infty}(t)\right]=\left[m_o^{\infty}\right]+t\left(\frac{\left[\overline{m}_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\theta\Delta t`}\right) \qquad (2.97)$$



<u>From equation (2.87):</u>

$$I_1=\int_0^{\theta\Delta t`}\left[m^{\infty}(\tau)\right]d\tau=\left(\left[m_o^{\infty}\right]\tau+\frac{\tau^2}{2}\left(\frac{\left[\overline{m}_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\theta\Delta t`}\right)\right)_0^{\theta\Delta t`}$$

$$I_1=\frac{\theta\Delta t`}{2}\left(\left[\overline{m}_1^{\infty}\right]+\left[m_o^{\infty}\right]\right) \qquad (2.98)$$

$$I_2=\int_0^{\theta\Delta t`}\int_0^{\tau}\left[m^{\infty}(\tau`)\right]d\tau`d\tau=\int_0^{\theta\Delta t`}\left(\left[m_o^{\infty}\right]\tau`+\frac{\tau`^2}{2}\left(\frac{\left[\overline{m}_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\theta\Delta t`}\right)\right)_0^{\tau}d\tau$$

$$I_2=\left(\left[m_o^{\infty}\right]\frac{\tau^2}{2}+\frac{\tau^3}{6}\left(\frac{\left[\overline{m}_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\theta\Delta t`}\right)\right)_0^{\theta\Delta t`}$$

$$I_2=\frac{(\theta\Delta t`)^2}{6}\left(2\left[m_o^{\infty}\right]+\left[\overline{m}_1^{\infty}\right]\right) \qquad (2.99)$$

$$I_3=\int_0^{\theta\Delta t`}\left[m^{\infty}(t-\tau)\right]\left[m^{\infty}(\tau)\right]d\tau=\int_0^{\theta\Delta t`}\left(\left[m_o^{\infty}\right]+(t-\tau)\left(\frac{\left[\overline{m}_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\theta\Delta t`}\right)\right)\left(\left[m_o^{\infty}\right]+\tau\left(\frac{\left[\overline{m}_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\theta\Delta t`}\right)\right)d\tau$$

$$I_3=\left(\begin{array}{l}\left[m_o^{\infty}\right]^2\tau+t\tau\left(\frac{\left[\overline{m}_1^{\infty}\right]\left[m_o^{\infty}\right]-\left[m_o^{\infty}\right]^2}{\theta\Delta t`}\right)+\frac{\tau^2}{2}\left(\frac{\left[m_o^{\infty}\right]\left[\overline{m}_1^{\infty}\right]-\left[\overline{m}_1^{\infty}\right]\left[m_o^{\infty}\right]}{\theta\Delta t`}\right)+\\ \left(\frac{t\tau^2}{2}-\frac{\tau^3}{3}\right)\left(\frac{\left[\overline{m}_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\theta\Delta t`}\right)^2\end{array}\right)_0^{\theta\Delta t`}$$

$$I_3=\theta\Delta t`\left(\frac{1}{6}\left[\overline{m}_1^{\infty}\right]^2+\frac{1}{6}\left[m_o^{\infty}\right]^2+\frac{1}{3}\left[\overline{m}_1^{\infty}\right]\left[m_o^{\infty}\right]+\frac{1}{3}\left[m_o^{\infty}\right]\left[\overline{m}_1^{\infty}\right]\right) \qquad (2.100)$$

Substitute by (2.98), (2.99) and (2.100) in (2.87)

$$\theta\Delta t'\left(\frac{1}{6}\left[\overline{m}_1^\infty\right]^2+\frac{1}{6}\left[m_o^\infty\right]^2+\frac{1}{3}\left[\overline{m}_1^\infty\right]\left[m_o^\infty\right]+\frac{1}{3}\left[m_o^\infty\right]\left[\overline{m}_1^\infty\right]\right)+\frac{(\theta\Delta t')^2}{6}\left(\left[\overline{m}_1^\infty\right]+2\left[m_o^\infty\right]\right)\left([e_1]-\left(\frac{s+1}{2}\right)[I]\right)^T+$$

$$\frac{(\theta\Delta t')^2}{6}\left([e_1]-\left(\frac{s+1}{2}\right)[I]\right)\left(\left[\overline{m}_1^\infty\right]+2\left[m_o^\infty\right]\right)-\frac{(\theta\Delta t')^3}{6}\left([e_2]-[e_1][e_1]^T\right)+\frac{(\theta\Delta t')^2}{2}\left(\left[\overline{m}_1^\infty\right]+\left[m_o^\infty\right]\right)-\theta\Delta t'[m_o]=[0]$$

Multiplying the above equation by $\frac{6}{\theta\Delta t'}$ and rearranging we get:

$$\left[\overline{m}_1^\infty\right][I]\left[\overline{m}_1^\infty\right]+[B_1]\left[\overline{m}_1^\infty\right]+\left[\overline{m}_1^\infty\right]\left[B_1^T\right]+[C_1]=[0] \qquad (2.101)$$

Where, $[B_1]=2\left[m_o^\infty\right]+1.5\theta\Delta t'[I]+\theta\Delta t'\left([e_1]-\frac{(s+1)}{2}[I]\right)$ (2.102)

$$[C_1]=\left[m_o^\infty\right]^2+3\theta\Delta t'\left[m_o^\infty\right]+2\theta\Delta t'\left([e_1]-\frac{(s+1)}{2}[I]\right)\left[m_o^\infty\right]+$$

$$2\theta\Delta t'\left[m_o^\infty\right]\left([e_1]-\frac{(s+1)}{2}[I]\right)^T-(\theta\Delta t')^2\left([e_2]-[e_1][e_1]^T\right)-6[m_o] \qquad (2.103)$$

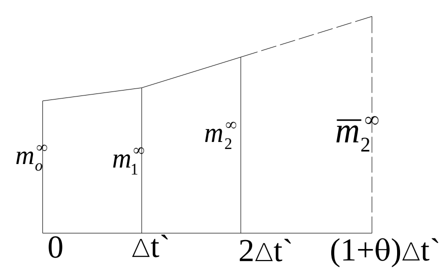
Solving equation (2.101) (*Riccati`s algebraic equation*) get $\left[\overline{m}_1^\infty\right]$ and using equation (2.88)

$$\left[m_1^\infty\right]=\frac{(\theta-1)}{\theta}\left[m_o^\infty\right]+\frac{1}{\theta}\left[\overline{m}_1^\infty\right] \qquad (2.104)$$

<u>For the second step m=2 :</u> $\left[m^\infty(\tau)\right]$ at $\Delta t'\le t\le(1+\theta)\Delta t'$

The assumed equation of the unit impulse will be:

$$\left[m^\infty(t)\right]=\begin{cases}\left[m_o^\infty\right]+t\left(\dfrac{\left[m_1^\infty\right]-\left[m_o^\infty\right]}{\Delta t'}\right) & 0\le t\le\Delta t'\\ \left[m_1^\infty\right]+(t-\Delta t')\left(\dfrac{\left[\overline{m}_2^\infty\right]-\left[m_1^\infty\right]}{\theta\Delta t'}\right) & \Delta t'\le t\le(1+\theta)\Delta t'\end{cases} \qquad (2.105)$$



<u>From equation (2.105):</u>

$$I_1=\int_0^{(1+\theta)\Delta t'}\left[m^\infty(\tau)\right]d\tau=\int_0^{\Delta t'}\left(\left[m_o^\infty\right]+\tau\left(\frac{\left[m_1^\infty\right]-\left[m_o^\infty\right]}{\Delta t'}\right)\right)d\tau+\int_{\Delta t'}^{(1+\theta)\Delta t'}\left(\left[m_1^\infty\right]+(\tau-\Delta t')\left(\frac{\left[\overline{m}_2^\infty\right]-\left[m_1^\infty\right]}{\theta\Delta t'}\right)\right)d\tau$$

$$I_1=\frac{\Delta t'}{2}\left(\left[m_o^\infty\right]+(1+\theta)\left[m_1^\infty\right]+\theta\left[\overline{m}_2^\infty\right]\right) \qquad (2.106)$$

$$I_2 = \int_0^{(1+\theta)\Delta t`}\int_0^{\tau}\left[m^{\infty}(\tau`)\right]d\tau`d\tau = \int_0^{\Delta t`}\int_0^{\tau}\left(\left[m_o^{\infty}\right]+\tau`\left(\frac{\left[m_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\Delta t`}\right)\right)d\tau`d\tau +$$

$$\int_{\Delta t`}^{(1+\theta)\Delta t`}\left(\int_0^{\Delta t`}\left(\left[m_o^{\infty}\right]+\tau`\left(\frac{\left[m_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\Delta t`}\right)\right)d\tau`+\int_{\Delta t`}^{\tau}\left(\left[m_1^{\infty}\right]+(\tau`-\Delta t`)\left(\frac{\left[\overline{m}_2^{\infty}\right]-\left[m_1^{\infty}\right]}{\theta\Delta t`}\right)\right)d\tau`\right)d\tau$$

$$I_2 = \Delta t`^2\left(\left(\frac{1}{3}+\frac{\theta}{2}\right)\left[m_o^{\infty}\right]+\left(\frac{1}{6}+\frac{\theta}{2}+\frac{\theta^2}{3}\right)\left[m_1^{\infty}\right]+\frac{\theta^2}{6}\left[\overline{m}_2^{\infty}\right]\right) \tag{2.107}$$

$$I_3 = \int_0^{(1+\theta)\Delta t`}\left[m^{\infty}(t-\tau)\right]\left[m^{\infty}(\tau)\right]d\tau = \int_0^{\Delta t`}\left(\left[m_1^{\infty}\right]+((1+\theta)\Delta t`-\tau-\Delta t`)\left(\frac{\left[\overline{m}_2^{\infty}\right]-\left[m_1^{\infty}\right]}{\theta\Delta t`}\right)\right)\left(\left[m_o^{\infty}\right]+\tau\left(\frac{\left[m_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\Delta t`}\right)\right)d\tau$$

$$+\int_{\Delta t`}^{\theta\Delta t`}\left(\left[m_1^{\infty}\right]+((1+\theta)\Delta t`-\tau-\Delta t`)\left(\frac{\left[\overline{m}_2^{\infty}\right]-\left[m_1^{\infty}\right]}{\theta\Delta t`}\right)\right)\left(\left[m_1^{\infty}\right]+(\tau-\Delta t`)\left(\frac{\left[\overline{m}_2^{\infty}\right]-\left[m_1^{\infty}\right]}{\theta\Delta t`}\right)\right)d\tau +$$

$$\int_{\theta\Delta t`}^{(1+\theta)\Delta t`}\left(\left[m_o^{\infty}\right]+((1+\theta)\Delta t`-\tau)\left(\frac{\left[m_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\Delta t`}\right)\right)\left(\left[m_1^{\infty}\right]+(\tau-\Delta t`)\left(\frac{\left[\overline{m}_2^{\infty}\right]-\left[m_1^{\infty}\right]}{\theta\Delta t`}\right)\right)d\tau$$

$$I_3 = \Delta t`\left(\begin{array}{l}\frac{1}{6\theta^2}\left[\overline{m}_2^{\infty}\right](\theta-1)^3[I]\left[\overline{m}_2^{\infty}\right]+\left(\left(\frac{1}{2}-\frac{1}{6\theta}\right)\left[m_o^{\infty}\right]+\left(\frac{\theta}{3}-\frac{1}{3\theta}+\frac{1}{6\theta^2}\right)\left[m_1^{\infty}\right]\right)\left[\overline{m}_2^{\infty}\right]+\\ \left[\overline{m}_2^{\infty}\right]\left(\left(\frac{1}{2}-\frac{1}{6\theta}\right)\left[m_o^{\infty}\right]+\left(\frac{\theta}{3}-\frac{1}{3\theta}+\frac{1}{6\theta^2}\right)\left[m_1^{\infty}\right]\right)+\left[m_1^{\infty}\right]^2\left(\frac{1}{2}+\frac{\theta}{6}+\frac{1}{6\theta}-\frac{1}{6\theta^2}\right)+\\ \left[m_1^{\infty}\right]\left[m_o^{\infty}\right]\left(\frac{1}{6\theta}\right)+\left[m_o^{\infty}\right]\left[m_1^{\infty}\right]\left(\frac{1}{6\theta}\right)\end{array}\right) \tag{2.108}$$

Substitute by (2.106), (2.107) and (2.108) in (2.87)

$$\Delta t`\left(\begin{array}{l}\frac{1}{6\theta^2}\left[\overline{m}_2^{\infty}\right](\theta-1)^3[I]\left[\overline{m}_2^{\infty}\right]+\left(\left(\frac{1}{2}-\frac{1}{6\theta}\right)\left[m_o^{\infty}\right]+\left(\frac{\theta}{3}-\frac{1}{3\theta}+\frac{1}{6\theta^2}\right)\left[m_1^{\infty}\right]\right)\left[\overline{m}_2^{\infty}\right]+\\ \left[\overline{m}_2^{\infty}\right]\left(\left(\frac{1}{2}-\frac{1}{6\theta}\right)\left[m_o^{\infty}\right]+\left(\frac{\theta}{3}-\frac{1}{3\theta}+\frac{1}{6\theta^2}\right)\left[m_1^{\infty}\right]\right)+\left[m_1^{\infty}\right]^2\left(\frac{1}{2}+\frac{\theta}{6}+\frac{1}{6\theta}-\frac{1}{6\theta^2}\right)+\\ \left[m_1^{\infty}\right]\left[m_o^{\infty}\right]\left(\frac{1}{6\theta}\right)+\left[m_o^{\infty}\right]\left[m_1^{\infty}\right]\left(\frac{1}{6\theta}\right)\end{array}\right)+$$

$$\Delta t`^2\left(\left(\frac{1}{3}+\frac{\theta}{2}\right)\left[m_o^{\infty}\right]+\left(\frac{1}{6}+\frac{\theta}{2}+\frac{\theta^2}{3}\right)\left[m_1^{\infty}\right]+\frac{\theta^2}{6}\left[\overline{m}_2^{\infty}\right]\right)\left([e_1]-\left(\frac{s+1}{2}\right)[I]\right)^T +$$

$$\Delta t`^2\left([e_1]-\left(\frac{s+1}{2}\right)[I]\right)\left(\left(\frac{1}{3}+\frac{\theta}{2}\right)\left[m_o^{\infty}\right]+\left(\frac{1}{6}+\frac{\theta}{2}+\frac{\theta^2}{3}\right)\left[m_1^{\infty}\right]+\frac{\theta^2}{6}\left[\overline{m}_2^{\infty}\right]\right)-$$

$$\frac{((1+\theta)\Delta t`)^3}{6}\left([e_2]-[e_1][e_1]^T\right)+(1+\theta)\frac{\Delta t`^2}{2}\left(\left[m_o^{\infty}\right]+(1+\theta)\left[m_1^{\infty}\right]+\theta\left[\overline{m}_2^{\infty}\right]\right)-(1+\theta)\Delta t`[m_o]=[0]$$

Multiplying the above equation by $\frac{6\theta^2}{\Delta t`}$ and rearranging we get:

$$\left[\overline{m}_2^\infty\right](\theta-1)^3[I]\left[\overline{m}_2^\infty\right]+[B_2]\left[\overline{m}_2^\infty\right]+\left[\overline{m}_2^\infty\right]\left[B_2^T\right]+[C_2]=[0] \tag{2.109}$$

Where,

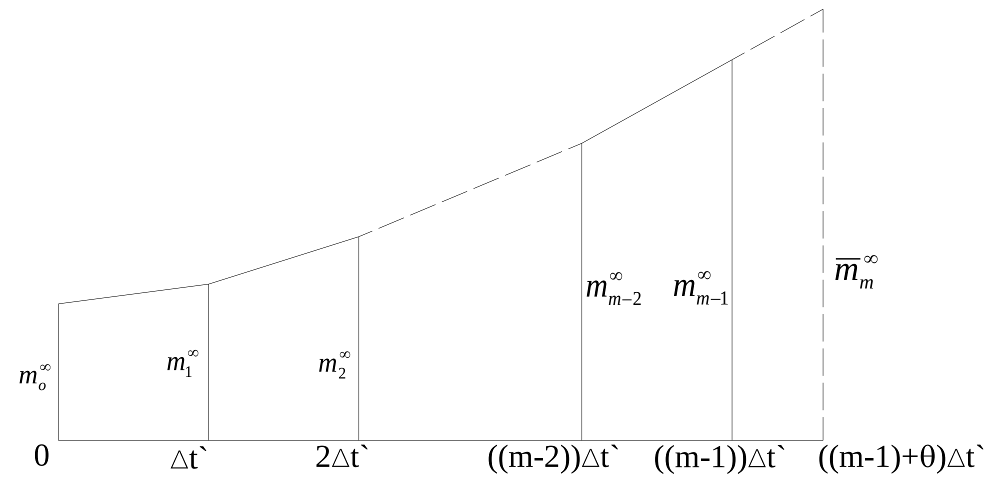
$$[B_2]=\left(3\theta^2-\theta\right)\left[m_o^\infty\right]+\left(2\theta^3-2\theta+1\right)\left[m_1^\infty\right]+1.5\theta^3(1+\theta)\Delta t`[I]+\theta^4\Delta t`\left([e_1]-\frac{(s+1)}{2}[I]\right) \tag{2.110}$$

$$\begin{aligned}[C_2]=&\left(\theta^3+3\theta^2+\theta-1\right)\left[m_1^\infty\right]^2+\Delta t`\left([e_1]-\frac{(s+1)}{2}[I]\right)\begin{pmatrix}\left(3\theta^3+2\theta^2\right)\left[m_o^\infty\right]+\\ \left(2\theta^4+3\theta^3+\theta^2\right)\left[m_1^\infty\right]\end{pmatrix}+\\ &\Delta t`\left(\left(3\theta^3+2\theta^2\right)\left[m_o^\infty\right]+\left(2\theta^4+3\theta^3+\theta^2\right)\left[m_1^\infty\right]\right)\left([e_1]-\frac{(s+1)}{2}[I]\right)^T+\\ &\theta\left(\left[m_1^\infty\right]\left[m_o^\infty\right]+\left[m_o^\infty\right]\left[m_1^\infty\right]\right)+\theta^2(1+\theta)\begin{pmatrix}3\Delta t`\left[m_o^\infty\right]+3\Delta t`(1+\theta)\left[m_1^\infty\right]-\\ (1+\theta)^2\Delta t`^2\left([e_2]-[e_1][e_1]^T\right)-6[m_o]\end{pmatrix}\end{aligned} \tag{2.111}$$

Solving equation (2.109) (*Riccati`s algebraic equation*) get $\left[\overline{m}_1^\infty\right]$ and using equation (2.88)

$$\left[m_2^\infty\right]=\frac{(\theta-1)}{\theta}\left[m_1^\infty\right]+\frac{1}{\theta}\left[\overline{m}_2^\infty\right] \tag{2.112}$$

For $m \in [3, m]$: $([m^{\infty}(\tau)]$ at $2\Delta t` \le t \le ((m-1)+\theta)\Delta t`)$



The assumed equation of the unit impulse will be:

$$[m^{\infty}(t)] = \begin{cases} [m_o^{\infty}] + t\left(\dfrac{[m_1^{\infty}] - [m_o^{\infty}]}{\Delta t`}\right) & 0 \le t \le \Delta t` \\ [m_1^{\infty}] + (t - \Delta t`)\left(\dfrac{[m_2^{\infty}] - [m_1^{\infty}]}{\Delta t`}\right) & \Delta t` \le t \le 2\Delta t` \\ \vdots & \\ \vdots & \\ [m_{m-2}^{\infty}] + (t - (m-2)\Delta t`)\left(\dfrac{[m_{m-1}^{\infty}] - [m_{m-2}^{\infty}]}{\Delta t`}\right) & (m-2)\Delta t` \le t \le (m-1)\Delta t` \\ [m_{m-1}^{\infty}] + (t - (m-1)\Delta t`)\left(\dfrac{[\overline{m}_m^{\infty}] - [m_{m-1}^{\infty}]}{\theta\Delta t`}\right) & (m-1)\Delta t` \le t \le (m-1+\theta)\Delta t` \end{cases} \tag{2.113}$$

From equation (2.113):

$$I_1 = \int_0^{((m-1)+\theta)\Delta t`} [m^{\infty}(\tau)]d\tau = \sum_{j=1}^{j=m-1}\left(\frac{[m_j^{\infty}] + [m_{j-1}^{\infty}]}{2}\right)\Delta t` + \left(\frac{[\overline{m}_m^{\infty}] + [m_{m-1}^{\infty}]}{2}\right)\theta\Delta t`$$

$$I_1 = I_{m-1} + \left(\frac{[\overline{m}_m^{\infty}] + [m_{m-1}^{\infty}]}{2}\right)\theta\Delta t` \tag{2.114}$$

$$I_2 = \int_0^{((m-1)+\theta)\Delta t`} \int_0^{\tau} [m^{\infty}(\tau`)]d\tau`d\tau = \sum_{j=1}^{j=m-1} Y_j + Y_m \tag{2.115}$$

Where, $Y_j = \int_{(j-1)\Delta t`}^{j\Delta t`} \left(\sum_{k=1}^{k=j-1} A_k + B \right) d\tau$ (2.116)

$$Y_m = \int_{(m-1)\Delta t`}^{((m-1)+\theta)\Delta t`} \left(\sum_{k=1}^{k=m-1} A_k + D \right) d\tau \qquad (2.117)$$

And, $A_k = \int_{(k-1)\Delta t`}^{k\Delta t`} \left(\left[m_{k-1}^{\infty}\right] + \left(\tau` - (k-1)\Delta t`\right) \left(\frac{\left[m_k^{\infty}\right] - \left[m_{k-1}^{\infty}\right]}{\Delta t`} \right) \right) d\tau`$ (2.118)

$$B = \int_{(j-1)\Delta t`}^{\tau} \left( \left[m_{j-1}^{\infty}\right] + \left(\tau` - (j-1)\Delta t`\right) \left( \frac{\left[m_j^{\infty}\right] - \left[m_{j-1}^{\infty}\right]}{\Delta t`} \right) \right) d\tau` \qquad (2.119)$$

$$D = \int_{(m-1)\Delta t`}^{\tau} \left( \left[m_{m-1}^{\infty}\right] + \left(\tau` - (m-1)\Delta t`\right) \left( \frac{\left[\overline{m}_m^{\infty}\right] - \left[m_{m-1}^{\infty}\right]}{\theta\Delta t`} \right) \right) d\tau` \qquad (2.120)$$

From (2.118)

$$A_k = \Delta t` \left(\left[m_{k-1}^{\infty}\right] + \frac{1}{2}\left(\left[m_k^{\infty}\right] - \left[m_{k-1}^{\infty}\right] \right) \right) \qquad (2.121)$$

From (2.119)

$$B = \left[m_{j-1}^{\infty}\right]\left(\tau - (j-1)\Delta t`\right) + \left( \frac{\tau^2}{2} - (j-1)\Delta t`\tau + \frac{(j-1)^2 \Delta t`^2}{2} \right) \left( \frac{\left[m_j^{\infty}\right] - \left[m_{j-1}^{\infty}\right]}{\Delta t`} \right) \qquad (2.122)$$

From (2.120) similar to (2.119)

$$D = \left[m_{m-1}^{\infty}\right]\left(\tau - (m-1)\Delta t`\right) + \left( \frac{\tau^2}{2} - (m-1)\Delta t`\tau + \frac{(m-1)^2 \Delta t`^2}{2} \right) \left( \frac{\left[\overline{m}_m^{\infty}\right] - \left[m_{m-1}^{\infty}\right]}{\theta\Delta t`} \right) \qquad (2.123)$$

Substitute by (2.121) and (2.122) in (2.116)

$$Y_j = \int_{(j-1)\Delta t`}^{j\Delta t`} \left(\begin{array}{l} \sum_{k=1}^{k=j-1} \Delta t` \left( \left[m_{k-1}^{\infty}\right] + \frac{1}{2}\left( \left[m_k^{\infty}\right] - \left[m_{k-1}^{\infty}\right] \right) \right) + \left[m_{j-1}^{\infty}\right]\left(\tau - (j-1)\Delta t`\right) + \\ \left(\frac{\tau^2}{2} - (j-1)\Delta t`\tau + \frac{(j-1)^2 \Delta t`^2}{2} \right) \left(\frac{\left[m_j^{\infty}\right] - \left[m_{j-1}^{\infty}\right]}{\Delta t`} \right) \end{array} \right) d\tau$$

$$Y_j = \Delta t`^2 \left(\sum_{k=1}^{k=j-1} \left(\left[m_{k-1}^{\infty}\right] + \frac{1}{2}\left(\left[m_k^{\infty}\right] - \left[m_{k-1}^{\infty}\right] \right) \right) + \frac{1}{3}\left[m_{j-1}^{\infty}\right] + \frac{1}{6}\left[m_j^{\infty}\right] \right) \qquad (2.124)$$

Substitute by (2.121) and (2.123) in (2.117)

$$Y_m = \int_{(m-1)\Delta t`}^{((m-1)+\theta)\Delta t`} \left(\begin{array}{l} \sum_{k=1}^{k=m-1} \Delta t` \left( \left[m_{k-1}^{\infty}\right] + \frac{1}{2}\left( \left[m_k^{\infty}\right] - \left[m_{k-1}^{\infty}\right] \right) \right) + \left[m_{m-1}^{\infty}\right]\left(\tau - (m-1)\Delta t`\right) + \\ \left(\frac{\tau^2}{2} - (m-1)\Delta t`\tau + \frac{(m-1)^2 \Delta t`^2}{2} \right) \left(\frac{\left[\overline{m}_m^{\infty}\right] - \left[m_{m-1}^{\infty}\right]}{\theta\Delta t`} \right) \end{array} \right) d\tau$$

$$Y_m = \Delta t^{`2}\left(\sum_{k=1}^{k=m-1}\theta\left(\left[m_{k-1}^{\infty}\right]+\frac{1}{2}\left(\left[m_k^{\infty}\right]-\left[m_{k-1}^{\infty}\right]\right)\right)+\frac{\theta^2}{3}\left[m_{m-1}^{\infty}\right]+\frac{\theta^2}{6}\left[\overline{m}_m^{\infty}\right]\right) \quad (2.125)$$

Substitute by (2.124) and (2.125) in equation (2.115)

$$I_2 = \Delta t^{`2}\left(\begin{array}{l}\sum_{j=1}^{j=m-1}\left(\sum_{k=1}^{k=j-1}\left(\left[m_{k-1}^{\infty}\right]+\frac{1}{2}\left(\left[m_k^{\infty}\right]-\left[m_{k-1}^{\infty}\right]\right)\right)+\frac{1}{3}\left[m_{j-1}^{\infty}\right]+\frac{1}{6}\left[m_j^{\infty}\right]\right)+\\ \sum_{k=1}^{k=m-1}\theta\left(\left[m_{k-1}^{\infty}\right]+\frac{1}{2}\left(\left[m_k^{\infty}\right]-\left[m_{k-1}^{\infty}\right]\right)\right)+\frac{\theta^2}{3}\left[m_{m-1}^{\infty}\right]+\frac{\theta^2}{6}\left[\overline{m}_m^{\infty}\right]\end{array}\right) \quad (2.126)$$

$$I_3 = \int_0^{((m-1)+\theta)\Delta t`}\left[m^{\infty}\left(((m-1)+\theta)\Delta t`-\tau\right)\right]\left[m^{\infty}(\tau)\right]d\tau = C+E+F+\sum_{j=2}^{j=m-2}(G+H)+N+Y \quad (2.127)$$

Where,

$$C = \int_0^{\Delta t`}\left(\left[m_{m-1}^{\infty}\right]+\left(\begin{array}{l}((m-1)+\theta)\Delta t`-\\ \tau-(m-1)\Delta t`\end{array}\right)\left(\frac{\left[\overline{m}_m^{\infty}\right]-\left[m_{m-1}^{\infty}\right]}{\theta\Delta t`}\right)\right)\left(\left[m_o^{\infty}\right]+\tau\left(\frac{\left[m_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\Delta t`}\right)\right)d\tau \quad (2.128)$$

$$E = \int_{\Delta t`}^{\theta\Delta t`}\left(\left[m_{m-1}^{\infty}\right]+\left(\begin{array}{l}((m-1)+\theta)\Delta t`-\\ \tau-(m-1)\Delta t`\end{array}\right)\left(\frac{\left[\overline{m}_m^{\infty}\right]-\left[m_{m-1}^{\infty}\right]}{\theta\Delta t`}\right)\right)\left(\begin{array}{l}\left[m_1^{\infty}\right]+\\ (\tau-\Delta t`)\left(\frac{\left[m_2^{\infty}\right]-\left[m_1^{\infty}\right]}{\Delta t`}\right)\end{array}\right)d\tau \quad (2.129)$$

$$F = \int_{\theta\Delta t`}^{2\Delta t`}\left(\left[m_{m-2}^{\infty}\right]+\left(\begin{array}{l}((m-1)+\theta)\Delta t`-\\ \tau-(m-2)\Delta t`\end{array}\right)\left(\frac{\left[m_{m-1}^{\infty}\right]-\left[m_{m-2}^{\infty}\right]}{\Delta t`}\right)\right)\left(\begin{array}{l}\left[m_1^{\infty}\right]+\\ (\tau-\Delta t`)\left(\frac{\left[m_2^{\infty}\right]-\left[m_1^{\infty}\right]}{\Delta t`}\right)\end{array}\right)d\tau \quad (2.130)$$

$$G = \int_{j\Delta t`}^{((j-1)+\theta)\Delta t`}\left(\begin{array}{l}\left[m_{m-j}^{\infty}\right]+\\ \left(\begin{array}{l}((m-1)+\theta)\Delta t`-\\ \tau-(m-j)\Delta t`\end{array}\right)\left(\frac{\left[m_{m-j+1}^{\infty}\right]-\left[m_{m-j}^{\infty}\right]}{\Delta t`}\right)\end{array}\right)\left(\begin{array}{l}\left[m_j^{\infty}\right]+\\ (\tau-j\Delta t`)\left(\frac{\left[m_{j+1}^{\infty}\right]-\left[m_j^{\infty}\right]}{\Delta t`}\right)\end{array}\right)d\tau \quad (2.131)$$

$$H = \int_{((j-1)+\theta)\Delta t`}^{(j+1)\Delta t`}\left(\begin{array}{l}\left[m_{m-j-1}^{\infty}\right]+\\ \left(\begin{array}{l}((m-1)+\theta)\Delta t`-\\ \tau-(m-j-1)\Delta t`\end{array}\right)\left(\frac{\left[m_{m-j}^{\infty}\right]-\left[m_{m-j-1}^{\infty}\right]}{\Delta t`}\right)\end{array}\right)\left(\begin{array}{l}\left[m_j^{\infty}\right]+\\ (\tau-j\Delta t`)\left(\frac{\left[m_{j+1}^{\infty}\right]-\left[m_j^{\infty}\right]}{\Delta t`}\right)\end{array}\right)d\tau \quad (2.132)$$

$$N = \int_{(m-1)\Delta t`}^{((m-2)+\theta)\Delta t`}\left(\begin{array}{l}\left[m_1^{\infty}\right]+\\ \left(\begin{array}{l}((m-1)+\theta)\Delta t`-\\ \tau-\Delta t`\end{array}\right)\left(\frac{\left[m_2^{\infty}\right]-\left[m_1^{\infty}\right]}{\Delta t`}\right)\end{array}\right)\left(\begin{array}{l}\left[m_{m-1}^{\infty}\right]+\\ (\tau-(m-1)\Delta t`)\left(\frac{\left[\overline{m}_m^{\infty}\right]-\left[m_{m-1}^{\infty}\right]}{\theta\Delta t`}\right)\end{array}\right)d\tau \quad (2.133)$$

$$Y = \int_{((m-2)+\theta)\Delta t`}^{((m-1)+\theta)\Delta t`}\left(\begin{array}{l}\left[m_o^{\infty}\right]+\\ (((m-1)+\theta)\Delta t`-\tau)\left(\frac{\left[m_1^{\infty}\right]-\left[m_o^{\infty}\right]}{\Delta t`}\right)\end{array}\right)\left(\begin{array}{l}\left[m_{m-1}^{\infty}\right]+\\ (\tau-(m-1)\Delta t`)\left(\frac{\left[\overline{m}_m^{\infty}\right]-\left[m_{m-1}^{\infty}\right]}{\theta\Delta t`}\right)\end{array}\right)d\tau \quad (2.134)$$

From (2.128)

$$C = \Delta t \left(\frac{1}{6\theta}\left[m_{m-1}^{\infty}\right]\left[m_{o}^{\infty}\right] + \frac{1}{3\theta}\left[m_{m-1}^{\infty}\right]\left[m_{1}^{\infty}\right] + \left(\frac{1}{2} - \frac{1}{6\theta}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{o}^{\infty}\right] + \left(\frac{1}{2} - \frac{1}{3\theta}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{1}^{\infty}\right] \right) \tag{2.135}$$

From (2.129)

$$E = \Delta t \left(\begin{array}{l} \left(\theta - \frac{2}{3\theta} - \frac{\theta^2}{3}\right)\left[m_{m-1}^{\infty}\right]\left[m_{1}^{\infty}\right] + \left(\frac{\theta^2}{3} - \frac{\theta}{2} + \frac{1}{6\theta}\right)\left[m_{m-1}^{\infty}\right]\left[m_{2}^{\infty}\right] + \\ \left(\theta - \frac{3}{2} + \frac{2}{3\theta} - \frac{\theta^2}{6}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{1}^{\infty}\right] + \left(\frac{\theta^2}{6} - \frac{\theta}{2} + \frac{1}{2} - \frac{1}{6\theta}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{2}^{\infty}\right] \end{array} \right) \tag{2.136}$$

From (2.130)

$$F = \Delta t \left(\begin{array}{l} \left(\frac{4}{3} - 2\theta + \theta^2 - \frac{\theta^3}{6}\right)\left[m_{m-2}^{\infty}\right]\left[m_{1}^{\infty}\right] + \left(\frac{\theta^3}{6} - \frac{\theta^2}{2} + \frac{2}{3}\right)\left[m_{m-2}^{\infty}\right]\left[m_{2}^{\infty}\right] + \\ \left(\frac{2}{3} - \frac{\theta^2}{2} + \frac{\theta^3}{6}\right)\left[m_{m-1}^{\infty}\right]\left[m_{1}^{\infty}\right] + \left(\theta - \frac{2}{3} - \frac{\theta^3}{6}\right)\left[m_{m-1}^{\infty}\right]\left[m_{2}^{\infty}\right] \end{array} \right) \tag{2.137}$$

From (2.131)

$$G = \Delta t \left(\begin{array}{l} \left(\frac{7}{2}\theta - \frac{13}{6} - \frac{3\theta^2}{2} + \frac{\theta^3}{6}\right)\left[m_{m-j}^{\infty}\right]\left[m_{j}^{\infty}\right] + \left(\theta^2 - \frac{3\theta}{2} + \frac{2}{3} - \frac{\theta^3}{6}\right)\left[m_{m-j}^{\infty}\right]\left[m_{j+1}^{\infty}\right] + \\ \left(\theta^2 - \frac{3\theta}{2} + \frac{2}{3} - \frac{\theta^3}{6}\right)\left[m_{m-j+1}^{\infty}\right]\left[m_{j}^{\infty}\right] + \left(\frac{\theta^3}{6} - \frac{\theta^2}{2} + \frac{\theta}{2} - \frac{1}{6}\right)\left[m_{m-j+1}^{\infty}\right]\left[m_{j+1}^{\infty}\right] \end{array} \right) \tag{2.138}$$

From (2.132)

$$H = \Delta t \left(\begin{array}{l} \left(\frac{4}{3} - 2\theta + \theta^2 - \frac{\theta^3}{6}\right)\left[m_{m-j-1}^{\infty}\right]\left[m_{j}^{\infty}\right] + \left(\frac{2}{3} - \frac{\theta^2}{2} + \frac{\theta^3}{6}\right)\left[m_{m-j-1}^{\infty}\right]\left[m_{j+1}^{\infty}\right] + \\ \left(\frac{2}{3} - \frac{\theta^2}{2} + \frac{\theta^3}{6}\right)\left[m_{m-j}^{\infty}\right]\left[m_{j}^{\infty}\right] + \left(\theta - \frac{\theta^3}{6} - \frac{2}{3}\right)\left[m_{m-j}^{\infty}\right]\left[m_{j+1}^{\infty}\right] \end{array} \right) \tag{2.139}$$

From (2.133)

$$N = \Delta t \left(\begin{array}{l} \left(\theta - \frac{2}{3\theta} - \frac{\theta^2}{3}\right)\left[m_{1}^{\infty}\right]\left[m_{m-1}^{\infty}\right] + \left(\theta - \frac{3}{2} + \frac{2}{3\theta} - \frac{\theta^2}{6}\right)\left[m_{1}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right] + \\ \left(\frac{\theta^2}{3} - \frac{\theta}{2} + \frac{1}{6\theta}\right)\left[m_{2}^{\infty}\right]\left[m_{m-1}^{\infty}\right] + \frac{(\theta - 1)^3}{6\theta}\left[m_{2}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right] \end{array} \right) \tag{2.140}$$

From (2.134)

$$Y = \Delta t \left(\begin{array}{l} \left(\frac{1}{6\theta}\right)\left[m_{o}^{\infty}\right]\left[m_{m-1}^{\infty}\right] + \left(\frac{1}{2} - \frac{1}{6\theta}\right)\left[m_{o}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right] + \left(\frac{1}{3\theta}\right)\left[m_{1}^{\infty}\right]\left[m_{m-1}^{\infty}\right] + \\ \left(\frac{1}{2} - \frac{1}{3\theta}\right)\left[m_{1}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right] \end{array} \right) \tag{2.141}$$

Substitute by equation (2.135) through (2.141) in equation (2.127) we get:

$$I_3 = \Delta t \left(\begin{aligned} & \left(\frac{1}{6\theta}\right)\left[m_{m-1}^{\infty}\right]\left[m_{o}^{\infty}\right] + \left(-\frac{1}{3\theta} + \theta - \frac{5\theta^2}{6} + \frac{2}{3} + \frac{\theta^3}{6}\right)\left[m_{m-1}^{\infty}\right]\left[m_{1}^{\infty}\right] + \\ & \left(\frac{1}{2} - \frac{1}{6\theta}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{o}^{\infty}\right] + \left(-1 + \frac{1}{3\theta} + \theta - \frac{\theta^2}{6}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{1}^{\infty}\right] + \\ & \left(\frac{\theta^2}{3} + \frac{\theta}{2} + \frac{1}{6\theta} - \frac{2}{3} - \frac{\theta^3}{6}\right)\left[m_{m-1}^{\infty}\right]\left[m_{2}^{\infty}\right] + \left(\frac{\theta^2}{6} - \frac{\theta}{2} + \frac{1}{2} - \frac{1}{6\theta}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{2}^{\infty}\right] + \\ & \left(\frac{4}{3} - 2\theta + \theta^2 - \frac{\theta^3}{6}\right)\left[m_{m-2}^{\infty}\right]\left[m_{1}^{\infty}\right] + \left(\frac{\theta^3}{6} - \frac{\theta^2}{2} + \frac{2}{3}\right)\left[m_{m-2}^{\infty}\right]\left[m_{2}^{\infty}\right] + \\ & \sum_{j=2}^{j=m-2} \left(\begin{aligned} & \left(\frac{7}{2}\theta - \frac{3}{2} - 2\theta^2 + \frac{\theta^3}{3}\right)\left[m_{m-j}^{\infty}\right]\left[m_{j}^{\infty}\right] + \left(\theta^2 - \frac{\theta}{2} - \frac{\theta^3}{3}\right)\left[m_{m-j}^{\infty}\right]\left[m_{j+1}^{\infty}\right] + \\ & \left(\theta^2 - \frac{3\theta}{2} + \frac{2}{3} - \frac{\theta^3}{6}\right)\left[m_{m-j+1}^{\infty}\right]\left[m_{j}^{\infty}\right] + \left(\frac{\theta^3}{6} - \frac{\theta^2}{2} + \frac{\theta}{2} - \frac{1}{6}\right)\left[m_{m-j+1}^{\infty}\right]\left[m_{j+1}^{\infty}\right] \\ & + \left(\frac{4}{3} - 2\theta + \theta^2 - \frac{\theta^3}{6}\right)\left[m_{m-j-1}^{\infty}\right]\left[m_{j}^{\infty}\right] + \left(\frac{2}{3} - \frac{\theta^2}{2} + \frac{\theta^3}{6}\right)\left[m_{m-j-1}^{\infty}\right]\left[m_{j+1}^{\infty}\right] \end{aligned} \right) \\ & + \left(\theta - \frac{2}{3\theta} - \frac{\theta^2}{3} + \frac{1}{3\theta}\right)\left[m_{1}^{\infty}\right]\left[m_{m-1}^{\infty}\right] + \left(\theta - 1 + \frac{1}{3\theta} - \frac{\theta^2}{6}\right)\left[m_{1}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right] + \\ & \left(\frac{\theta^2}{3} - \frac{\theta}{2} + \frac{1}{6\theta}\right)\left[m_{2}^{\infty}\right]\left[m_{m-1}^{\infty}\right] + \frac{(\theta-1)^3}{6\theta}\left[m_{2}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right] + \left(\frac{1}{6\theta}\right)\left[m_{o}^{\infty}\right]\left[m_{m-1}^{\infty}\right] + \\ & \left(\frac{1}{2} - \frac{1}{6\theta}\right)\left[m_{o}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right] \end{aligned} \right) \qquad (2.142)$$

Substitute by (2.114), (2.126) and (2.142) in (2.87)

$$
\Delta t` \left(\begin{array}{l}
\left(\frac{1}{6\theta}\right)\left[m_{m-1}^{\infty}\right]\left[m_{o}^{\infty}\right]+\left(-\frac{1}{3\theta}+\theta-\frac{5\theta^{2}}{6}+\frac{2}{3}+\frac{\theta^{3}}{6}\right)\left[m_{m-1}^{\infty}\right]\left[m_{1}^{\infty}\right]+ \\
\left(\frac{1}{2}-\frac{1}{6\theta}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{o}^{\infty}\right]+\left(-1+\frac{1}{3\theta}+\theta-\frac{\theta^{2}}{6}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{1}^{\infty}\right]+ \\
\left(\frac{\theta^{2}}{3}+\frac{\theta}{2}+\frac{1}{6\theta}-\frac{2}{3}-\frac{\theta^{3}}{6}\right)\left[m_{m-1}^{\infty}\right]\left[m_{2}^{\infty}\right]+\left(\frac{\theta^{2}}{6}-\frac{\theta}{2}+\frac{1}{2}-\frac{1}{6\theta}\right)\left[\overline{m}_{m}^{\infty}\right]\left[m_{2}^{\infty}\right]+ \\
\left(\frac{4}{3}-2\theta+\theta^{2}-\frac{\theta^{3}}{6}\right)\left[m_{m-2}^{\infty}\right]\left[m_{1}^{\infty}\right]+\left(\frac{\theta^{3}}{6}-\frac{\theta^{2}}{2}+\frac{2}{3}\right)\left[m_{m-2}^{\infty}\right]\left[m_{2}^{\infty}\right]+ \\
\sum_{j=2}^{j=m-2}\left(\begin{array}{l}
\left(\frac{7}{2}\theta-\frac{3}{2}-2\theta^{2}+\frac{\theta^{3}}{3}\right)\left[m_{m-j}^{\infty}\right]\left[m_{j}^{\infty}\right]+\left(\theta^{2}-\frac{\theta}{2}-\frac{\theta^{3}}{3}\right)\left[m_{m-j}^{\infty}\right]\left[m_{j+1}^{\infty}\right]+ \\
\left(\theta^{2}-\frac{3\theta}{2}+\frac{2}{3}-\frac{\theta^{3}}{6}\right)\left[m_{m-j+1}^{\infty}\right]\left[m_{j}^{\infty}\right]+\left(\frac{\theta^{3}}{6}-\frac{\theta^{2}}{2}+\frac{\theta}{2}-\frac{1}{6}\right)\left[m_{m-j+1}^{\infty}\right]\left[m_{j+1}^{\infty}\right] \\
+\left(\frac{4}{3}-2\theta+\theta^{2}-\frac{\theta^{3}}{6}\right)\left[m_{m-j-1}^{\infty}\right]\left[m_{j}^{\infty}\right]+\left(\frac{2}{3}-\frac{\theta^{2}}{2}+\frac{\theta^{3}}{6}\right)\left[m_{m-j-1}^{\infty}\right]\left[m_{j+1}^{\infty}\right]
\end{array}\right) \\
+\left(\theta-\frac{2}{3\theta}-\frac{\theta^{2}}{3}+\frac{1}{3\theta}\right)\left[m_{1}^{\infty}\right]\left[m_{m-1}^{\infty}\right]+\left(\theta-1+\frac{1}{3\theta}-\frac{\theta^{2}}{6}\right)\left[m_{1}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right]+ \\
\left(\frac{\theta^{2}}{3}-\frac{\theta}{2}+\frac{1}{6\theta}\right)\left[m_{2}^{\infty}\right]\left[m_{m-1}^{\infty}\right]+\frac{(\theta-1)^{3}}{6\theta}\left[m_{2}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right]+\left(\frac{1}{6\theta}\right)\left[m_{o}^{\infty}\right]\left[m_{m-1}^{\infty}\right]+\left(\frac{1}{2}-\frac{1}{6\theta}\right)\left[m_{o}^{\infty}\right]\left[\overline{m}_{m}^{\infty}\right]+ \\
\Delta t`\left(\begin{array}{l}
\sum_{j=1}^{j=m-1}\left(\sum_{k=1}^{k=j-1}\left(\left[m_{k-1}^{\infty}\right]+\frac{1}{2}\left(\left[m_{k}^{\infty}\right]-\left[m_{k-1}^{\infty}\right]\right)\right)+\frac{1}{3}\left[m_{j-1}^{\infty}\right]+\frac{1}{6}\left[m_{j}^{\infty}\right]\right)+ \\
\sum_{k=1}^{k=m-1}\theta\left(\left[m_{k-1}^{\infty}\right]+\frac{1}{2}\left(\left[m_{k}^{\infty}\right]-\left[m_{k-1}^{\infty}\right]\right)\right)+\frac{\theta^{2}}{3}\left[m_{m-1}^{\infty}\right]+\frac{\theta^{2}}{6}\left[\overline{m}_{m}^{\infty}\right]
\end{array}\right)\left(\left[e_{1}\right]-\frac{(s+1)}{2}\left[I\right]\right)^{T}+ \\
\Delta t`\left(\left[e_{1}\right]-\frac{(s+1)}{2}\left[I\right]\right)\left(\begin{array}{l}
\sum_{j=1}^{j=m-1}\left(\sum_{k=1}^{k=j-1}\left(\left[m_{k-1}^{\infty}\right]+\frac{1}{2}\left(\left[m_{k}^{\infty}\right]-\left[m_{k-1}^{\infty}\right]\right)\right)+\frac{1}{3}\left[m_{j-1}^{\infty}\right]+\frac{1}{6}\left[m_{j}^{\infty}\right]\right)+ \\
\sum_{k=1}^{k=m-1}\theta\left(\left[m_{k-1}^{\infty}\right]+\frac{1}{2}\left(\left[m_{k}^{\infty}\right]-\left[m_{k-1}^{\infty}\right]\right)\right)+\frac{\theta^{2}}{3}\left[m_{m-1}^{\infty}\right]+\frac{\theta^{2}}{6}\left[\overline{m}_{m}^{\infty}\right]
\end{array}\right)- \\
\frac{((m-1)+\theta)^{3}\Delta t`^{2}}{6}\left(\left[e_{2}\right]-\left[e_{1}\right]\left[e_{1}\right]^{T}\right)+((m-1)+\theta)\left(I_{m-1}+\left(\frac{\left[\overline{m}_{m}^{\infty}\right]+\left[m_{m-1}^{\infty}\right]}{2}\right)\theta\Delta t`\right)- \\
((m-1)+\theta)\left[m_{o}\right]
\end{array}\right) = [0]
$$

Multiplying the above equation by $\frac{6\theta}{\Delta t`}$ and rearranging we get:

$$3\theta^2 \bar{t}_m [\overline{m}_m^\infty] + [B_m][\overline{m}_m^\infty] + [\overline{m}_m^\infty][B_m^T] + [C_m] = [0] \tag{2.143}$$

Where,

$$[B_m] = (3\theta - 1)[m_o^\infty] + (2 - 6\theta + 6\theta^2 - \theta^3)[m_1^\infty] + (\theta - 1)^3 [m_2^\infty] + \theta^3 \Delta t` \left([e_1] - \frac{(s+1)}{2}[I] \right) \tag{2.144}$$

$$\begin{aligned}
[C_m] = {} & \theta \sum_{j=2}^{j=m-2} conv_j + \theta(2-\theta)^2 [m_{m-2}^\infty]((2-\theta)[m_1^\infty] + (\theta+1)[m_2^\infty]) + \\
& \theta \bar{t}_m \left(\frac{6I_{m-1}}{\Delta t`} + 3\theta [m_{m-1}^\infty] \right) + \left( [e_1] - \frac{(s+1)}{2}[I] \right) \left( \frac{6\theta J_{m-1}}{\Delta t`} + 6\theta^2 I_{m-1} + 2\theta^3 \Delta t` [m_{m-1}^\infty] \right) + \\
& \left(\frac{6\theta J_{m-1}}{\Delta t`} + 6\theta^2 I_{m-1} + 2\theta^3 \Delta t` [m_{m-1}^\infty] \right) \left([e_1] - \frac{(s+1)}{2}[I] \right)^T + \\
& \begin{pmatrix} [m_o^\infty] + (-2\theta^3 + 6\theta^2 - 2)[m_1^\infty] + \\ (\theta-1)^2 (2\theta+1)[m_2^\infty] \end{pmatrix} [m_{m-1}^\infty] - \frac{\theta \bar{t}_m}{\Delta t`} \left(\bar{t}_m^2 ([e_2] - [e_1][e_1]^T) + 6[m_o] \right) + \\
& [m_{m-1}^\infty] \begin{pmatrix} [m_o^\infty] + (\theta^4 - 5\theta^3 + 6\theta^2 + 4\theta - 2)[m_1^\infty] + \\ (-\theta^4 + 2\theta^3 + 3\theta^2 - 4\theta + 1)[m_2^\infty] \end{pmatrix} = 0
\end{aligned} \tag{2.145}$$

And,
$$\sum_{j=2}^{j=m-2} conv_j = \sum_{j=2}^{j=m-2} \begin{pmatrix} (\theta-1)^2 [m_{m-j+1}^\infty]((4-\theta)[m_j^\infty] + (\theta-1)[m_{j+1}^\infty]) + \\ (1 + 2(2-\theta)(\theta-1))[m_{m-j}^\infty]((3-\theta)[m_j^\infty] + \theta[m_{j+1}^\infty]) + \\ (2-\theta)^2 [m_{m-j-1}^\infty]((2-\theta)[m_j^\infty] + (\theta+1)[m_{j+1}^\infty]) \end{pmatrix} \tag{2.146}$$

$$I_{m-1} = \sum_{j=1}^{j=m-1} \left(\frac{[m_j^\infty] + [m_{j-1}^\infty]}{2} \right) \Delta t` \tag{2.147}$$

$$J_{m-1} = \sum_{j=1}^{j=m-1} I_{j-1} \Delta t` + \frac{\Delta t`^2}{6} ([m_j^\infty] + 2[m_{j-1}^\infty]) \tag{2.148}$$

Solving equation (2.143) (*Lyapunov`s algebraic equation*) get $[\overline{m}_m^\infty]$ and using equation (2.88) get $[m_m^\infty]$.

2.8. Response calculation

First we need the force-displacement relationship at the interface we will write it here again for convenience

$$\{R(t)\} = \int_0^t [M^\infty(\tau)]\{\ddot{u}(t-\tau)\} d\tau \tag{2.149}$$

As $\left[M^{\infty}(\tau)\right]$ is approximated as piece wise linear function in any interval $[t_1, t_2]$ so the above equation can be solved using integration by parts as follows [22]:

$$\int_{t_1}^{t_2}\left[M^{\infty}(\tau)\right]\{\ddot{u}(t-\tau)\}d\tau = -\left[M^{\infty}(t_2)\right]\{\dot{u}(t-t_2)\} + \left[M^{\infty}(t_1)\right]\{\dot{u}(t-t_1)\} + \int_{t_1}^{t_2}\frac{d\left[M^{\infty}(\tau)\right]}{d\tau}\{\dot{u}(t-\tau)\}d\tau$$

Since, $\left[M^{\infty}(\tau)\right]$ is piece wise linear function, therefore:

$$\frac{d\left[M^{\infty}(\tau)\right]}{d\tau} = \frac{\left[M^{\infty}(t_2)\right]-\left[M^{\infty}(t_1)\right]}{\Delta t} = \text{constant}$$

$$\int_{t_1}^{t_2}\left[M^{\infty}(\tau)\right]\{\ddot{u}(t-\tau)\}d\tau = -\left[M^{\infty}(t_2)\right]\{\dot{u}(t-t_2)\} + \left[M^{\infty}(t_1)\right]\{\dot{u}(t-t_1)\} + \frac{\left[M^{\infty}(t_2)\right]-\left[M^{\infty}(t_1)\right]}{\Delta t}\int_{t_1}^{t_2}\{\dot{u}(t-\tau)\}d\tau$$

Therefore,

$$\begin{aligned}\int_{t_1}^{t_2}\left[M^{\infty}(\tau)\right]\{\ddot{u}(t-\tau)\}d\tau = -\left[M^{\infty}(t_2)\right]\{\dot{u}(t-t_2)\} + \left[M^{\infty}(t_1)\right]\{\dot{u}(t-t_1)\} - \\ \left(\frac{\left[M^{\infty}(t_2)\right]-\left[M^{\infty}(t_1)\right]}{\Delta t}\right)\left(\{u(t-t_2)\}-\{u(t-t_1)\}\right)\end{aligned} \qquad (2.150)$$

Here we are going to use equation (2.150) to get the force in any interval $[t_1, t_2]$. Also it is noticed as said in [16] and [22] that the unit impulse starts non-linear to certain time t_M then it changes to nearly linear, so the unit impulse response function will be calculated till time t_M and also the force then an extrapolation for $\left[M^{\infty}(\tau)\right]$will be used to get the force at any time after t_M . The time t_M will be assumed so that the response results do not change for larger time.

First calculate $\{R(t)\}$ in the interval from $t = \Delta t$ to $t = \Delta t` = N\Delta t$:

Using equation (2.150), let $t_1 = (i-1)\Delta t$, $t_2 = i\Delta t$ and $t = n\Delta t$ where, $i = 1 \rightarrow n$ and $n\Delta t \in [1, N\Delta t]$

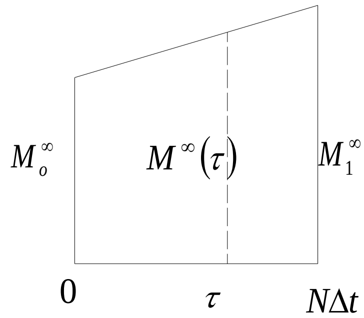
$$\{R_n\} = \sum_{i=1}^{i=n}\left(\begin{aligned}-\left[M^{\infty}(i\Delta t)\right]\{\dot{u}(n\Delta t - i\Delta t)\} + \left[M^{\infty}((i-1)\Delta t)\right]\{\dot{u}(n\Delta t-(i-1)\Delta t)\} - \\ \left(\frac{\left[M^{\infty}(i\Delta t)\right]-\left[M^{\infty}((i-1)\Delta t)\right]}{\Delta t}\right)\left(\{u(n\Delta t - i\Delta t)\}-\{u(n\Delta t-(i-1)\Delta t)\}\right)\end{aligned}\right) \qquad (2.151)$$

Since, $\left[M^{\infty}(\tau)\right]=\left[M_{o}^{\infty}\right]+\left(\frac{\left[M_{1}^{\infty}\right]-\left[M_{o}^{\infty}\right]}{N\Delta t}\right)\tau$ (2.152)

Therefore,

$$\left[M^{\infty}(i\Delta t)\right]=\left[M_{o}^{\infty}\right]+\left(\frac{\left[M_{1}^{\infty}\right]-\left[M_{o}^{\infty}\right]}{N}\right)i \tag{2.153}$$

$$\left[M^{\infty}((i-1)\Delta t)\right]=\left[M_{o}^{\infty}\right]+\left(\frac{\left[M_{1}^{\infty}\right]-\left[M_{o}^{\infty}\right]}{N}\right)(i-1) \tag{2.154}$$



Let, $\{\dot{u}(n\Delta t-i\Delta t)\}=\dot{u}_{n-i}$, $\{\dot{u}(n\Delta t-(i-1)\Delta t)\}=\dot{u}_{n-i+1}$, $\{u(n\Delta t-i\Delta t)\}=u_{n-i}$ and $\{u(n\Delta t-(i-1)\Delta t)\}=u_{n-i+1}$

Substitute in equation (2.151) by (2.153) and (2.154)

$$\{R_n\}=\sum_{i=1}^{i=n}\left(\begin{array}{l}-\left(\left[M_{o}^{\infty}\right]+\left(\frac{\left[M_{1}^{\infty}\right]-\left[M_{o}^{\infty}\right]}{N}\right)i\right)\{\dot{u}_{n-i}\}+\left(\left[M_{o}^{\infty}\right]+\left(\frac{\left[M_{1}^{\infty}\right]-\left[M_{o}^{\infty}\right]}{N}\right)(i-1)\right)\{\dot{u}_{n-i+1}\}-\\ \left(\frac{\left[M_{1}^{\infty}\right]-\left[M_{o}^{\infty}\right]}{N\Delta t}\right)(\{u_{n-i}\}-\{u_{n-i+1}\})\end{array}\right)$$

Rearranging,

$$\begin{aligned}\{R_n\}=&\left[M_{o}^{\infty}\right]\left(\left(-1+\frac{1}{N}\right)\{\dot{u}_{n-1}\}+\{\dot{u}_n\}+\frac{1}{N\Delta t}(\{u_{n-1}\}-\{u_n\})\right)+\\ &\left[M_{1}^{\infty}\right]\left(-\frac{1}{N}\{\dot{u}_{n-1}\}-\frac{1}{N\Delta t}(\{u_{n-1}\}-\{u_n\})\right)\\ &+\sum_{i=2}^{i=n}\left(\begin{array}{l}\left[M_{o}^{\infty}\right]\left(\left(-1+\frac{i}{N}\right)\{\dot{u}_{n-i}\}+\left(1-\frac{i-1}{N}\right)\{\dot{u}_{n-i+1}\}+\frac{1}{N\Delta t}(\{u_{n-i}\}-\{u_{n-i+1}\})\right)+\\ \left[M_{1}^{\infty}\right]\left(-\frac{i}{N}\{\dot{u}_{n-i}\}+\frac{i-1}{N}\{\dot{u}_{n-i+1}\}-\frac{1}{N\Delta t}(\{u_{n-i}\}-\{u_{n-i+1}\})\right)\end{array}\right)\end{aligned} \tag{2.155}$$

Using Newmark time integration method (see [15] chapter 5) the above equation can be written in one unknown $\{u_n\}$.

At any time j:

$$\{\dot{u}_j\}=\{\dot{u}_{j-1}\}+(1-\gamma)\Delta t\{\ddot{u}_{j-1}\}+\gamma\Delta t\{\ddot{u}_j\} \tag{2.156}$$

$$\{u_j\} = \{u_{j-1}\} + \Delta t\{\dot{u}_{j-1}\} + (0.5-\beta)\Delta t^2\{\ddot{u}_{j-1}\} + \beta\Delta t^2\{\ddot{u}_j\} \tag{2.157}$$

From (2.157) substitute in (2.156)

$$\{\dot{u}_j\} = a_1(\{u_j\} - \{u_{j-1}\}) - a_4\{\dot{u}_{j-1}\} - a_5\{\ddot{u}_{j-1}\} \tag{2.158}$$

Where, $a_1 = \dfrac{\gamma}{\beta\Delta t}$, $a_4 = \dfrac{\gamma}{\beta} - 1$ and $a_5 = 0.5\Delta t\left(\dfrac{\gamma}{\beta} - 2\right)$

Substitute by (2.158) in (2.155) and rearrange,

$$\begin{aligned}\{R_n\} = {} & [M_o^\infty]\left(\left(-1+\frac{1}{N}\right)\{\dot{u}_{n-1}\} + a_1\{u_n\} + \{\dot{\bar{u}}_n\} + a_{10}(\{u_{n-1}\} - \{u_n\})\right) + \\ & [M_1^\infty]\left(-\frac{1}{N}\{\dot{u}_{n-1}\} - a_{10}(\{u_{n-1}\} - \{u_n\})\right) \\ & + \sum_{i=2}^{i=n}\left([M_o^\infty]\left(\left(-1+\frac{i}{N}\right)\{\dot{u}_{n-i}\} + a_{10}\{u_{n-i}\}\right) + [M_1^\infty]\left(-\frac{i}{N}\{\dot{u}_{n-i}\} - a_{10}\{u_{n-i}\}\right)\right) + \\ & \sum_{i=2}^{i=n}\left([M_o^\infty]\left(\left(1-\frac{i-1}{N}\right)\{\dot{u}_{n-i+1}\} - a_{10}\{u_{n-i+1}\}\right) + [M_1^\infty]\left(\frac{i-1}{N}\{\dot{u}_{n-i+1}\} + a_{10}\{u_{n-i+1}\}\right)\right)\end{aligned} \tag{2.159}$$

Where, $\{\dot{\bar{u}}_n\} = -(a_1\{u_{n-1}\} + a_4\{\dot{u}_{n-1}\} + a_5\{\ddot{u}_{n-1}\})$ (2.160)

and $a_{10} = \dfrac{1}{N\Delta t}$ (2.161)

$$\begin{aligned}\{R_n\} = {} & ((a_1 - a_{10})[M_o^\infty] + a_{10}[M_1^\infty])\{u_n\} + [M_o^\infty]\left(\left(-1+\frac{1}{N}\right)\{\dot{u}_{n-1}\} + \{\dot{\bar{u}}_n\} + a_{10}\{u_{n-1}\}\right) + \\ & [M_1^\infty]\left(-\frac{1}{N}\{\dot{u}_{n-1}\} - a_{10}\{u_{n-1}\}\right) + \sum_{i=2}^{i=n}\begin{pmatrix}[M_o^\infty]\left(\left(-1+\frac{i}{N}\right)\{\dot{u}_{n-i}\} + a_{10}\{u_{n-i}\}\right) + \\ [M_1^\infty]\left(-\frac{i}{N}\{\dot{u}_{n-i}\} - a_{10}\{u_{n-i}\}\right)\end{pmatrix} + \\ & \sum_{i=1}^{i=n-1}\left([M_o^\infty]\left(\left(1-\frac{i}{N}\right)\{\dot{u}_{n-i}\} - a_{10}\{u_{n-i}\}\right) + [M_1^\infty]\left(\frac{i}{N}\{\dot{u}_{n-i}\} + a_{10}\{u_{n-i}\}\right)\right)\end{aligned}$$

It is clear that the two summations in the above equation will cancel each other except for the first and last term of the second and first summation, respectively.

$$\begin{aligned}\{R_n\} = {} & ((a_1 - a_{10})[M_o^\infty] + a_{10}[M_1^\infty])\{u_n\} + [M_o^\infty]\left(\{\dot{\bar{u}}_n\} + a_{10}\{u_0\} - \left(1-\frac{n}{N}\right)\{\dot{u}_0\}\right) + \\ & [M_1^\infty]\left(-a_{10}\{u_0\} - \frac{n}{N}\{\dot{u}_0\}\right)\end{aligned} \tag{2.162}$$

Second calculate $\{R_n\}$ in the interval from $t=(N+1)\Delta t$ to $t=MN\Delta t$:

M is the point at which the unit impulse function changes from non-linear to nearly linear.

Using equation (2.150), let $t_1=(i-1)\Delta t$, $t_2=i\Delta t$ and $t=n\Delta t$ where, $i=1\to n$ and $n\Delta t\in[(N+1)\Delta t, MN\Delta t]$

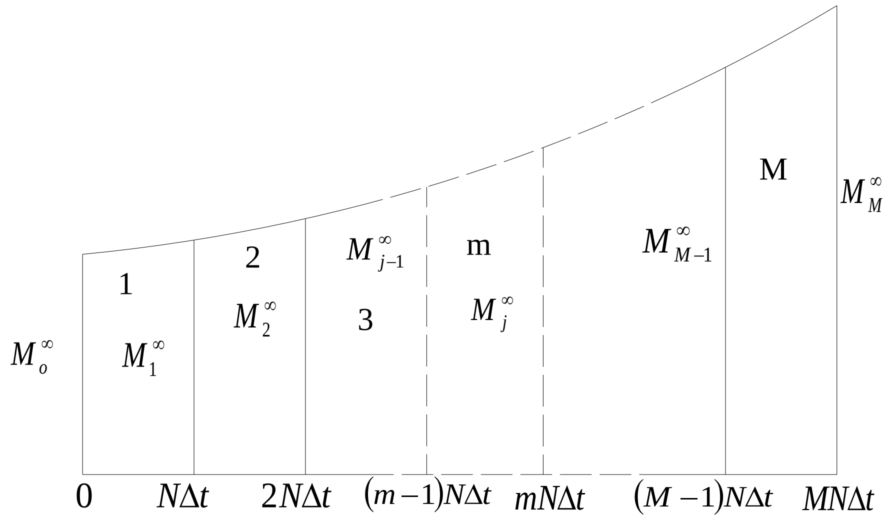
$$\{R_n\}=\{A\}+\{B\}+\{C\} \tag{2.163}$$

$$\{A\}=\sum_{i=1}^{i=N}\left(\begin{array}{l}-\left[M^{\infty}(i\Delta t)\right]\{\dot{u}_{n-i}\}+\left[M^{\infty}((i-1)\Delta t)\right]\{\dot{u}_{n-i+1}\}-\\ \left(\dfrac{\left[M^{\infty}(i\Delta t)\right]-\left[M^{\infty}((i-1)\Delta t)\right]}{\Delta t}\right)(\{u_{n-i}\}-\{u_{n-i+1}\})\end{array}\right) \tag{2.164}$$

$$\{B\}=\sum_{j=2}^{j=m}\sum_{i=(j-1)N+1}^{i=jN}\left(\begin{array}{l}-\left[M^{\infty}(i\Delta t)\right]\{\dot{u}_{n-i}\}+\left[M^{\infty}((i-1)\Delta t)\right]\{\dot{u}_{n-i+1}\}-\\ \left(\dfrac{\left[M^{\infty}(i\Delta t)\right]-\left[M^{\infty}((i-1)\Delta t)\right]}{\Delta t}\right)(\{u_{n-i}\}-\{u_{n-i+1}\})\end{array}\right) \tag{2.165}$$

$$\{C\}=\sum_{i=mN+1}^{i=n}\left(\begin{array}{l}-\left[M^{\infty}(i\Delta t)\right]\{\dot{u}_{n-i}\}+\left[M^{\infty}((i-1)\Delta t)\right]\{\dot{u}_{n-i+1}\}-\\ \left(\dfrac{\left[M^{\infty}(i\Delta t)\right]-\left[M^{\infty}((i-1)\Delta t)\right]}{\Delta t}\right)(\{u_{n-i}\}-\{u_{n-i+1}\})\end{array}\right) \tag{2.166}$$

Where, m is the number of time steps Δt`for the unit impulse function $[M^{\infty}(\tau)]$



From equation (2.152)

$$\left[M^{\infty}(i\Delta t)\right]=\left[M^{\infty}_{j-1}\right]+\left(\frac{\left[M^{\infty}_{j}\right]-\left[M^{\infty}_{j-1}\right]}{N}\right)\left(i-(j-1)N\right) \qquad (2.167)$$

$$\left[M^{\infty}((i-1)\Delta t)\right]=\left[M^{\infty}_{j-1}\right]+\left(\frac{\left[M^{\infty}_{j}\right]-\left[M^{\infty}_{j-1}\right]}{N}\right)\left((i-1)-(j-1)N\right) \qquad (2.168)$$

<u>First</u> calculate $\{A\}$:

Substitute by (2.167) and (2.168) in $\{A\}$ (put $j=1$).

Similar to equation (2.151)

$$\{A\}=\left((a_1-a_{10})\left[M^{\infty}_{o}\right]+a_{10}\left[M^{\infty}_{1}\right]\right)\{u_n\}+\left[M^{\infty}_{o}\right]\left(\{\dot{u}_n\}+a_{10}\{u_{n-N}\}\right)-\left[M^{\infty}_{1}\right]\left(\{\dot{u}_{n-N}\}+a_{10}\{u_{n-N}\}\right) \qquad (2.169)$$

<u>Second</u> calculate $\{B\}$:

Substitute by (2.167) and (2.168) in $\{B\}$

$$\{B\}=\sum_{j=2}^{j=m}\sum_{i=(j-1)N+1}^{i=jN}\left(\begin{array}{l}-\left(\left[M^{\infty}_{j-1}\right]+\left(\frac{\left[M^{\infty}_{j}\right]-\left[M^{\infty}_{j-1}\right]}{N}\right)\left(i-(j-1)N\right)\right)\{\dot{u}_{n-i}\}+\\ \left(\left[M^{\infty}_{j-1}\right]+\left(\frac{\left[M^{\infty}_{j}\right]-\left[M^{\infty}_{j-1}\right]}{N}\right)\left((i-1)-(j-1)N\right)\right)\{\dot{u}_{n-i+1}\}-\left(\frac{\left[M^{\infty}_{j}\right]-\left[M^{\infty}_{j-1}\right]}{N\Delta t}\right)\left(\{u_{n-i}\}-\{u_{n-i+1}\}\right)\end{array}\right)$$

$$\{B\}=\sum_{j=2}^{j=m}\left(\begin{array}{l}\sum_{i=(j-1)N+1}^{i=jN}\left(\begin{array}{l}\left[M^{\infty}_{j-1}\right]\left(\left(-1+\frac{i-(j-1)N}{N}\right)\{\dot{u}_{n-i}\}+a_{10}\{u_{n-i}\}\right)+\\ \left[M^{\infty}_{j}\right]\left(-\left(\frac{i-(j-1)N}{N}\right)\{\dot{u}_{n-i}\}-a_{10}\{u_{n-i}\}\right)\end{array}\right)+\\ \sum_{i=(j-1)N+1}^{i=jN}\left(\begin{array}{l}\left[M^{\infty}_{j-1}\right]\left(\left(1-\frac{(i-1)-(j-1)N}{N}\right)\{\dot{u}_{n-i+1}\}-a_{10}\{u_{n-i+1}\}\right)+\\ \left[M^{\infty}_{j}\right]\left(\left(\frac{(i-1)-(j-1)N}{N}\right)\{\dot{u}_{n-i+1}\}+a_{10}\{u_{n-i+1}\}\right)\end{array}\right)\end{array}\right)$$

Similar to that in equation (2.159)

$$\{B\}=\sum_{j=2}^{j=m}\left[M^{\infty}_{j}\right]\left(a_{10}\left(\{u_{n-(j-1)N}\}-\{u_{n-jN}\}\right)-\{\dot{u}_{n-jN}\}\right)-\left[M^{\infty}_{j-1}\right]\left(\begin{array}{l}a_{10}\left(\{u_{n-(j-1)N}\}-\{u_{n-jN}\}\right)\\ -\{\dot{u}_{n-(j-1)N}\}\end{array}\right) \qquad (2.170)$$

Third calculate $\{C\}$: (similar to $\{B\}$:

$$\{C\}=\left[M_m^\infty\right]\left(\{\dot{u}_{n-mN}\}-\left(1-\frac{k}{N}\right)\{\dot{u}_0\}-a_{10}\left(\{u_{n-mN}\}-\{u_0\}\right)\right)+ \left[M_{m+1}^\infty\right]\left(a_{10}\left(\{u_{n-mN}\}-\{u_0\}\right)-\frac{k}{N}\{\dot{u}_0\}\right) \tag{2.171}$$

Where, $k=n-mN$ (2.172)

Substitute by (2.169), (2.170) and (2.171) in (2.163)

$$\begin{aligned}\{R_n\}=&\left((a_1-a_{10})\left[M_o^\infty\right]+a_{10}\left[M_1^\infty\right]\right)\{u_n\}+\left[M_o^\infty\right]\left(\{\dot{u}_n\}+a_{10}\{u_{n-N}\}\right)-\\&\left[M_1^\infty\right]\left(\{\dot{u}_{n-N}\}+a_{10}\{u_{n-N}\}\right)+\sum_{j=2}^{j=m}\left(\begin{array}{l}\left[M_j^\infty\right]\left(a_{10}\left(\{u_{n-(j-1)N}\}-\{u_{n-jN}\}\right)-\{\dot{u}_{n-jN}\}\right)-\\\left[M_{j-1}^\infty\right]\left(a_{10}\left(\{u_{n-(j-1)N}\}-\{u_{n-jN}\}\right)-\{\dot{u}_{n-(j-1)N}\}\right)\end{array}\right)+\\&\left[M_m^\infty\right]\left(\{\dot{u}_{n-mN}\}-\left(1-\frac{k}{N}\right)\{\dot{u}_0\}-a_{10}\left(\{u_{n-mN}\}-\{u_0\}\right)\right)+\\&\left[M_{m+1}^\infty\right]\left(a_{10}\left(\{u_{n-mN}\}-\{u_0\}\right)-\frac{k}{N}\{\dot{u}_0\}\right)\end{aligned} \tag{2.173}$$

<u>Third calculate $\{R_n\}$ in the interval from $t=(MN+1)\Delta t$ to $t=n\Delta t$</u>:

Using equation (2.150), let $t_1=(i-1)\Delta t$, $t_2=i\Delta t$ and $t=n\Delta t$ where, $i=1\rightarrow n$ and $n\Delta t\in[(MN+1)\Delta t,t]$ (where t is any time $>MN\Delta t$)

$$\{R_n\}=\{A\}+\{F\}+\{H\} \tag{2.174}$$

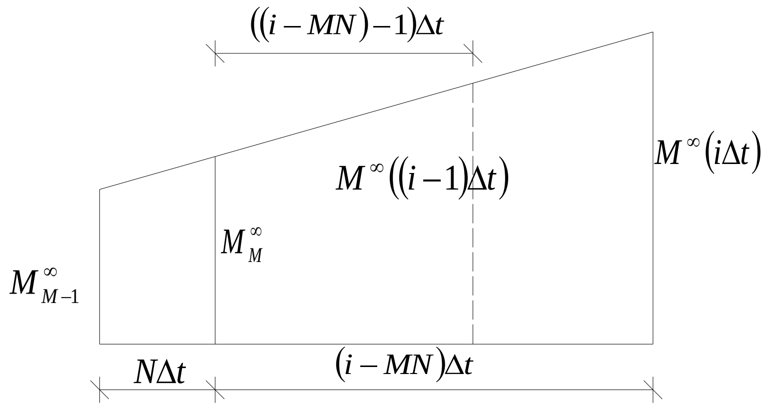
$$\{F\}=\sum_{j=2}^{j=M}\sum_{i=(j-1)N+1}^{i=jN}\left(\begin{array}{l}-\left[M^\infty(i\Delta t)\right]\{\dot{u}_{n-i}\}+\left[M^\infty((i-1)\Delta t)\right]\{\dot{u}_{n-i+1}\}-\\\left(\dfrac{\left[M^\infty(i\Delta t)\right]-\left[M^\infty((i-1)\Delta t)\right]}{\Delta t}\right)\left(\{u_{n-i}\}-\{u_{n-i+1}\}\right)\end{array}\right) \tag{2.175}$$

$$\{H\}=\sum_{i=MN+1}^{i=n}\left(\begin{array}{l}-\left[M^\infty(i\Delta t)\right]\{\dot{u}_{n-i}\}+\left[M^\infty((i-1)\Delta t)\right]\{\dot{u}_{n-i+1}\}-\\\left(\dfrac{\left[M^\infty(i\Delta t)\right]-\left[M^\infty((i-1)\Delta t)\right]}{\Delta t}\right)\left(\{u_{n-i}\}-\{u_{n-i+1}\}\right)\end{array}\right) \tag{2.176}$$

$\{A\}$ is calculated above equation (2.169).

Since, $\left[M^\infty(\tau)\right]$ is nearly linear after $t=MN\Delta t$

Therefore, all the values of $\left[M^\infty(\tau)\right]$ after $t=MN\Delta t$ can be written using extrapolation as follows:



$$\left[M^{\infty}(i\Delta t)\right]=\left[M^{\infty}_{M}\right]+\left(\frac{\left[M^{\infty}_{M}\right]-\left[M^{\infty}_{M-1}\right]}{N}\right)(i-MN) \tag{2.177}$$

$$\left[M^{\infty}((i-1)\Delta t)\right]=\left[M^{\infty}_{M}\right]+\left(\frac{\left[M^{\infty}_{M}\right]-\left[M^{\infty}_{M-1}\right]}{N}\right)((i-1)-MN) \tag{2.178}$$

<u>First</u> calculate $\{F\}$: (similar as $\{B\}$)

$$\{F\}=\sum_{j=2}^{j=M}\left[M^{\infty}_{j}\right]\left(a_{10}\left(\{u_{n-(j-1)N}\}-\{u_{n-jN}\}\right)-\{\dot{u}_{n-jN}\}\right)-\left[M^{\infty}_{j-1}\right]\begin{pmatrix}a_{10}\left(\{u_{n-(j-1)N}\}-\{u_{n-jN}\}\right)\\-\{\dot{u}_{n-(j-1)N}\}\end{pmatrix} \tag{2.179}$$

<u>Second</u> calculate $\{H\}$:

Substitute by (2.177) and (2.178) in (2.176)

$$\{H\}=\sum_{i=MN+1}^{i=n}\begin{pmatrix}-\left(\left[M^{\infty}_{M}\right]+\left(\frac{\left[M^{\infty}_{M}\right]-\left[M^{\infty}_{M-1}\right]}{N}\right)(i-MN)\right)\{\dot{u}_{n-i}\}+\\ \left(\left[M^{\infty}_{M}\right]+\left(\frac{\left[M^{\infty}_{M}\right]-\left[M^{\infty}_{M-1}\right]}{N}\right)((i-1)-MN)\right)\{\dot{u}_{n-i+1}\}-\left(\frac{\left[M^{\infty}_{M}\right]-\left[M^{\infty}_{M-1}\right]}{N\Delta t}\right)\left(\{u_{n-i}\}-\{u_{n-i+1}\}\right)\end{pmatrix}$$

As done in $\{B\}$ and using equation (2.161)

$$\{H\}=\left[M^{\infty}_{M-1}\right]\left(\frac{k}{N}\{\dot{u}_0\}-a_{10}\left(\{u_{n-MN}\}-\{u_0\}\right)\right)+\left[M^{\infty}_{M}\right]\begin{pmatrix}\{\dot{u}_{n-MN}\}-\left(1+\frac{k}{N}\right)\{\dot{u}_0\}+\\a_{10}\left(\{u_{n-MN}\}-\{u_0\}\right)\end{pmatrix} \tag{2.180}$$

Where, $k=n-MN$ (2.181)

Substitute by (2.169), (2.179) and (2.180) in (2.174)

$$
\begin{aligned}
\{R_n\} = & \left((a_1 - a_{10})\left[M_{\circ}^{\infty}\right] + a_{10}\left[M_1^{\infty}\right]\right)\{u_n\} + \left[M_{\circ}^{\infty}\right]\left(\{\dot{\bar{u}}_n\} + a_{10}\{u_{n-N}\}\right) - \\
& \left[M_1^{\infty}\right]\left(\{\dot{u}_{n-N}\} + a_{10}\{u_{n-N}\}\right) + \sum_{j=2}^{j=M}\begin{pmatrix}\left[M_j^{\infty}\right]\left(a_{10}\left(\{u_{n-(j-1)N}\} - \{u_{n-jN}\}\right) - \{\dot{u}_{n-jN}\}\right) - \\ \left[M_{j-1}^{\infty}\right]\left(a_{10}\left(\{u_{n-(j-1)N}\} - \{u_{n-jN}\}\right) - \{\dot{u}_{n-(j-1)N}\}\right)\end{pmatrix} + \\
& \left[M_{M-1}^{\infty}\right]\left(\frac{k}{N}\{\dot{u}_0\} - a_{10}\left(\{u_{n-MN}\} - \{u_0\}\right)\right) + \left[M_M^{\infty}\right]\begin{pmatrix}\{\dot{u}_{n-MN}\} - \left(1 + \dfrac{k}{N}\right)\{\dot{u}_0\} + \\ a_{10}\left(\{u_{n-MN}\} - \{u_0\}\right)\end{pmatrix}
\end{aligned}
\tag{2.182}
$$

Therefore, using equations (2.162), (2.173) and (2.182) in the intervals $[0, N\Delta t]$, $[(N+1)\Delta t, MN\Delta t]$ and $[(MN+1)\Delta t, t]$, respectively.

If we have interaction between a structure and the unbounded domain, we are going to use equation (1.1), we are going to write it here in a detailed form:

$$
\begin{aligned}
& \begin{bmatrix}[M_{bb}] & [M_{bs}] \\ [M_{sb}] & [M_{ss}]\end{bmatrix}\begin{Bmatrix}\{\ddot{u}_b(t)\} \\ \{\ddot{u}_s(t)\}\end{Bmatrix} + \begin{bmatrix}[C_{bb}] & [C_{bs}] \\ [C_{sb}] & [C_{ss}]\end{bmatrix}\begin{Bmatrix}\{\dot{u}_b(t)\} \\ \{\dot{u}_s(t)\}\end{Bmatrix} + \begin{bmatrix}[K_{bb}] & [K_{bs}] \\ [K_{sb}] & [K_{ss}]\end{bmatrix}\begin{Bmatrix}\{u_b(t)\} \\ \{u_s(t)\}\end{Bmatrix} \\
& = \begin{Bmatrix}\{P_b(t)\} \\ \{P_x(t)\}\end{Bmatrix} - \begin{Bmatrix}\{R_b(t)\} \\ 0\end{Bmatrix}
\end{aligned}
\tag{2.183}
$$

The mass, damping and stiffness matrices are calculated from the properties of the structure using finite element method (subscript *b* and *s* stands for nodes on the interface and on the structure, respectively)

Using Newmark time scheme to solve the above second order differential equation, so substitute by equations (2.157) and (2.158) in (2.183)

$$
\begin{aligned}
& \left(\frac{1}{\beta\Delta t^2}\begin{bmatrix}[M_{bb}] & [M_{bs}] \\ [M_{sb}] & [M_{ss}]\end{bmatrix} + a_1\begin{bmatrix}[C_{bb}] & [C_{bs}] \\ [C_{sb}] & [C_{ss}]\end{bmatrix} + \begin{bmatrix}[K_{bb}] & [K_{bs}] \\ [K_{sb}] & [K_{ss}]\end{bmatrix}\right)\begin{Bmatrix}\{u_{bn}\} \\ \{u_{sn}\}\end{Bmatrix} = \\
& \left(\frac{1}{\beta\Delta t^2}\begin{bmatrix}[M_{bb}] & [M_{bs}] \\ [M_{sb}] & [M_{ss}]\end{bmatrix} + a_1\begin{bmatrix}[C_{bb}] & [C_{bs}] \\ [C_{sb}] & [C_{ss}]\end{bmatrix}\right)\begin{Bmatrix}\{u_{b(n-1)}\} \\ \{u_{s(n-1)}\}\end{Bmatrix} + \\
& \left(\frac{1}{\beta\Delta t}\begin{bmatrix}[M_{bb}] & [M_{bs}] \\ [M_{sb}] & [M_{ss}]\end{bmatrix} + a_4\begin{bmatrix}[C_{bb}] & [C_{bs}] \\ [C_{sb}] & [C_{ss}]\end{bmatrix}\right)\begin{Bmatrix}\{\dot{u}_{b(n-1)}\} \\ \{\dot{u}_{s(n-1)}\}\end{Bmatrix} + \\
& \left(\frac{0.5-\beta}{\beta}\begin{bmatrix}[M_{bb}] & [M_{bs}] \\ [M_{sb}] & [M_{ss}]\end{bmatrix} + a_5\begin{bmatrix}[C_{bb}] & [C_{bs}] \\ [C_{sb}] & [C_{ss}]\end{bmatrix}\right)\begin{Bmatrix}\{\ddot{u}_{b(n-1)}\} \\ \{\ddot{u}_{s(n-1)}\}\end{Bmatrix} + \begin{Bmatrix}\{P_{bn}\} \\ \{P_{xn}\}\end{Bmatrix} - \begin{Bmatrix}0 \\ \{R_{bn}\}\end{Bmatrix}
\end{aligned}
\tag{2.184}
$$

From equations (2.162), (2.173) and (2.182) $\{R_{bn}\}$ can be written as follows

$\{R_{bn}\} = \left((a_1 - a_{10})\left[M_{\circ}^{\infty}\right] + a_{10}\left[M_1^{\infty}\right]\right)\{u_{bn}\} + \{q_{bn}\}$, where, $\{q_{bn}\}$ is the different term in the three equations (2.185)

Substitute by (2.185) in (2.184) and rearrange we get:

$$\begin{bmatrix}[S_{bb}] & [S_{bs}] \\ [S_{sb}] & [S_{ss}]\end{bmatrix}\begin{Bmatrix}\{u_{bn}\} \\ \{u_{sn}\}\end{Bmatrix} = \begin{bmatrix}[Y_{bb}] & [Y_{bs}] \\ [Y_{sb}] & [Y_{ss}]\end{bmatrix}\begin{Bmatrix}\{u_{b(n-1)}\} \\ \{u_{s(n-1)}\}\end{Bmatrix} + \begin{bmatrix}[L_{bb}] & [L_{bs}] \\ [L_{sb}] & [L_{ss}]\end{bmatrix}\begin{Bmatrix}\{\dot{u}_{b(n-1)}\} \\ \{\dot{u}_{s(n-1)}\}\end{Bmatrix} + \begin{bmatrix}[X_{bb}] & [X_{bs}] \\ [X_{sb}] & [X_{ss}]\end{bmatrix}\begin{Bmatrix}\{\ddot{u}_{b(n-1)}\} \\ \{\ddot{u}_{s(n-1)}\}\end{Bmatrix} + \begin{Bmatrix}\{P_{bn}\} \\ \{P_{sn}\}\end{Bmatrix} - \begin{Bmatrix}0 \\ \{q_{bn}\}\end{Bmatrix} \tag{2.186}$$

where,

$$\begin{bmatrix}[S_{bb}] & [S_{bs}] \\ [S_{sb}] & [S_{ss}]\end{bmatrix} = \left(\begin{matrix}\frac{1}{\beta\Delta t^2}\begin{bmatrix}[M_{bb}] & [M_{bs}] \\ [M_{sb}] & [M_{ss}]\end{bmatrix} + a_1\begin{bmatrix}[C_{bb}] & [C_{bs}] \\ [C_{sb}] & [C_{ss}]\end{bmatrix} + \begin{bmatrix}[K_{bb}] & [K_{bs}] \\ [K_{sb}] & [K_{ss}]\end{bmatrix} + \\ \begin{bmatrix}\left((a_1 - a_{10})[M_o^\infty] + a_{10}[M_1^\infty]\right) & [0] \\ [0] & [0]\end{bmatrix}\end{matrix}\right) \tag{2.187}$$

$$\begin{bmatrix}[Y_{bb}] & [Y_{bs}] \\ [Y_{sb}] & [Y_{ss}]\end{bmatrix} = \left(\frac{1}{\beta\Delta t^2}\begin{bmatrix}[M_{bb}] & [M_{bs}] \\ [M_{sb}] & [M_{ss}]\end{bmatrix} + a_1\begin{bmatrix}[C_{bb}] & [C_{bs}] \\ [C_{sb}] & [C_{ss}]\end{bmatrix}\right) \tag{2.188}$$

$$\begin{bmatrix}[L_{bb}] & [L_{bs}] \\ [L_{sb}] & [L_{ss}]\end{bmatrix} = \left(\frac{1}{\beta\Delta t}\begin{bmatrix}[M_{bb}] & [M_{bs}] \\ [M_{sb}] & [M_{ss}]\end{bmatrix} + a_4\begin{bmatrix}[C_{bb}] & [C_{bs}] \\ [C_{sb}] & [C_{ss}]\end{bmatrix}\right) \tag{2.189}$$

$$\begin{bmatrix}[X_{bb}] & [X_{bs}] \\ [X_{sb}] & [X_{ss}]\end{bmatrix} = \left(\frac{0.5-\beta}{\beta}\begin{bmatrix}[M_{bb}] & [M_{bs}] \\ [M_{sb}] & [M_{ss}]\end{bmatrix} + a_5\begin{bmatrix}[C_{bb}] & [C_{bs}] \\ [C_{sb}] & [C_{ss}]\end{bmatrix}\right) \tag{2.190}$$

From (2.162), (2.173) and (2.182):

In the interval $[\Delta t, N\Delta t]$

$$\{q_{bn}\} = [M_o^\infty]\left(\{\dot{\bar{u}}_n\} + a_{10}\{u_0\} - \left(1 - \frac{n}{N}\right)\{\dot{u}_0\}\right) + [M_1^\infty]\left(-a_{10}\{u_0\} - \frac{n}{N}\{\dot{u}_0\}\right) \tag{2.191}$$

In the interval $[(N+1)\Delta t, MN\Delta t]$

$$\begin{aligned}\{q_{bn}\} = {} & [M_o^\infty](\{\dot{\bar{u}}_n\} + a_{10}\{u_{n-N}\}) - [M_1^\infty](\{\dot{u}_{n-N}\} + a_{10}\{u_{n-N}\}) + \\ & \sum_{j=2}^{j=m}\left(\begin{matrix}[M_j^\infty](a_{10}(\{u_{n-(j-1)N}\} - \{u_{n-jN}\}) - \{\dot{u}_{n-jN}\}) - \\ [M_{j-1}^\infty](a_{10}(\{u_{n-(j-1)N}\} - \{u_{n-jN}\}) - \{\dot{u}_{n-(j-1)N}\})\end{matrix}\right) + \\ & [M_m^\infty]\left(\{\dot{u}_{n-mN}\} - \left(1 - \frac{k}{N}\right)\{\dot{u}_0\} - a_{10}(\{u_{n-mN}\} - \{u_0\})\right) + \\ & [M_{m+1}^\infty]\left(a_{10}(\{u_{n-mN}\} - \{u_0\}) - \frac{k}{N}\{\dot{u}_0\}\right)\end{aligned} \tag{2.192}$$

$$k = n - mN \tag{2.193}$$

$$\{q_{bn}\} = [M_o^\infty](\{\dot{\bar{u}}_n\} + a_{10}\{u_{n-N}\}) - [M_1^\infty](\{\dot{u}_{n-N}\} + a_{10}\{u_{n-N}\}) +$$

$$\sum_{j=2}^{j=M}\begin{pmatrix} [M_j^\infty](a_{10}(\{u_{n-(j-1)N}\} - \{u_{n-jN}\}) - \{\dot{u}_{n-jN}\}) - \\ [M_{j-1}^\infty](a_{10}(\{u_{n-(j-1)N}\} - \{u_{n-jN}\}) - \{\dot{u}_{n-(j-1)N}\}) \end{pmatrix} +$$

In the interval $[(MN+1)\Delta t, t]$

$$[M_{M-1}^\infty]\left(\frac{k}{N}\{\dot{u}_0\} - a_{10}(\{u_{n-MN}\} - \{u_0\})\right) + \tag{2.194}$$

$$[M_M^\infty]\begin{pmatrix} \{\dot{u}_{n-MN}\} - \left(1 + \frac{k}{N}\right)\{\dot{u}_0\} + \\ a_{10}(\{u_{n-MN}\} - \{u_0\}) \end{pmatrix}$$

$$k = n - MN \tag{2.195}$$

In all the examples present in this thesis the damping term is not taken in consideration.

Chapter 3: Computer Program

3.1. Introduction

In this chapter the program made by the author is discussed. The program here solves the unbounded domain using SBFEM as discussed in the previous chapter for 2D (in-plane and out of plane); also it solves problems of coupled FE/SBFE in 2D problems. The program is written using MATLAB program ver.7.6. A program flow chart is shown; the input files and the main program functions are discussed, after it.

3.2. Program flow chart

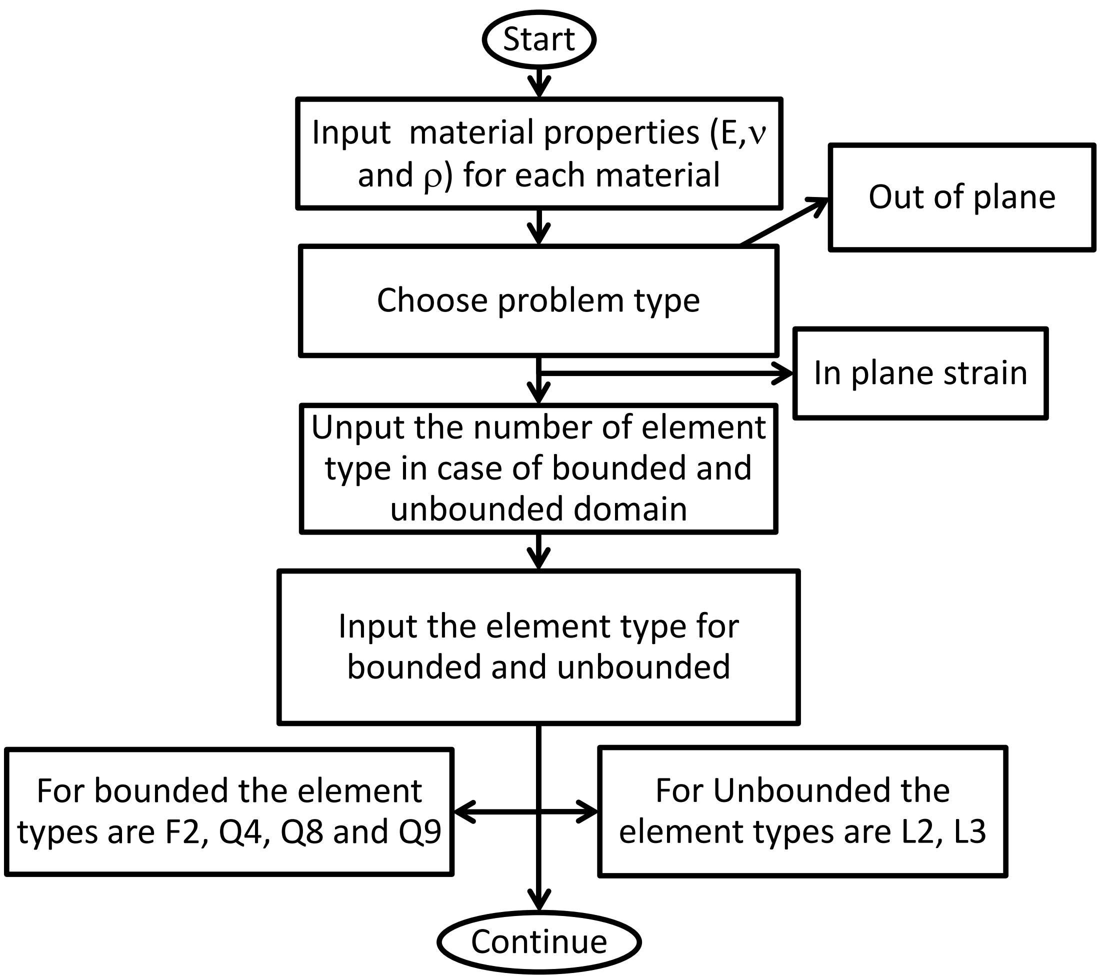


Fig.(3.1): Main program flow chart

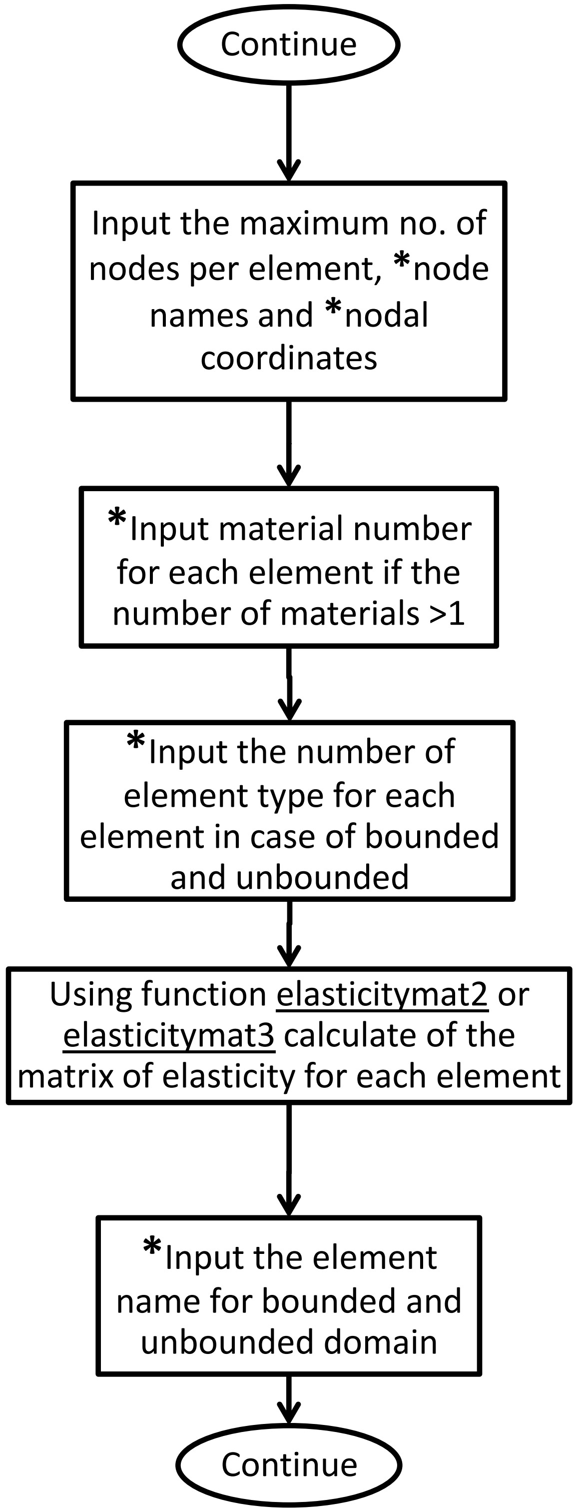


Fig.(3.2): Main program flow chart (Continue)

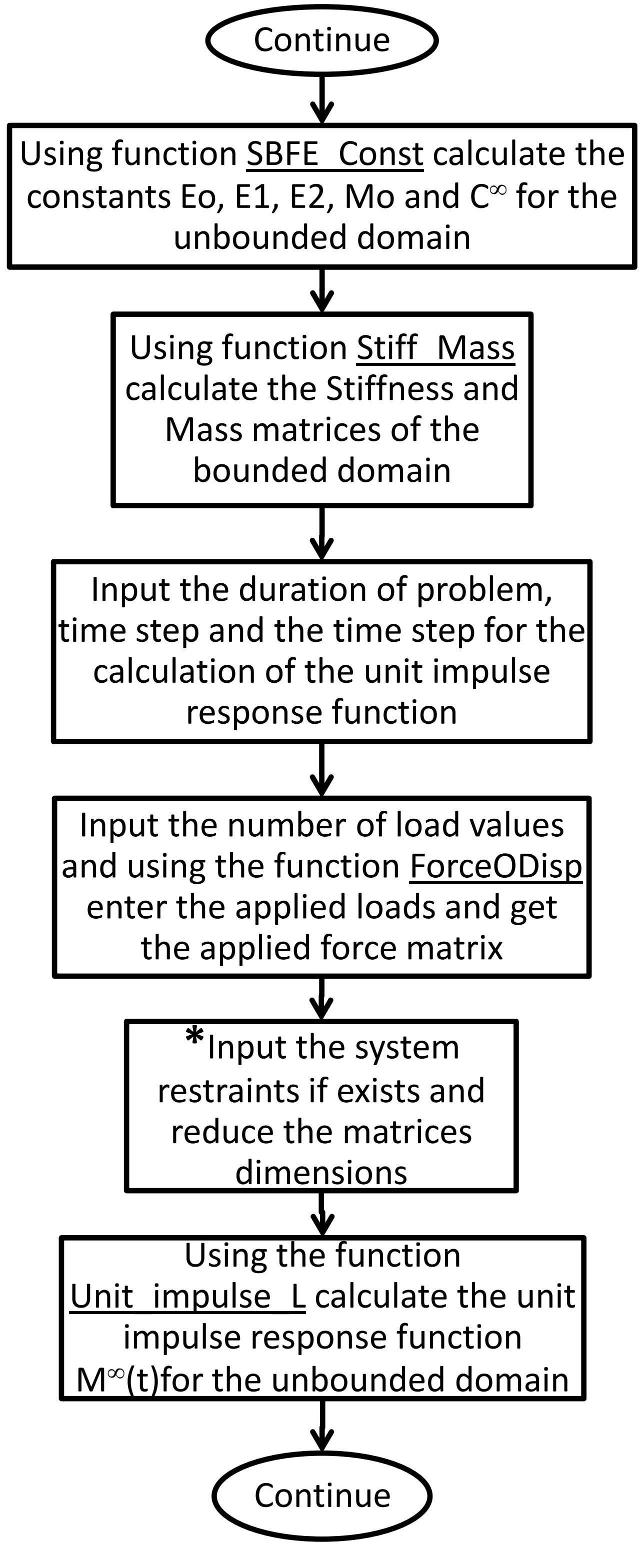


Fig.(3.3): Main program flow chart (Continue)

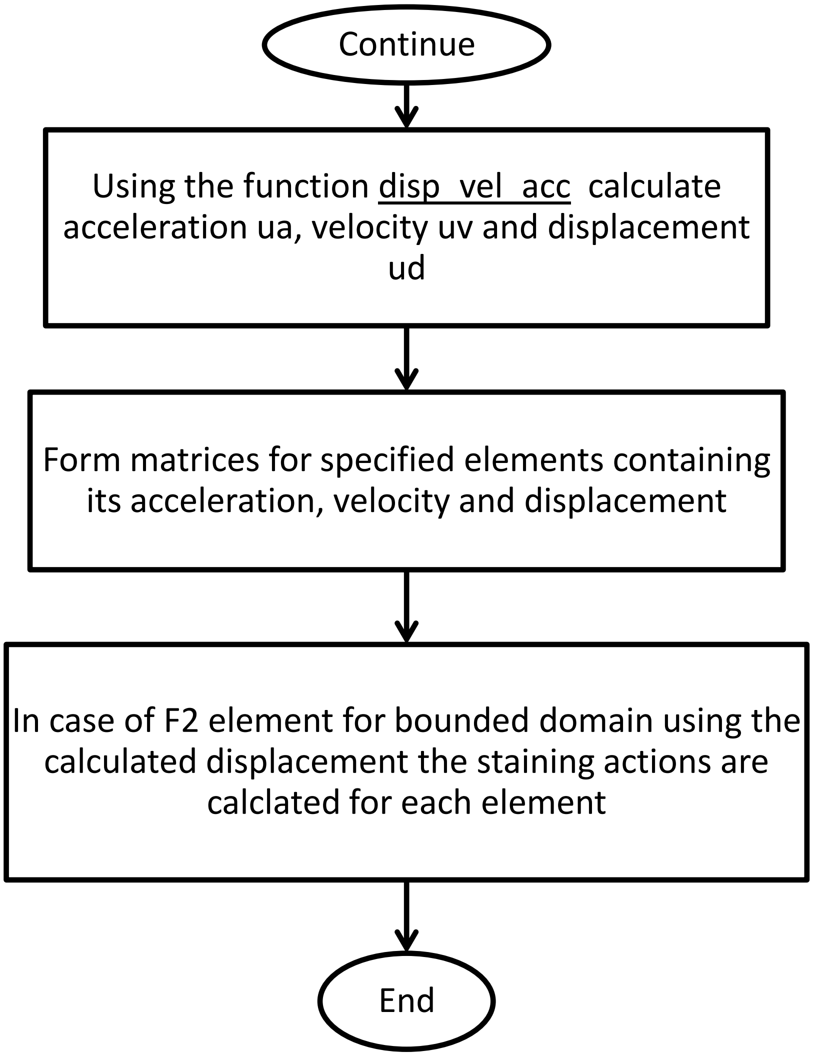


Fig.(3.4): Main program flow chart (Continue)

3.3. Main program

In the program there are data entered as input files and other entered directly by the user. The data entered from input files are marked above in the flow chart by astrex (*****). The input files in the main program are: *nodename, Coordinates*, *material*, *type_ele_unb*, *type_ele_b, elem_boun*, *elem_unboun, restraints*, *ele_sa_plane* and *ele_sa_frame.*

Input files	Description
nodename	A matrix with number of columns equal the number of elements and in each column is the name of nodes ordered in anti-clockwise (The elements of the unbounded domain SBFE is inserted first then the elements of the bounded FE, the SBFE nodes are entered in anti-clock wise with respect to the scaling center, but the FE quadrilateral element are with respect to the normal to the plane, also the names of the nodes of the SBFEs are numbered first)
Coordinates	A matrix with number of columns equals 2 for 2D and 3 for 3D each columns represent a direction ordered X, Y for 2D (except for axi-symmetric R, Z) and X, Y, Z for 3D, where it contains the coordinates of each node in the system
material	A row vector with number of columns equals the number of elements in the system where in each column the number of the material as entered in the beginning of the program
type_ele_unb	A row vector containing numbers that corresponds to element type as assigned for each element for unbounded
type_ele_b	A row vector containing numbers that corresponds to element type as assigned for each element for bounded
elem_boun	A row vector containing the names of elements in bounded domain
elem_unboun	A row vector containing the names of elements in unbounded domain
Restraints	A row vector containing the restraints of the system as 1 for restraint direction and 0 for unrestraint direction (the number of columns is the number of nodes*degrees of freedom (Dofs) for each node). The Dofs are ordered as X, Y for 2D (except for axi-symmetric R, Z, θ, and in case frame is present the Dofs are X, Y, θ) and X, Y, Z for 3D
ele_sa_plane	A row vector containing the name of plane elements needed to get the internal force in them
ele_sa_frame	A row vector containing the name of frame elements needed to get the

	internal force in them

Element L2 and L3 are line elements with 2 and 3 nodes, respectively; each has 2 Dofs (translation). Element Q4, Q8 and Q9 are quadrilateral isoparametric elements with 4, 8 and 9 nodes, respectively, each has 2Dofs (translation). Element F2 is a frame element with 2 nodes, each has 3 Dofs (two translations and one rotation).

It should be noted that in case of frame element presence the Dofs = number of nodes*2 + number of nodes joining the frame elements, also the dimensions of matrices E_o, E_1, E_2, Mo and the unit impulse response matrix is calculated using the Dof of the unbounded domain only, i.e. the number of rows or columns = number of nodes on the structure domain interface* spatial dimension.

3.3.1.Function-*elasticitymat2* and function *elasticitymat3*

In this function the properties of each material (Young`s modulus E and Poisson`s ratio ν) are entered and the elasticity matrix is calculated.

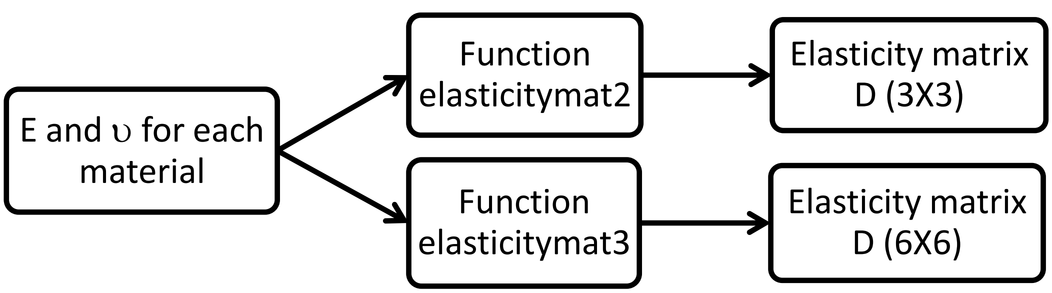


Fig.(3.5): elasticitymat2 and elasticitymat3 function flow chart

3.3.2.Function-*SBFE_Const*

In this function the constants used in the SBFE equation (E_o, E_1, E_2 and M_o) are calculated according to equations (2.48, 2.49, 2.50 and 2.51, respectively) also the initial value of the unit impulse response function ($C_{\infty} = M_{o}^{\infty}$) is calculated (equation (2.96)).

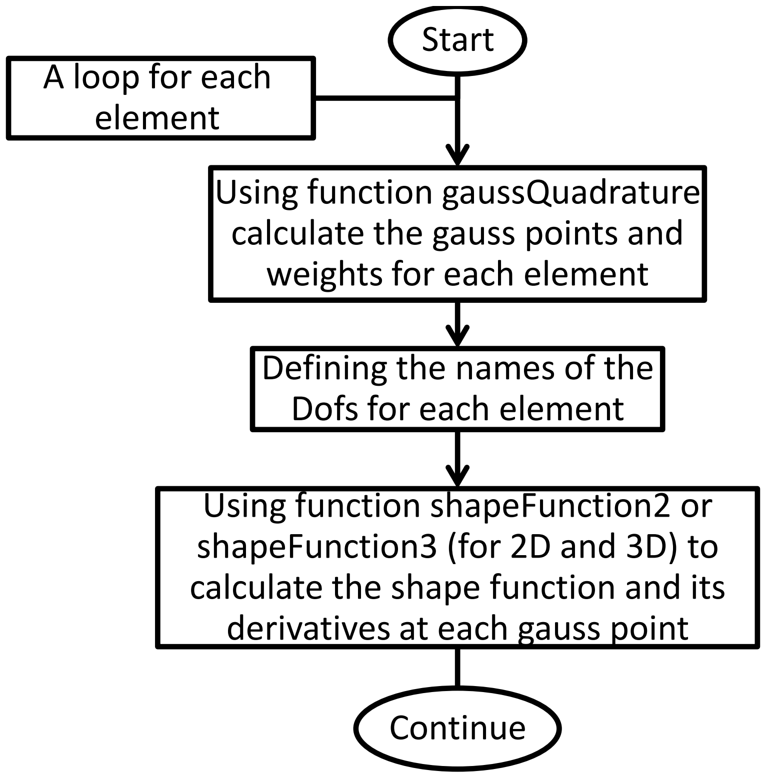


Fig.(3.6): SBFE_Const function flow chart

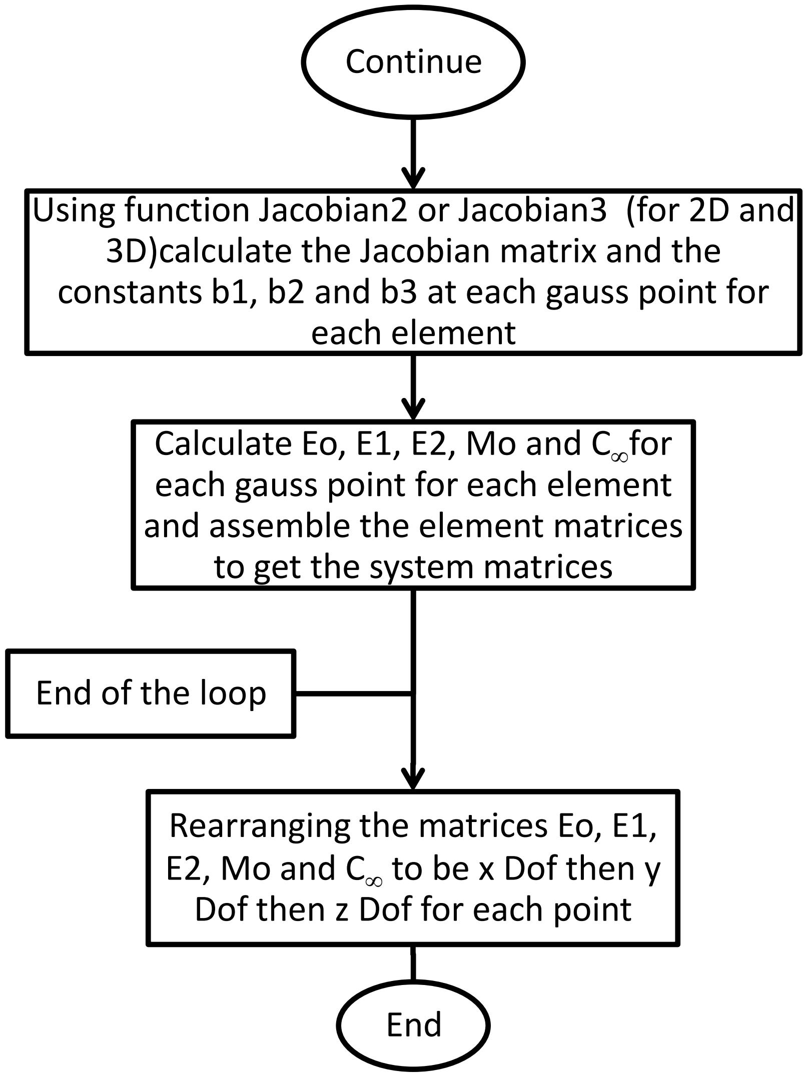


Fig.(3.7): SBFE_Const function flow chart (Continue)

3.3.3.Function-*Stiff_Mass*

In this function the stiffness and mass matrices for the FE part of the model are calculated. (For these matrices see [3] chapters 9 and 11, respectively)

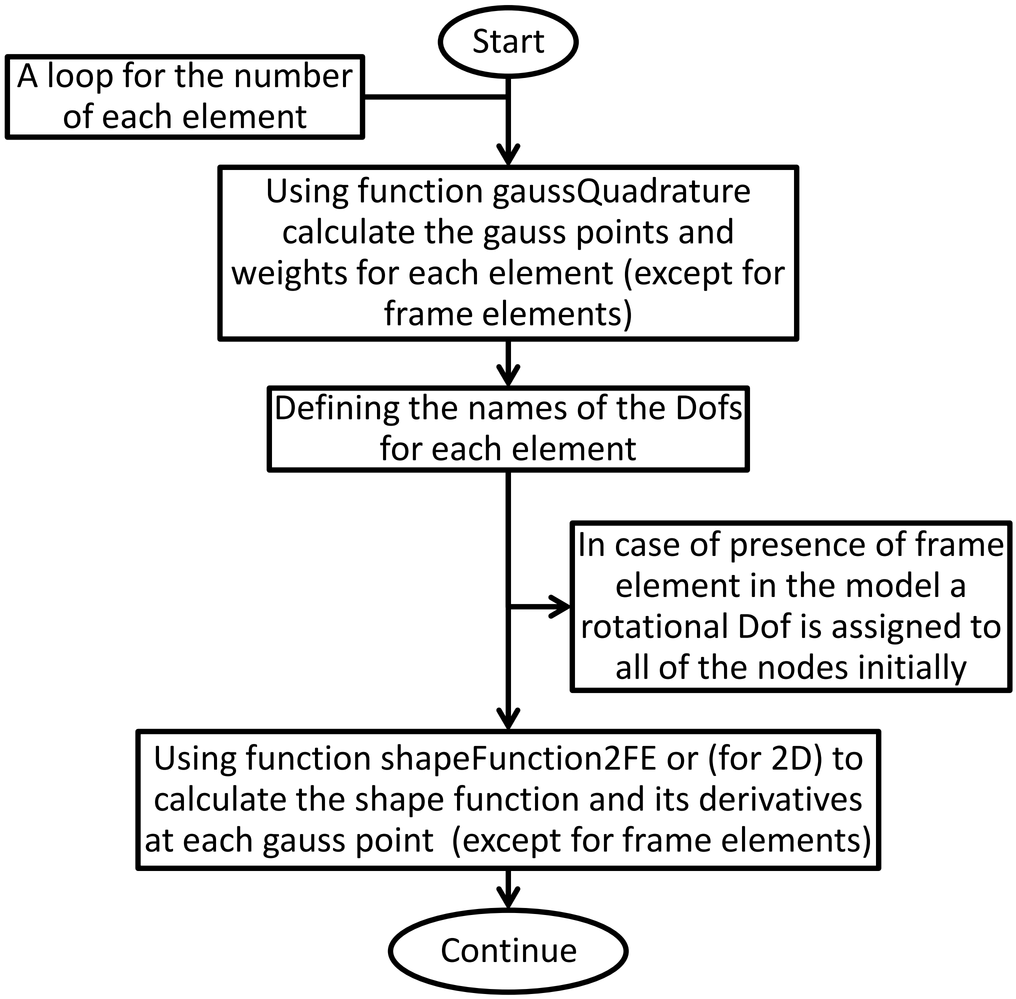


Fig.(3.8): Stiff_Mass function flow chart

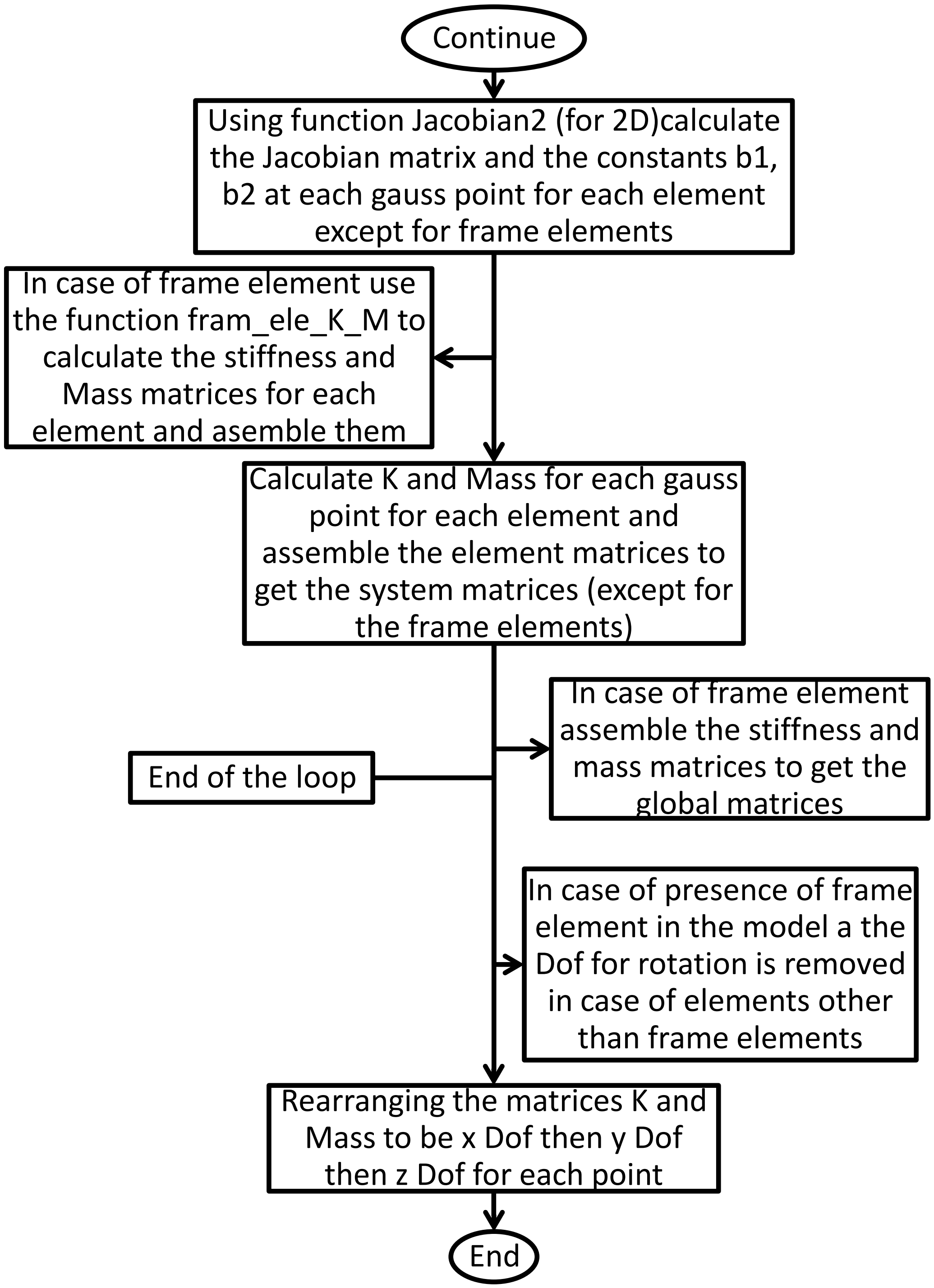


Fig.(3.9): Stiff_Mass function flow chart (Continue)

In this function two input files are needed in case of presence of frame element in the model line_bre and line_depth in which the breadth and depth of the frame elements are entered in row vector.

3.3.3.1. Function-*fram_ele_K_M*

In this function the stiffness and mass matrices of each frame element are calculated. (For these matrices see [3] chapters 9 and 11, respectively)

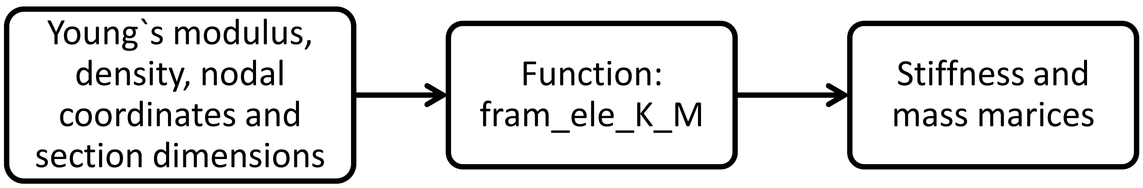


Fig.(3.10): Fram_ele_K_M function flow chart

3.3.4.Function-*ForceODisp*

In this function the applied force is inserted in a global vector for each time for each Dof.

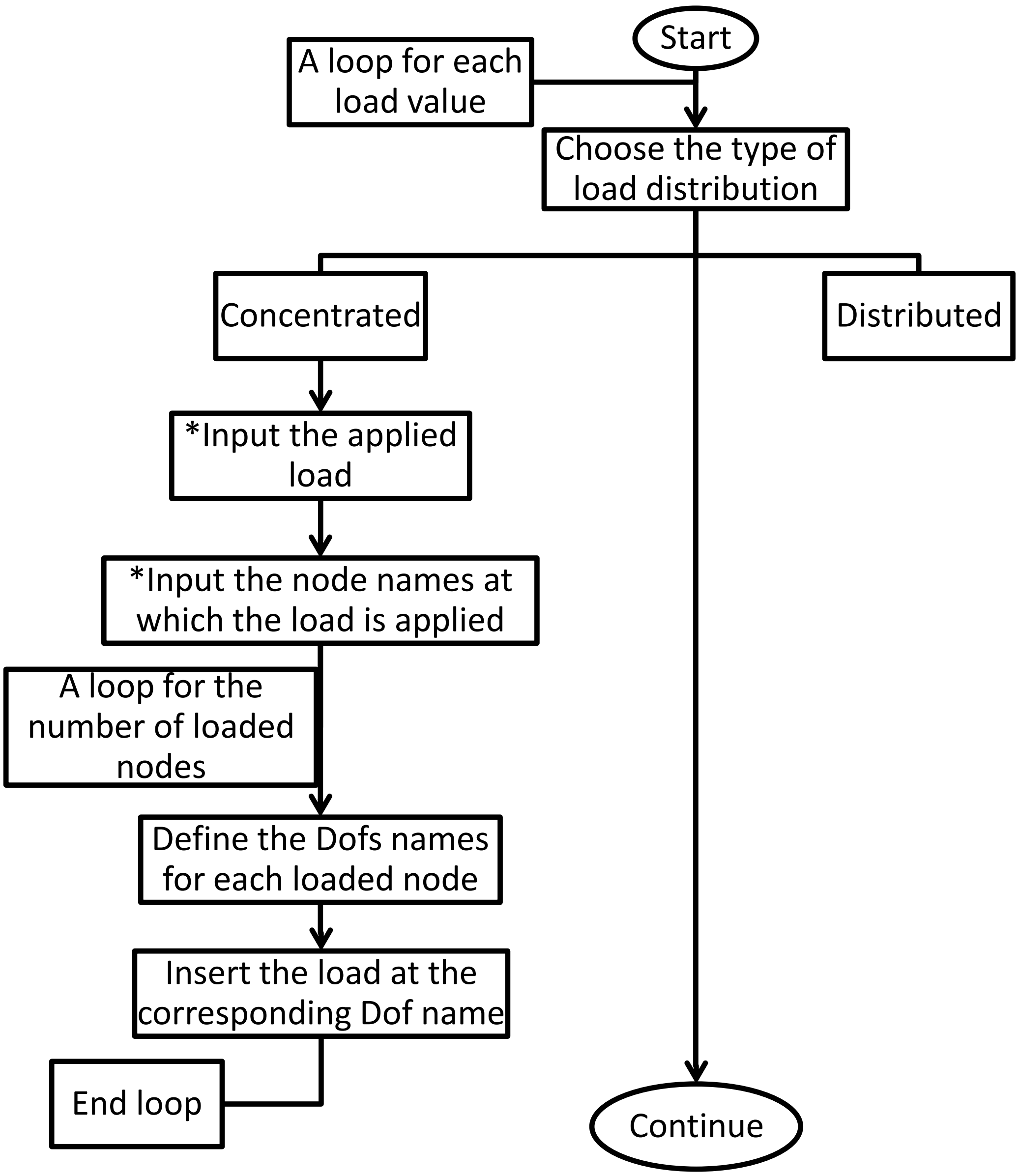


Fig.(3.11): ForceODisp function flow chart

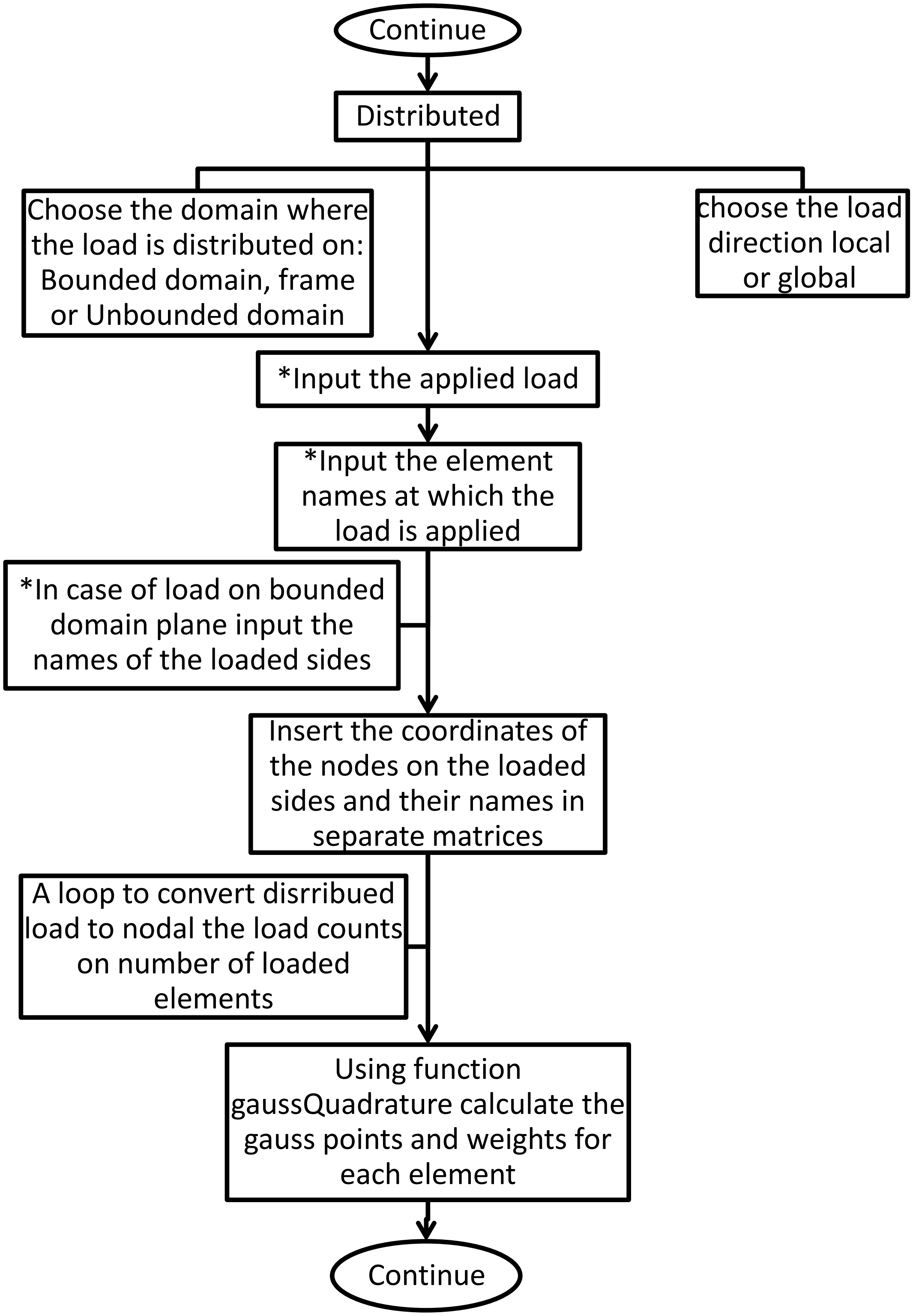


Fig.(3.12): ForceODisp function flow chart (Continue)

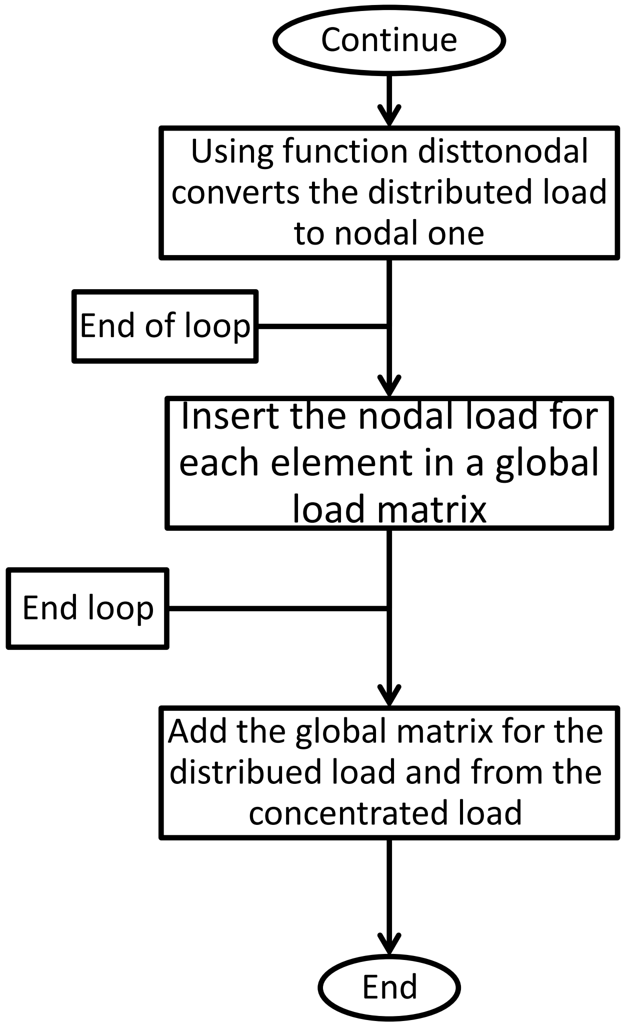


Fig.(3.13): ForceODisp function flow chart (Continue)

In this function there are four input files needed *Applied_load*, *App_nod*, *App_ele* and *sidename*, they are referred to in the above flow chart by astreix (*).

Input files	Description
Applied_load	The load is entered in local or global coordinates, the input is a matrix with number of rows equal to the spatial dimensions (2 or 3) and number of columns equal to the time steps+1. The rows in global direction

	represents X,Y in 2D and X,Y and θ in 2D for frame element concentrated loads, and X,Y,Z in 3D. In case of local it will be ξ,η,ζ. (It should be noted that in local case the +ve of ξ direction is toward the unbounded domain)
App_nod	The name of nodes at which the load is applied is entered in the form of row.
App_ele	The name of elements at which the load is applied is entered in the form of row.
sidename	A files containing the names of the sides at which the distributed load is applied, each column represents a side of an element at which the load is applied, the names of the nodes on the side is entered in the rows of each column

3.3.4.1. Function-*disttonodal*

In this function the distributed load is converted to nodal one.

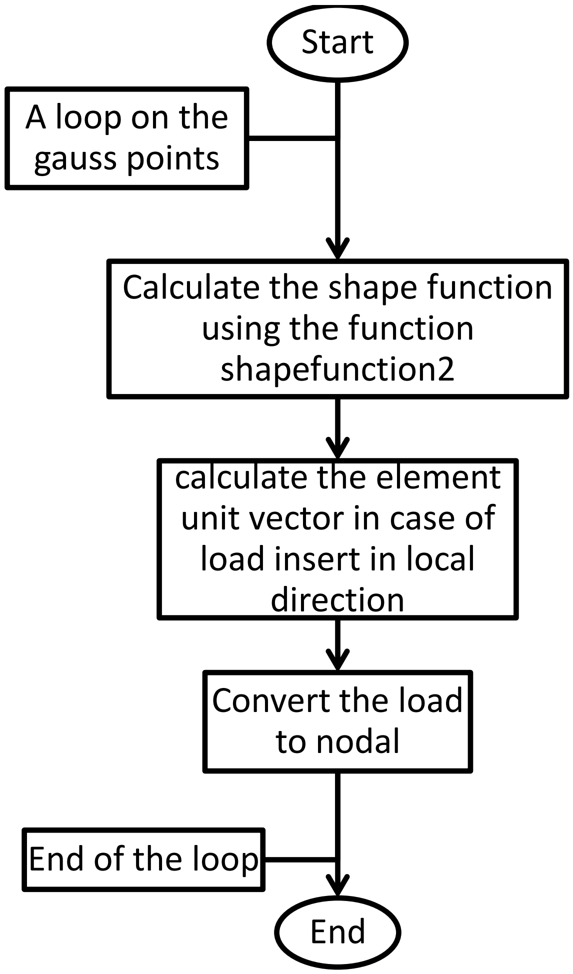


Fig.(3.14): disttonodal function flow chart

3.3.5.Function-*Unit_impulse_L*

In this function the unit impulse response function is calculated as described in chapter 2

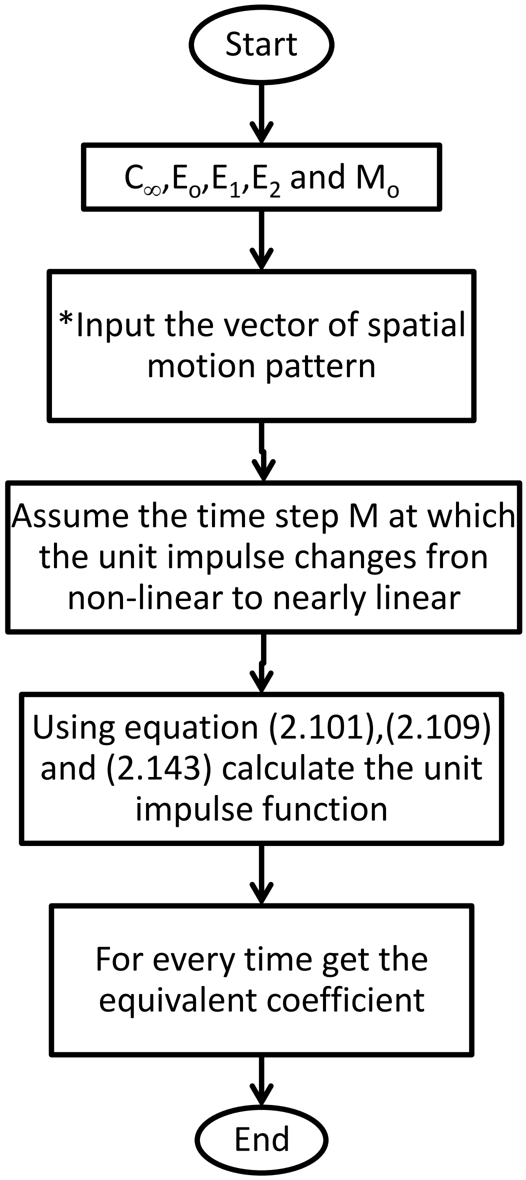


Fig.(3.15): Unit_impulse_L function flow chart

In this function there is one input file which is spatial_motion_pattern which is a column vector containing the position vector in the direction at which the load is applied to the unbounded domain. Here is the MATLAB code for this function

```
function [MI,MMI,M]=Unit_impulse_L(CI,Eo,E1,E2,Mo,nt,dt,s,Dof,th,xxi)
% A function to calculate the acceleration unit impulse response
function
% assuming the function as linear during each time interval
clear U;
U=chol(Eo); % Cholesky decomposition
e1=inv(U')*E1*inv(U);
e2=inv(U')*E2*inv(U);
mo=inv(U')*Mo*inv(U);
p2=(dt)*(e1-(s+1)*0.5*eye(Dof));
p3=((dt^2)/6)*((e1*e1')-e2);
% MI : unit impulse response function for acceleration
% mi : inv(U')*MI*inv(U)
% sm : integration of mi with respect to time at the n-1 step
%      =dt*mi (equ(67a))
% ssm : double integration of mi with respect to time
%       =dt*sm(n-1)+ssm(n-1) (equ(67b))
% smm : convolution integral (equ(68))
% MMI : total unit impulse response function of the medium
%Initializing
clear MI;clear mi;clear smm;clear sm;clear ssm;clear S;clear MMI;clear
phi;
disp('Loading the spatial motion pattern....')
%phi=xlsread('coordinates.xlsx',-1);
phi=load('spatial_motion_pattern.txt');
disp('M must be 3=<M<=total time steps')
M=input('Enter the assumed truncation time before nearly linear
behavour: ');
if M<=floor(nt*(1/dt))
    ttsteps=M;
else
    ttsteps=floor(nt*(1/dt));
    M=ttsteps;
end
disp('Initializing the matrices....')
MMI=zeros(1,ttsteps+2);
MI=zeros(Dof,Dof,ttsteps+1);
mi=zeros(Dof,Dof,ttsteps+1);
smm=zeros(Dof,Dof,ttsteps+1);
sm=zeros(Dof,Dof,ttsteps+1);
ssm=zeros(Dof,Dof,ttsteps+1);
MI(:,:,1)=CI;
mi(:,:,1)=inv(U')*MI(:,:,1)*inv(U);
MMI(1)=phi'*MI(:,:,1)*phi;
%Subistitute in the SBFE equation in time domain for unit impulse
%First time step t=dt
disp('Calculating unit impulse at t=N*dt....')
clear B1;clear C1;clear I1;
B1=-(xxi^(s-2))*((xxi^(2-
s))*2*mi(:,:,1)+th*p2+1.5*th*dt*eye(Dof));I1=eye(Dof);
C1=-(xxi^(s-2))*((xxi^(2-
s))*mi(:,:,1)*mi(:,:,1)+3*th*dt*mi(:,:,1)+2*th*p2*mi(:,:,1)+2*th*mi(:,:,
1)*p2'+(xxi^(s-2))*6*(th^2)*p3-(xxi^s)*6*mo);
% Solving Riccati algebraic equation equ(69)
```

```
[mi(:,:,2)]=care(B1',I1,C1);
mi(:,:,2)=(th-1)*(1/th)*mi(:,:,1)+(1/th)*mi(:,:,2);
MI(:,:,2)=U'*mi(:,:,2)*U;
MMI(2)=phi'*MI(:,:,2)*phi;
%Second time step t=2*dt
disp('Calculating unit impulse at t=2*N*dt....')
clear B2;clear C2;clear I2;clear a;clear b;clear c;clear f;
B2=-(xxi^(s-2))*((xxi^(2-s))*(3*(th^2)-th)*mi(:,:,1)+(xxi^(2-
s))*(2*(th^3)-
2*th+1)*mi(:,:,2)+(th^4)*p2+1.5*(th^3)*(th+1)*dt*eye(Dof));I2=sqrt((th-
1)^3)*eye(Dof);
a=p2*((3*(th^3)+2*(th^2))*mi(:,:,1)+(2*(th^4)+3*(th^3)+(th^2))*mi(:,:,2)
);
b=(th^2)*(th+1)*(3*dt*mi(:,:,1)+3*dt*(th+1)*mi(:,:,2)+(xxi^(s-
2))*((th+1)^2)*6*p3-(xxi^s)*6*mo);
c=(xxi^(2-s))*th*(mi(:,:,2)*mi(:,:,1)+mi(:,:,1)*mi(:,:,2));
f=(xxi^(2-s))*((th^3)+3*(th^2)+th-1)*mi(:,:,2)*mi(:,:,2);
C2=-(xxi^(s-2))*(a+a'+b+c+f);
% Solving Riccati algebraic equation equ(69)
[mi(:,:,3)]=care(B2',I2,C2);
mi(:,:,3)=(th-1)*(1/th)*mi(:,:,2)+(1/th)*mi(:,:,3);
MI(:,:,3)=U'*mi(:,:,3)*U;
MMI(3)=phi'*MI(:,:,3)*phi;
%nth time step t=n*dt
disp('Calculating unit impulse at t=n*N*dt....')
clear a1;
a1=(xxi^(2-s))*((3*th-1)*mi(:,:,1)+(2-6*th+6*(th^2)-
(th^3))*mi(:,:,2)+((th-1)^3)*mi(:,:,3))+(th^3)*p2;
for i=3:ttsteps %A counter for the time steps
    clear Bm;clear Cm;
    t_m=((i-1)+th)*dt;
    Bm=a1+1.5*(th^2)*t_m*eye(Dof);
    ssm(:,:,:)=0;smm(:,:,:)=0;sm(:,:,:)=0;
    for j=1:i-1 %A counter for the time inside the integration
        if j==1
            sm(:,:,j)=.5*dt*(mi(:,:,j+1)+mi(:,:,j));
            ssm(:,:,j)=((dt^2)/6)*(2*mi(:,:,j)+mi(:,:,j+1));
        else
            sm(:,:,j)=.5*dt*(mi(:,:,j+1)+mi(:,:,j))+sm(:,:,j-1);
            ssm(:,:,j)=ssm(:,:,j-1)+dt*sm(:,:,j-
                        1)+((dt^2)/6)*(2*mi(:,:,j)+mi(:,:,j+1));
        end
    end
    %Convolution term
    for k=2:i-2 %A counter for the time inside the integration
        clear b1;clear c1;clear d1;
        b1=mi(:,:,i-k+2)*((4-th)*mi(:,:,k+1)+(th-1)*mi(:,:,k+2))*((th-
          1)^2);
        c1=mi(:,:,i-k+1)*((3-th)*mi(:,:,k+1)+th*mi(:,:,k+2))*(1+2*(2-
          th)*(th-1));
        d1=mi(:,:,i-k)*((2-th)*mi(:,:,k+1)+(th+1)*mi(:,:,k+2))*((2-
          th)^2);
        smm(:,:,k)=b1+c1+d1+smm(:,:,k-1);
    end
```

```
     if i==3
         k=1;
     end
     f1=mi(:,:,i-1)*((2-th)*mi(:,:,2)+(th+1)*mi(:,:,3))*th*((2-th)^2);
    g1=th*t_m*(6*(1/dt)*sm(:,:,j)+3*th*mi(:,:,i));
    h1=p2*(1/dt)*(6*th*(1/dt)*ssm(:,:,j)+6*(th^2)*sm(:,:,j)+2*(th^3)*dt*
    mi(:,:,i));
    k1=(mi(:,:,1)+(-2*(th^3)+6*(th^2)-2)*mi(:,:,2)+((th-
    1)^2)*(2*th+1)*mi(:,:,3))*mi(:,:,i);
    l1=-th*t_m*(1/dt)*(-(xxi^(s-
    2))*(t_m^2)*6*p3*(1/(dt^2))+(xxi^s)*6*mo);
    n1=mi(:,:,i)*(mi(:,:,1)+((th^4)-5*(th^3)+6*(th^2)+4*th-
    2)*mi(:,:,2)+(-(th^4)+2*(th^3)+3*(th^2)-4*th+1)*mi(:,:,3));
    Cm=(xxi^(2-s))*th*smm(:,:,k)+(xxi^(2-s))*f1+g1+h1+h1'+(xxi^(2-
    s))*k1+l1+(xxi^(2-s))*n1;
     mi(:,:,i+1)=lyap(Bm,Cm);
     mi(:,:,i+1)=(th-1)*(1/th)*mi(:,:,i)+(1/th)*mi(:,:,i+1);
     MI(:,:,i+1)=U'*mi(:,:,i+1)*U;
     MMI(i+1)=phi'*MI(:,:,i+1)*phi;
end
%Plot t-MI diagram at first node
for i=1:(ttsteps+1)
    plot((i-1)*dt,MMI(i),'r');
    hold on
end
if M<=floor(nt*(1/dt))
    MMI(ttsteps+2)=((MMI(ttsteps+1)-MMI(ttsteps))*(round(nt*(1/dt))-
    ttsteps))+MMI(ttsteps+1);
    plot(round(nt*(1/dt))*dt,MMI(ttsteps+2),'r');
    hold off
else
    hold off
end
end
```

3.3.6.Function-*disp_vel_acc*

In this function the displacement, velocity and acceleration of every point in the model is calculated.

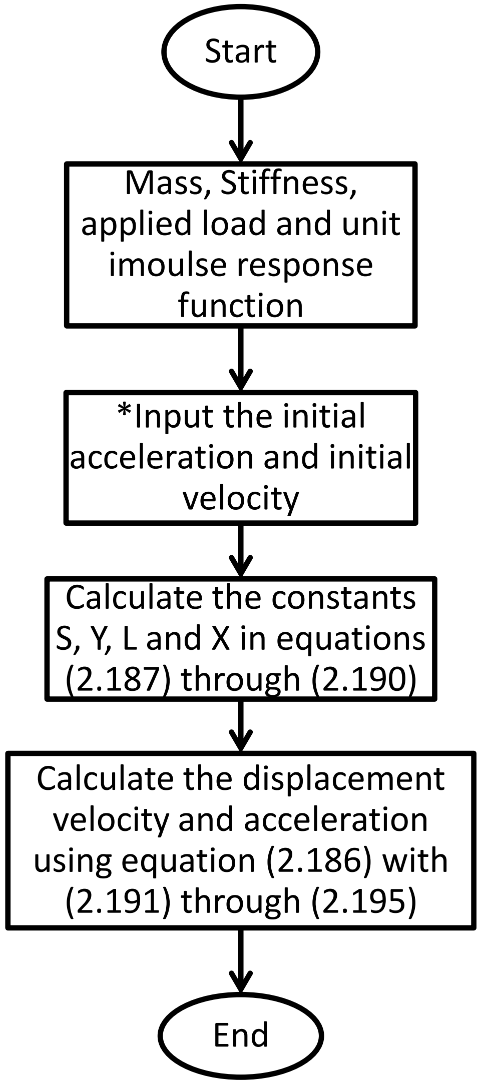


Fig.(3.16): disp_vel_acc function flow chart

In this function to input files are needed for acceleration and velocity, int_acc and int_vel, respectively. The input in each file is a column vector whose row number is the number of unrestraint Dof. Here is the MATLAB code for this function.

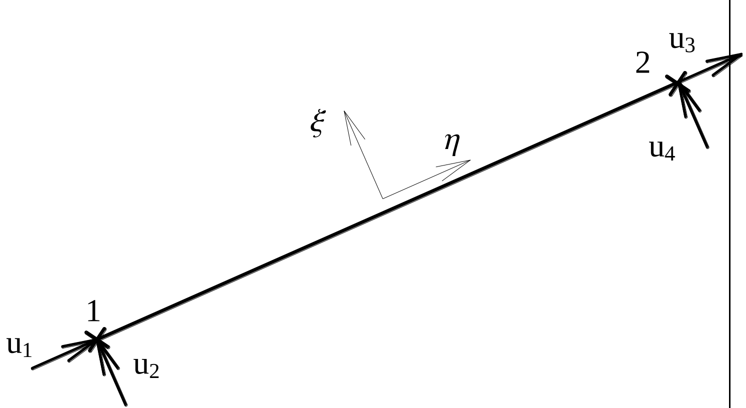
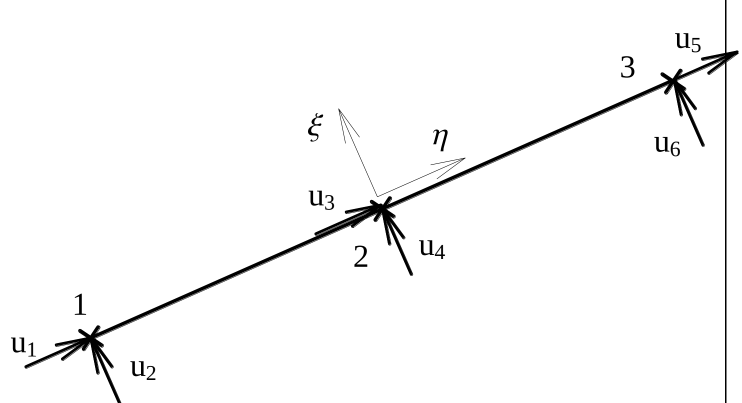
```
function [ud,uv,ua]=disp_vel_acc(MI,Mass,K,C,P,nt,dt,Nn,M,Dof,Dof2)
%A function to calculate the displacement velocity and acceleration of
the
%elastic space using coupled FE/SBFE methods
%Dof: number of degrees of freedom of the system
%Dof2: number of degrees of freedom of the unbounded domain only
%Intializing ua,uv,ud...
clear u_v;clear smu1;clear r;clear q1;clear qq;
disp('initializing the response quantities....')
ua=zeros(Dof,round((nt*(1/dt))+1));%initializing the acceleration
uv=zeros(Dof,round((nt*(1/dt))+1));%initializing the velocity
ud=zeros(Dof,round((nt*(1/dt))+1));%initializing the displacement
u_v=zeros(Dof2,round((nt*(1/dt))+1));%initializing u_v
r=zeros(Dof,round((nt*(1/dt))+1));%initializing the force
q1=zeros(Dof,round((nt*(1/dt))+1));%initializing q1
%qq=zeros(Dof,Dof);%initializing qq
disp('Loading the initial acceleration....')
%ua(:,1)=xlsread('coordinates.xlsx',-1); %initial acceleration (t=0)
ua(:,1)=load('int_acc.txt'); %initial acceleration (t=0)
disp('Loading the initial velocity....')
%uv(:,1)=xlsread('coordinates.xlsx',-1); %initial velocity (t=0)
uv(:,1)=load('int_vel.txt'); %initial velocity (t=0)
%For the theoretical bases of the solution see ref. "High preformance
SBFEM"
disp('Calculating the constants in the equation of motion....')
ga=0.5;beta=0.25;
a1=ga/(beta*dt);a10=1/(Nn*dt);a4=(ga/beta)-1;a5=0.5*dt*((ga/beta)-2);
%qq(1:Dof2,1:Dof2)=(a1-a10)*MI(:,:,1)+a10*MI(:,:,2);
clear S;clear Y;clear L;clear X;clear INVS;
disp('Y')
Y=(1/(beta*(dt^2)))*Mass+a1*C;
disp('S')
%S=Y+K+qq;
S=Y+K;
S(1:Dof2,1:Dof2)=S(1:Dof2,1:Dof2)+(a1-a10)*MI(:,:,1)+a10*MI(:,:,2);
disp('INVS')
INVS=inv(S);
disp('L')
L=(1/(beta*dt))*Mass+a4*C;
disp('X')
X=((0.5-beta)/beta)*Mass+a5*C;
disp('Calculating the response at d_t=1....')
%From t=0 to t=Nn*dt
for i=1:Nn
    u_v(:,i+1)=-(a1*ud(1:Dof2,i)+a4*uv(1:Dof2,i)+a5*ua(1:Dof2,i));
    r(1:Dof2,i+1)=MI(:,:,1)*(u_v(:,i+1)+a10*ud(1:Dof2,1)-(1-
(i/Nn))*uv(1:Dof2,1))+MI(:,:,2)*(-a10*ud(1:Dof2,1)-(i/Nn)*uv(1:Dof2,1));
    ud(:,i+1)=INVS*(P(:,i+1)-r(:,i+1)+Y*ud(:,i)+L*uv(:,i)+X*ua(:,i));
    uv(:,i+1)=a1*(ud(:,i+1)-ud(:,i))-a4*uv(:,i)-a5*ua(:,i);
    ua(:,i+1)=(1/(beta*(dt^2)))*(ud(:,i+1)-ud(:,i)-dt*uv(:,i)-(0.5-
beta)*(dt^2)*ua(:,i));
end
%From t=Nn*dt to t=m*Nn*dt (m is the no of steps of the unit impulse
before time of calculation of the force)
```

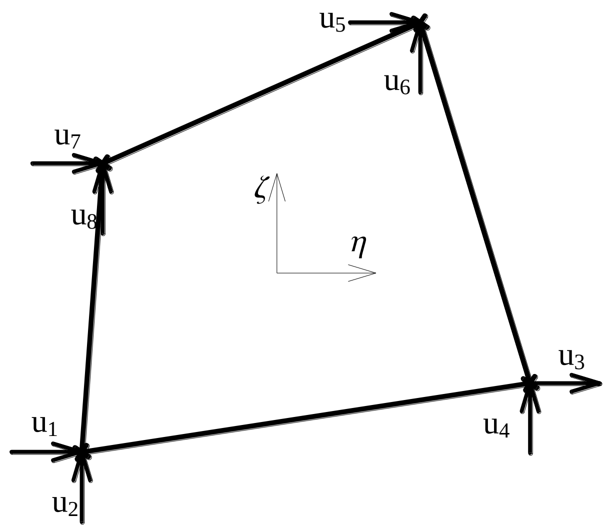
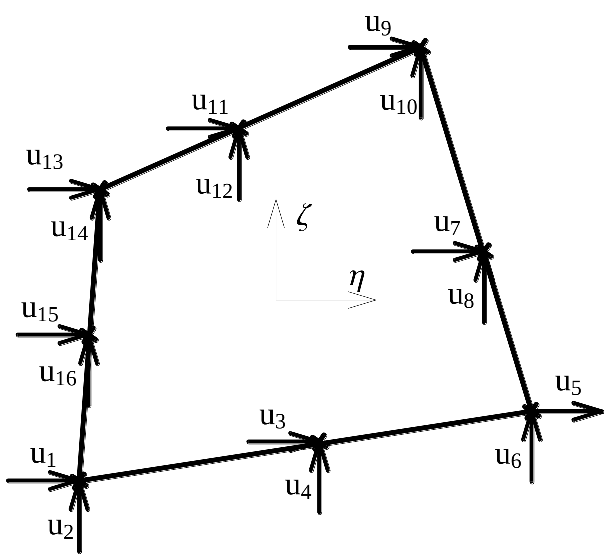
```
%m from 2 to M-1 (M is the step at which the unit impulse function
changes
%from non-linear to nearly linear)
%Calculation of the force between time (Nn*dt) to (2*Nn*dt) take m=1
disp('Calculating the response at from d_t=1 to d_t=2....')
m=1;
for i=(Nn+1):(2*Nn)
    u_v(:,i+1)=-(a1*ud(1:Dof2,i)+a4*uv(1:Dof2,i)+a5*ua(1:Dof2,i));
    q1(1:Dof2,i+1)=MI(:,:,1)*(u_v(:,i+1)+a10*ud(1:Dof2,i-Nn+1))-
MI(:,:,2)*(uv(1:Dof2,i-Nn+1)+a10*ud(1:Dof2,i-Nn+1));
    Kn=i-m*Nn;
    r(1:Dof2,i+1)=q1(1:Dof2,i+1)+MI(:,:,m+1)*((-
1+(Kn/Nn))*uv(1:Dof2,1)+uv(1:Dof2,i-m*Nn+1)+a10*(ud(1:Dof2,1)-
ud(1:Dof2,i-m*Nn+1)))+MI(:,:,m+2)*(-
(Kn/Nn)*uv(1:Dof2,1)+a10*(ud(1:Dof2,i-m*Nn+1)-ud(1:Dof2,1)));
    ud(:,i+1)=INVS*(P(:,i+1)-r(:,i+1)+Y*ud(:,i)+L*uv(:,i)+X*ua(:,i));
    uv(:,i+1)=a1*(ud(:,i+1)-ud(:,i))-a4*uv(:,i)-a5*ua(:,i);
    ua(:,i+1)=(1/(beta*(dt^2)))*(ud(:,i+1)-ud(:,i)-dt*uv(:,i)-(0.5-
beta)*(dt^2)*ua(:,i));
end
%Calculation of the force between time (2*Nn*dt) to (m*Nn*dt), m from 2
to
%(M-1)
disp('Calculating the response at from d_t=2 to d_t=M-1....')
for m=2:(M-1)
    for i=(m*Nn+1):((m+1)*Nn)
        smu1=zeros(Dof2,m+1);%initializing smu1
        u_v(:,i+1)=-(a1*ud(1:Dof2,i)+a4*uv(1:Dof2,i)+a5*ua(1:Dof2,i));
        q1(1:Dof2,i+1)=MI(:,:,1)*(u_v(:,i+1)+a10*ud(1:Dof2,i-Nn+1))-
MI(:,:,2)*(uv(1:Dof2,i-Nn+1)+a10*ud(1:Dof2,i-Nn+1));
        for j=2:m
            smu1(:,j+1)=MI(:,:,j+1)*(a10*(ud(1:Dof2,i-(j-1)*Nn+1)-
ud(1:Dof2,i-j*Nn+1))-uv(1:Dof2,i-j*Nn+1))-MI(:,:,j)*(a10*(ud(1:Dof2,i-
(j-1)*Nn+1)-ud(1:Dof2,i-j*Nn+1))-uv(1:Dof2,i-(j-1)*Nn+1))+smu1(:,j);
        end
        Kn=i-m*Nn;
        r(1:Dof2,i+1)=q1(1:Dof2,i+1)+smu1(:,j+1)+MI(:,:,m+1)*((-
1+(Kn/Nn))*uv(1:Dof2,1)+uv(1:Dof2,i-m*Nn+1)+a10*(ud(1:Dof2,1)-
ud(1:Dof2,i-m*Nn+1)))+MI(:,:,m+2)*(-
(Kn/Nn)*uv(1:Dof2,1)+a10*(ud(1:Dof2,i-m*Nn+1)-ud(1:Dof2,1)));
        ud(:,i+1)=INVS*(P(:,i+1)-
r(:,i+1)+Y*ud(:,i)+L*uv(:,i)+X*ua(:,i));
        uv(:,i+1)=a1*(ud(:,i+1)-ud(:,i))-a4*uv(:,i)-a5*ua(:,i);
        ua(:,i+1)=(1/(beta*(dt^2)))*(ud(:,i+1)-ud(:,i)-dt*uv(:,i)-(0.5-
beta)*(dt^2)*ua(:,i));
    end
end
%Calculation of the force between time (M*Nn*dt) to (nt*(1/dt))
disp('Calculating the response at from d_t=M to d_t=nt/d_t....')
clear smu1;
for i=M*Nn+1:(nt*(1/dt))
    smu1=zeros(Dof2,M+1);%initializing smu1
    u_v(:,i+1)=-(a1*ud(1:Dof2,i)+a4*uv(1:Dof2,i)+a5*ua(1:Dof2,i));
```

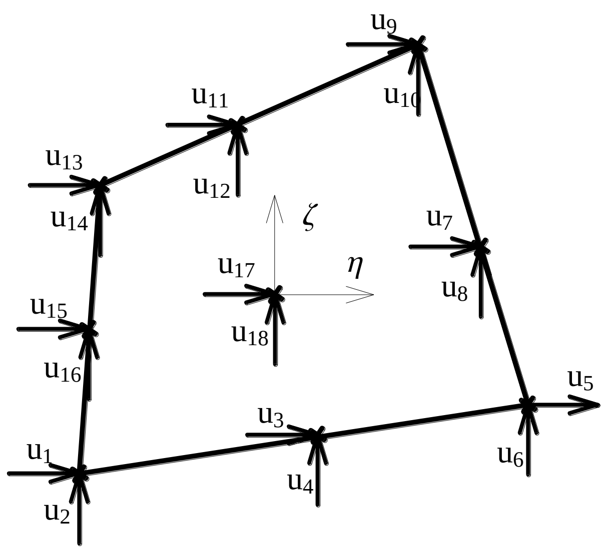
```
    q1(1:Dof2,i+1)=MI(:,:,1)*(u_v(:,i+1)+a10*ud(1:Dof2,i-Nn+1))-
MI(:,:,2)*(uv(1:Dof2,i-Nn+1)+a10*ud(1:Dof2,i-Nn+1));
    for j=2:M
        smu1(:,j+1)=MI(:,:,j+1)*(a10*(ud(1:Dof2,i-(j-1)*Nn+1)-
ud(1:Dof2,i-j*Nn+1))-uv(1:Dof2,i-j*Nn+1))-MI(:,:,j)*(a10*(ud(1:Dof2,i-
(j-1)*Nn+1)-ud(1:Dof2,i-j*Nn+1))-uv(1:Dof2,i-(j-1)*Nn+1))+smu1(:,j);
    end
    Kn=i-M*Nn;

r(1:Dof2,i+1)=q1(1:Dof2,i+1)+smu1(1:Dof2,j+1)+MI(:,:,M)*((Kn/Nn)*uv(1:Do
f2,1)-a10*(ud(1:Dof2,Kn+1)-ud(1:Dof2,1)))+MI(:,:,M+1)*(uv(1:Dof2,Kn+1)-
(1+(Kn/Nn))*uv(1:Dof2,1)+a10*(ud(1:Dof2,Kn+1)-ud(1:Dof2,1)));
    ud(:,i+1)=INVS*(P(:,i+1)-r(:,i+1)+Y*ud(:,i)+L*uv(:,i)+X*ua(:,i));
    uv(:,i+1)=a1*(ud(:,i+1)-ud(:,i))-a4*uv(:,i)-a5*ua(:,i);
    ua(:,i+1)=(1/(beta*(dt^2)))*(ud(:,i+1)-ud(:,i)-dt*uv(:,i)-(0.5-
beta)*(dt^2)*ua(:,i));
end
%Plot t-ud diagram at first node
disp('Plot t-ud diagram at first node - first row....')
for i=1:round((nt*(1/dt))+1)
    if i==1
        plot((i-1)*dt,ud(1,1),'r');
    else
        plot((i-1)*dt,ud(1,i),'r');
    end
    hold on
end
hold off
end
```

The shape functions used in the calculations are Lagrange`s functions except for the frame element Hermit functions are used instead (see [3] chapters 8 and 9, respectively), as shown below:

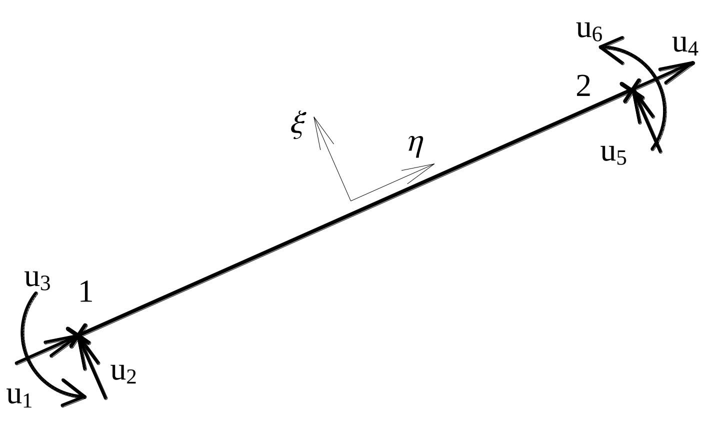
For SBFE:

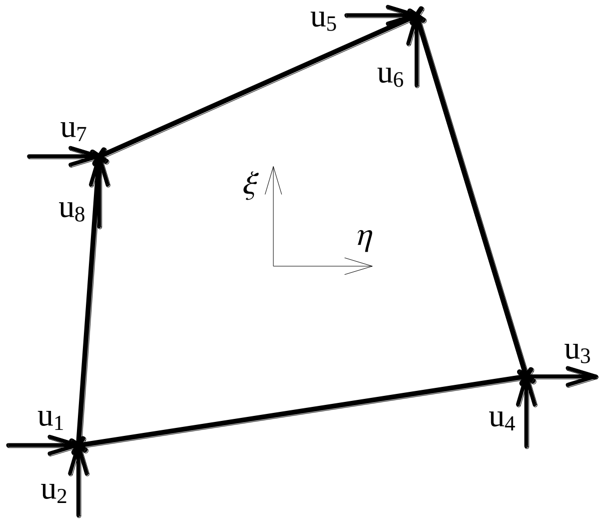
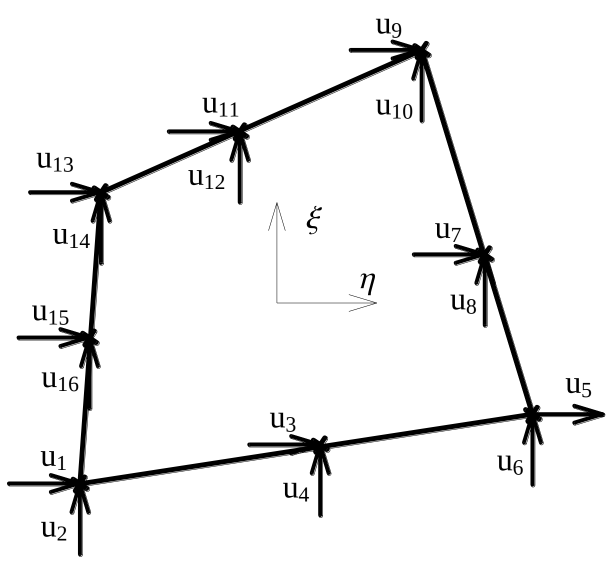
Type of problem	Element type	Element Dofs	Shape functions
2D	L2		$N_1 = \frac{1}{2}(1-\eta)$ $N_2 = \frac{1}{2}(1+\eta)$
	L3		$N_1 = -\frac{1}{2}\eta(1-\eta)$ $N_2 = (1-\eta)(1+\eta)$ $N_3 = \frac{1}{2}\eta(1+\eta)$

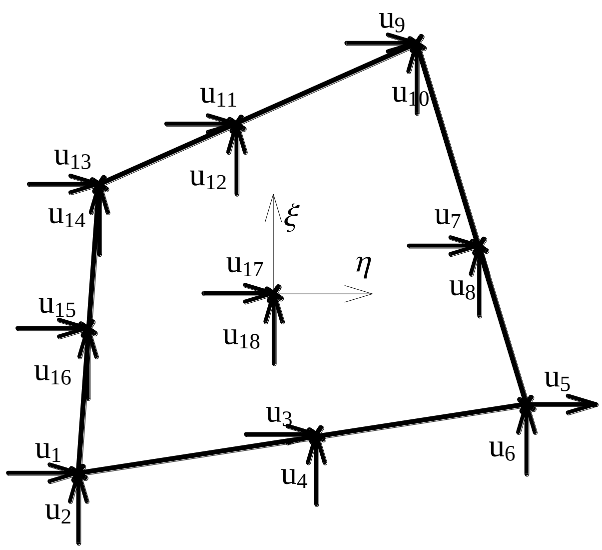
<table>
<tr><td rowspan="2">3D</td><td>Q4</td><td>
</td><td>$N_1 = \frac{1}{4}(1+\eta)(1+\zeta)$
$N_2 = \frac{1}{4}(1-\eta)(1+\zeta)$
$N_3 = \frac{1}{4}(1-\eta)(1-\zeta)$
$N_4 = \frac{1}{4}(1+\eta)(1-\zeta)$</td></tr>
<tr><td>Q8</td><td>
</td><td>$N_1 = \frac{1}{4}(1+\eta)(1+\zeta)(\eta+\zeta-1)$
$N_2 = \frac{1}{2}(1-\eta^2)(1+\zeta)$
$N_3 = \frac{1}{4}(1-\eta)(1+\zeta)(\zeta-\eta-1)$
$N_4 = \frac{1}{2}(1-\zeta^2)(1-\eta)$
$N_5 = \frac{1}{4}(1-\eta)(1-\zeta)(-\eta-\zeta-1)$
$N_6 = \frac{1}{2}(1-\eta^2)(1-\zeta)$
$N_7 = \frac{1}{4}(1+\eta)(1-\zeta)(\eta-\zeta-1)$
$N_8 = \frac{1}{2}(1-\zeta^2)(1+\eta)$</td></tr>
</table>

	Q9		$N_1 = \frac{1}{4}\eta\zeta(1+\eta)(1+\zeta)$ $N_2 = \frac{1}{2}\zeta(1-\eta^2)(1+\zeta)$ $N_3 = -\frac{1}{4}\eta\zeta(1-\eta)(1+\zeta)$ $N_4 = -\frac{1}{2}\eta(1-\zeta^2)(1-\eta)$ $N_5 = \frac{1}{4}\eta\zeta(1-\eta)(1-\zeta)$ $N_6 = -\frac{1}{2}\zeta(1-\eta^2)(1-\zeta)$ $N_7 = -\frac{1}{4}\eta\zeta(1+\eta)(1-\zeta)$ $N_8 = \frac{1}{2}\eta(1-\zeta^2)(1+\eta)$ $N_9 = (1-\eta^2)(1-\zeta^2)$

For FE:

Type of problem	Element type	Element Dofs	Shape functions
2D	F2		$N_1 = \frac{1}{2}(1-\eta)$ $N_2 = \frac{1}{4}(\eta^3 - 3\eta + 2)$ $N_3 = \frac{L}{8}(\eta^3 - \eta^2 - \eta + 1)$ $N_4 = \frac{1}{2}(1+\eta)$ $N_5 = \frac{1}{4}(-\eta^3 + 3\eta + 2)$ $N_6 = \frac{L}{8}(\eta^3 + \eta^2 - \eta - 1)$

	Q4		$N_1 = \frac{1}{4}(1+\eta)(1+\xi)$ $N_2 = \frac{1}{4}(1-\eta)(1+\xi)$ $N_3 = \frac{1}{4}(1-\eta)(1-\xi)$ $N_4 = \frac{1}{4}(1+\eta)(1-\xi)$
	Q8		$N_1 = \frac{1}{4}(1+\eta)(1+\xi)(\eta+\xi-1)$ $N_2 = \frac{1}{2}(1-\eta^2)(1+\xi)$ $N_3 = \frac{1}{4}(1-\eta)(1+\xi)(\xi-\eta-1)$ $N_4 = \frac{1}{2}(1-\xi^2)(1-\eta)$ $N_5 = \frac{1}{4}(1-\eta)(1-\xi)(-\eta-\xi-1)$ $N_6 = \frac{1}{2}(1-\eta^2)(1-\xi)$ $N_7 = \frac{1}{4}(1+\eta)(1-\xi)(\eta-\xi-1)$ $N_8 = \frac{1}{2}(1-\xi^2)(1+\eta)$

	Q9		$N_1=\frac{1}{4}\eta\xi(1+\eta)(1+\xi)$ $N_2=\frac{1}{2}\xi(1-\eta^2)(1+\xi)$ $N_3=-\frac{1}{4}\eta\xi(1-\eta)(1+\xi)$ $N_4=-\frac{1}{2}\eta(1-\xi^2)(1-\eta)$ $N_5=\frac{1}{4}\eta\xi(1-\eta)(1-\xi)$ $N_6=-\frac{1}{2}\xi(1-\eta^2)(1-\xi)$ $N_7=-\frac{1}{4}\eta\xi(1+\eta)(1-\xi)$ $N_8=\frac{1}{2}\eta(1-\xi^2)(1+\eta)$ $N_9=(1-\eta^2)(1-\xi^2)$

Chapter 4: Validation of the program using some benchmark problems

4.1. Introduction

In this chapter a validation of the program is done using some benchmark problems, in 3 dimensions and 2 dimensions.

In the stated problems E is the Young`s modulus, G is the shear modulus, $\nu$ is Poisson`s ratio, t is the time, ω is the circular frequency, ρ is the mass density, c_s is the shear wave velocity, c_p is the longitudinal wave velocity and a_o is the dimensionless frequency.

Where, $c_s = \sqrt{\frac{G}{\rho}}$ and $c_p = \sqrt{\frac{2(1-\nu)}{(1-2\nu)}} c_s$

In the problem where coupled FE/SBFE is used damping is not taken in consideration

4.2. Spherical cavity in full space subjected to uniform radial load

Here is a very simple example, a one dimensional wave propagation problem having an analytical solution ([7] Appendix A) is introduced, a spherical cavity embedded in a full space with radius r_o and subjected to uniform radial load as shown (fig.(4.1)).

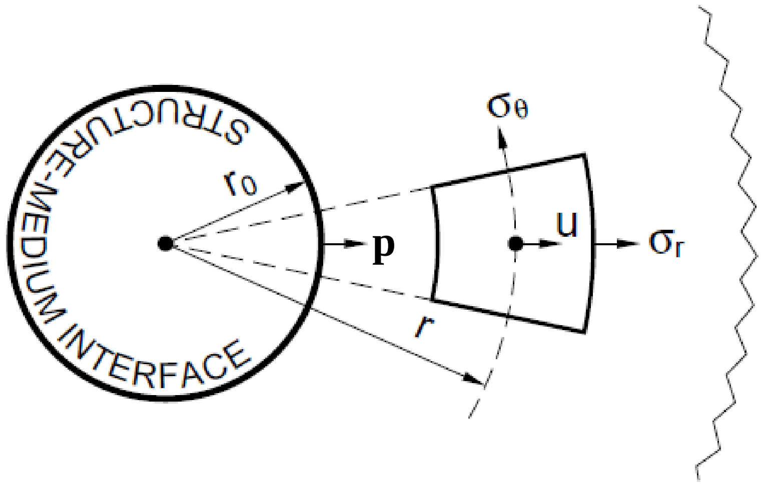


Fig.(4.1): A spherical cavity subjected to uniform radial loading

Since, the load is symmetric therefore there will be no shear stresses so the stress strain relationship can be written in spherical coordinates as follows:

$$\begin{Bmatrix} \sigma_r \\ \sigma_\theta \\ \sigma_\varphi \end{Bmatrix} = \frac{2G}{1-2\nu} \begin{bmatrix} (1-\nu) & \nu & \nu \\ \nu & (1-\nu) & \nu \\ \nu & \nu & (1-\nu) \end{bmatrix} \begin{Bmatrix} \varepsilon_r \\ \varepsilon_\theta \\ \varepsilon_\varphi \end{Bmatrix} \tag{4.1}$$

where, $\sigma_r, \sigma_\theta, \sigma_\varphi$, ε_r , $\varepsilon_\theta, \varepsilon_\varphi$ are the stresses and strains in the direction of the radius r and the angles θ and φ, respectively.

$$\varepsilon_r = u_{,r} \tag{4.2}$$

$$\varepsilon_\theta = \varepsilon_\varphi = \frac{1}{r} u \tag{4.3}$$

where, u is the displacement in the radial direction.

The equation of motion in r direction can be written as follows:

$$\sigma(t)_{r,r} + \frac{(2\sigma(t)_r - \sigma(t)_\theta - \sigma(t)_\varphi)}{r} = \rho \ddot{u}(t) \tag{4.4}$$

Substituting by equation (4.1), (4.2) and (4.3) in (4.4) the following differential equation results:

$$u(t)_{,rr} + \frac{2}{r} u(t)_{,r} - \frac{2}{r^2} u(t) - \frac{1}{c_p^2} \ddot{u}(t) = 0 \tag{4.5}$$

The dynamic stiffness will be first calculated and from it the unit impulse response for acceleration will be calculated.

Suppose the spherical cavity is subjected to a uniform displacement u_o (t).

Therefore,

The initial conditions: $u(r=r_o)=u_o$, $u(t=0)=\dot{u}(t=0)=0$ (4.6)

where, u_o is the displacement at the medium interface

To solve the above equation it is written in frequency domain:

$$u(\omega)_{,rr}+\frac{2}{r}u(\omega)_{,r}-\frac{2}{r^2}u(\omega)+\frac{\omega^2}{c_p^2}u(\omega)=0 \quad (4.7)$$

As solved in ([7] Appendix A) the displacement is

$$u(\omega)=\frac{c_1}{r^2}\left(-1+i\frac{\omega}{c_p}r\right)e^{+i\frac{\omega}{c_p}r}+\frac{c_2}{r^2}\left(-1-i\frac{\omega}{c_p}r\right)e^{-i\frac{\omega}{c_p}r} \quad (4.8)$$

, where, c_1 and c_2 are arbitrary constants

In order to satisfy the radiation condition (i.e. the displacement must decrease as r approaches infinity) c_1 must equal zero. So using the initial conditions (4.6) equation (4.8) will be written as:

$$u(\omega)=u_o(\omega)\left(\frac{r_o}{r}\right)^2\left(\frac{c_p+i\omega r}{c_p+i\omega r_o}\right)e^{-i\frac{\omega}{c_p}(r-r_o)} \quad (4.9)$$

To get the dynamic stiffness, the relation between the force and the dynamic stiffness in the frequency domain is written as:

$R(\omega)=S^{\infty}(\omega)u(\omega)$, where $R(\omega)$ is the internal force (4.10)

At the interface $R(\omega)=-4\pi r_o^2\sigma_r$ (4.11)

Substituting by (4.1) then (4.2) and (4.3), then (4.9) in (4.11), therefore the dynamic stiffness is:

$$S^{\infty}(\omega)=16\pi G r_o\left(1+\frac{1-\nu}{2(1-2\nu)}\frac{1}{c_p}\frac{(i\omega r_o)^2}{c_p+i\omega r_o}\right) \quad (4.12)$$

And at ω=0 the static stiffness is obtained

$$S^{\infty}(0)=K^{\infty}=16\pi G r_o \quad (4.13)$$

To get the dynamic stiffness for acceleration $M^{\infty}(\omega)$

Since, $M^{\infty}(\omega) = \dfrac{S^{\infty}(\omega)}{(i\omega)^2}$

Therefore, $M^{\infty}(\omega) = K^{\infty}\left(\dfrac{1}{(i\omega)^2} + \dfrac{1-\nu}{2(1-2\nu)}\dfrac{1}{c_p}\dfrac{r_o^{\,2}}{c_p + i\omega r_o}\right)$ (4.14)

Applying the inverse Fourier transform to the above equation, the unit impulse response for acceleration is obtained:

$$M^{\infty}(t) = K^{\infty}\left(t + \frac{r_o}{c_p}\frac{1-\nu}{2(1-2\nu)}e^{-\frac{r_o}{c_p}t}\right)H(t) \qquad (4.15)$$

where $H(t)$ is the Heaviside function.

The above problem is solved using the SBFEM, and due to axi-symmetry of the problem only an arc is modeled as shown in fig.(4.2) (see [13] for axisymmetric) where a four 3-noded line elements are used. The problem is solved using ν=0.25 and $\Delta t = 0.0346 t c_p / r_o$. The results for the unit impulse are shown in the fig.(4.3) for theta=1, 1.1,1.2, 1.3, 1.5 and 2 at N=5 to see the effect of theta and the total time is used with no truncation i.e. M=total.

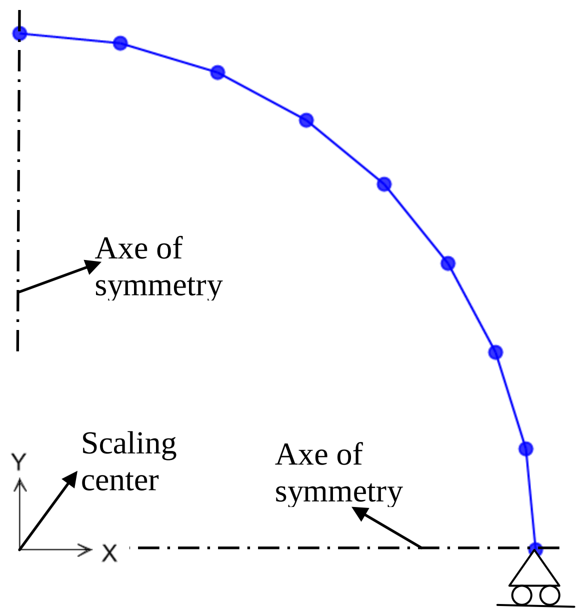


Fig.(4.2): The SBFE model of one arc of a spherical cavity

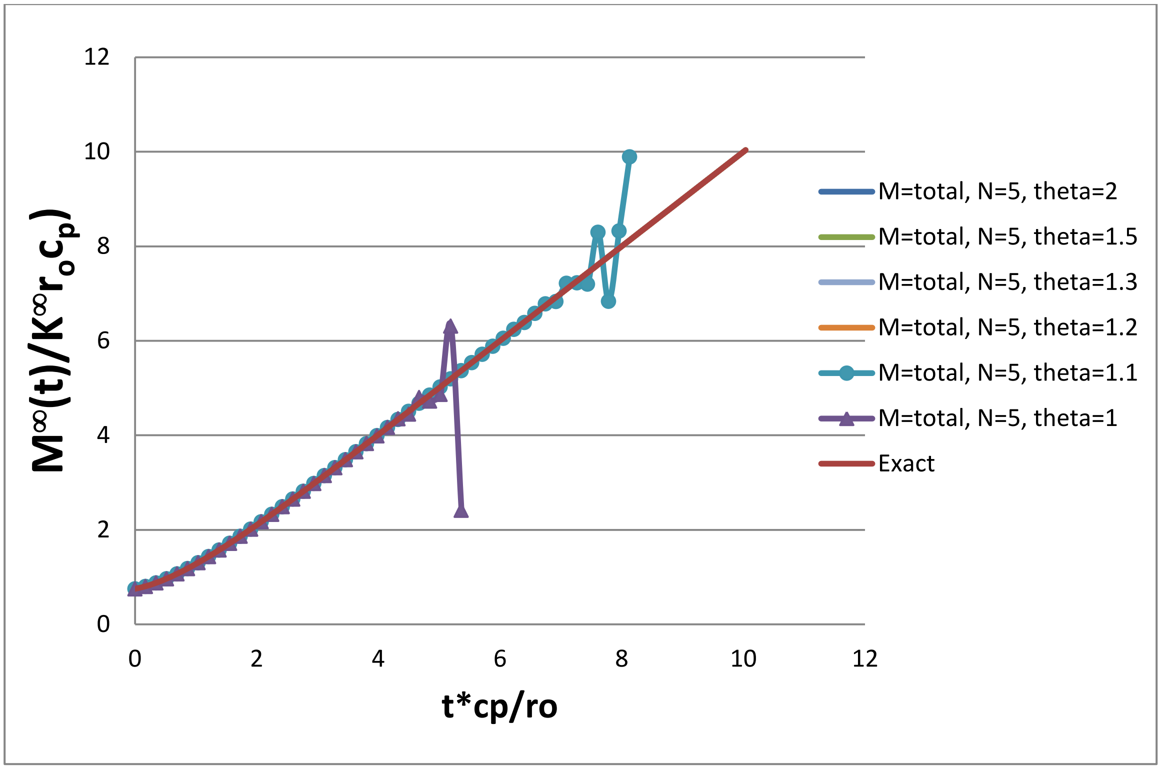


Fig.(4.3): The unit impulse response function for the spherical cavity subjected to uniform radial load for different values of theta

And here is a table comparing the values of the static stiffness K^{∞} with the exact solution for different values of theta and N:

$(K^{\infty}(approx.) - K^{\infty}(exact))/\min(K^{\infty}(approx.), K^{\infty}(exact))$ %			
θ / N	1	1.5	2
2	0.151687	0.01045	0.010446
4		0.01081	0.010776
5		0.010608	0.010544

It is clear from the above chart that excellent agreement is obtained compared to the exact solution except for theta=1 and 1.1 where the solution is unstable. Also the above table shows excellent agreement.

To see the effect of truncating the time of calculation of the unit impulse the problem is solved again using N=5, theta=1.3 and M=27, the result is shown in fig.(4.4) with the exact solution, where the two results agree.

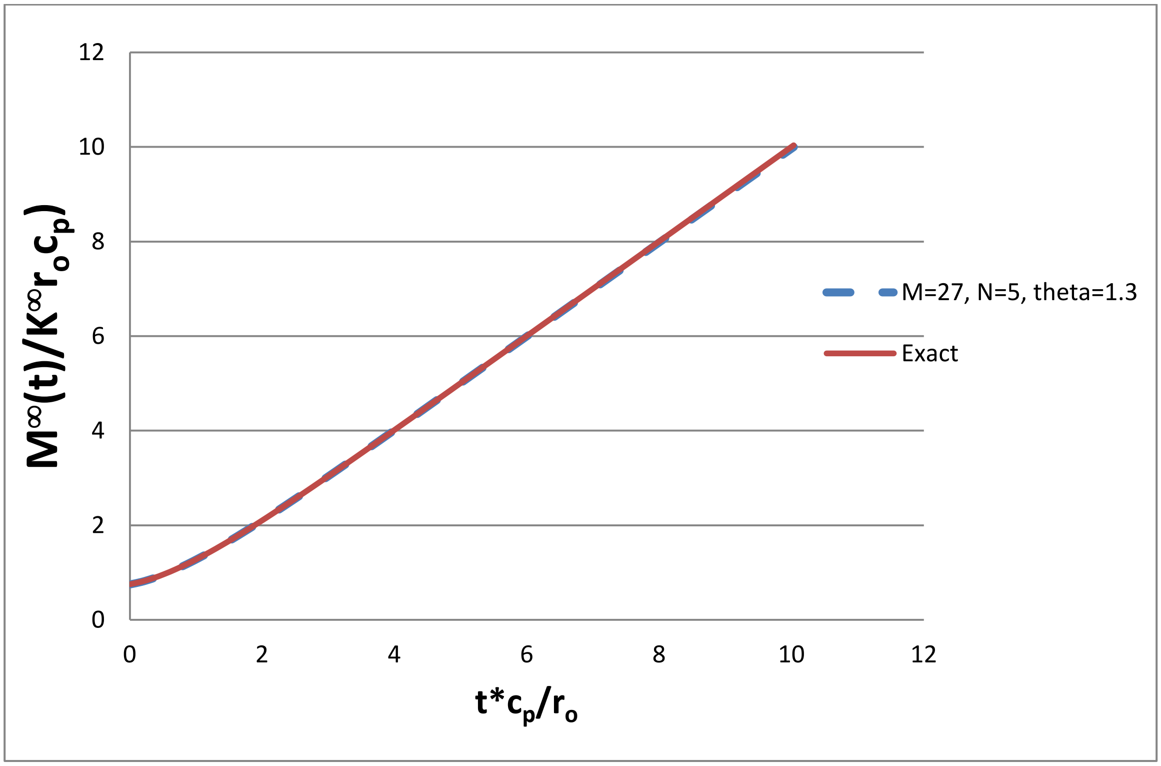


Fig.(4.4): The unit impulse response function for the spherical cavity subjected to uniform radial load for N=5 and M=27 compared to exact solution

Now we will calculate the displacement of the spherical interface due to a uniform distributed load p(t) (Rounded triangular impulse fig.(4.5)):

$$p(t) = \begin{cases} \dfrac{p_o}{2}\left(1 - \cos\dfrac{2\pi t}{t_o}\right) & 0 \le t \le t_o \\ 0 & t > t_o \end{cases} \quad \text{, where } t_o = 3.46 r_o / c_p \text{ function} \tag{4.16}$$

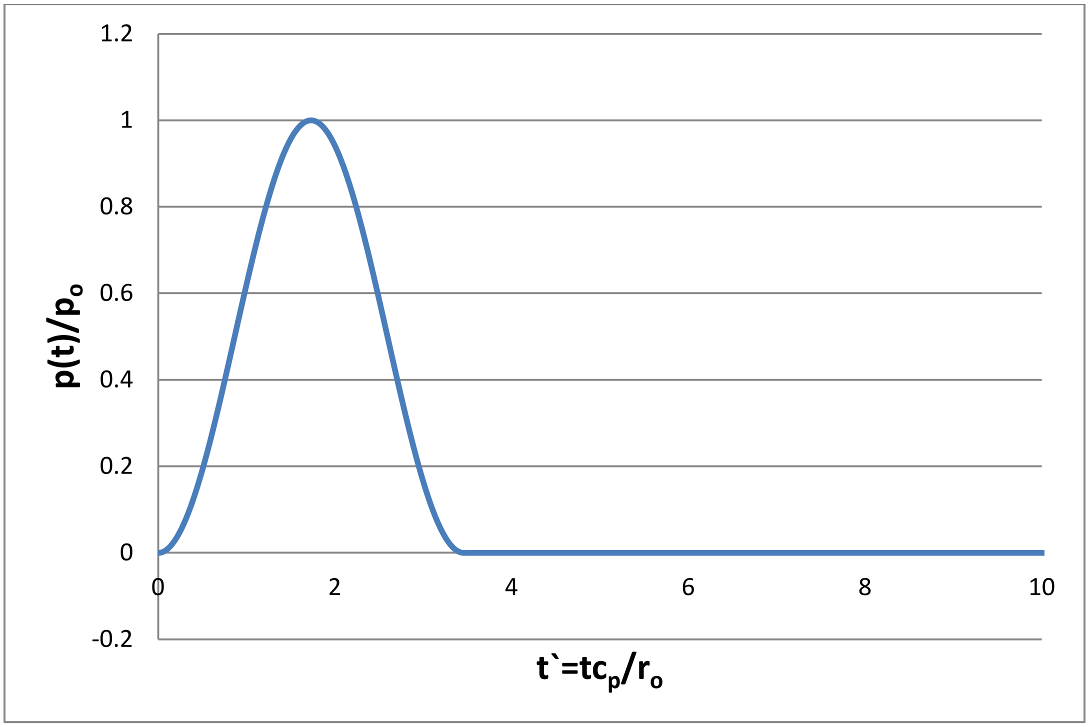


Fig.(4.5): Applied rounded triangular impulse load

To calculate the response of the sphere, equation (4.15) and (4.16) is substituted in the force-acceleration relationship:

$$R(t) = \int_0^t M^{\infty}(\tau)\ddot{u}(t-\tau)d\tau \tag{4.17}$$

The above equation is solved numerically to get the acceleration and the using Newmark time integration scheme the displacement is achieved.

The comparison between the above calculated displacement and the SBFE is shown in the fig.(4.6) below for N=5and M=total at different values of theta (1, 1.1, 1.2, 1.3, 1.5 and 2). Also good agreement is achieved except for the case of theta=1, 1.1 and 1.2.

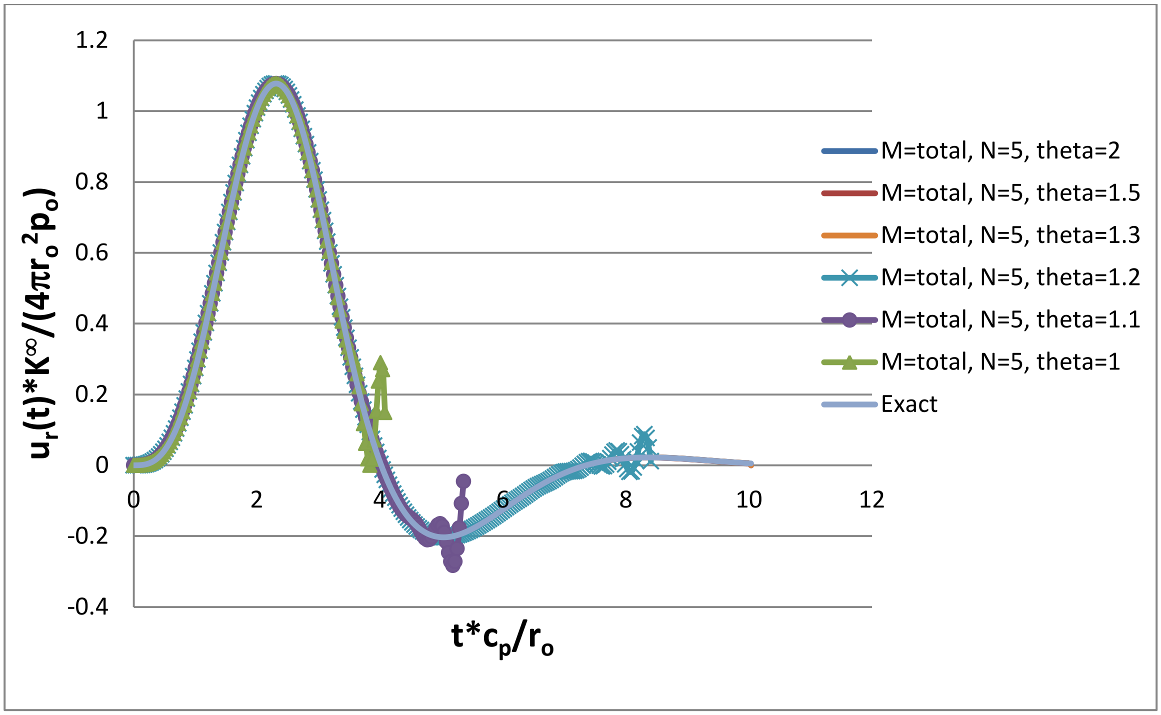


Fig.(4.6): The radial displacement of the spherical cavity for different values of theta

To see the effect of truncating the time of calculation of the unit impulse the results for the case of M=27, N=5 and theta=1.3 is shown below (fig.(4.7)) with the exact solution, where the two results agree.

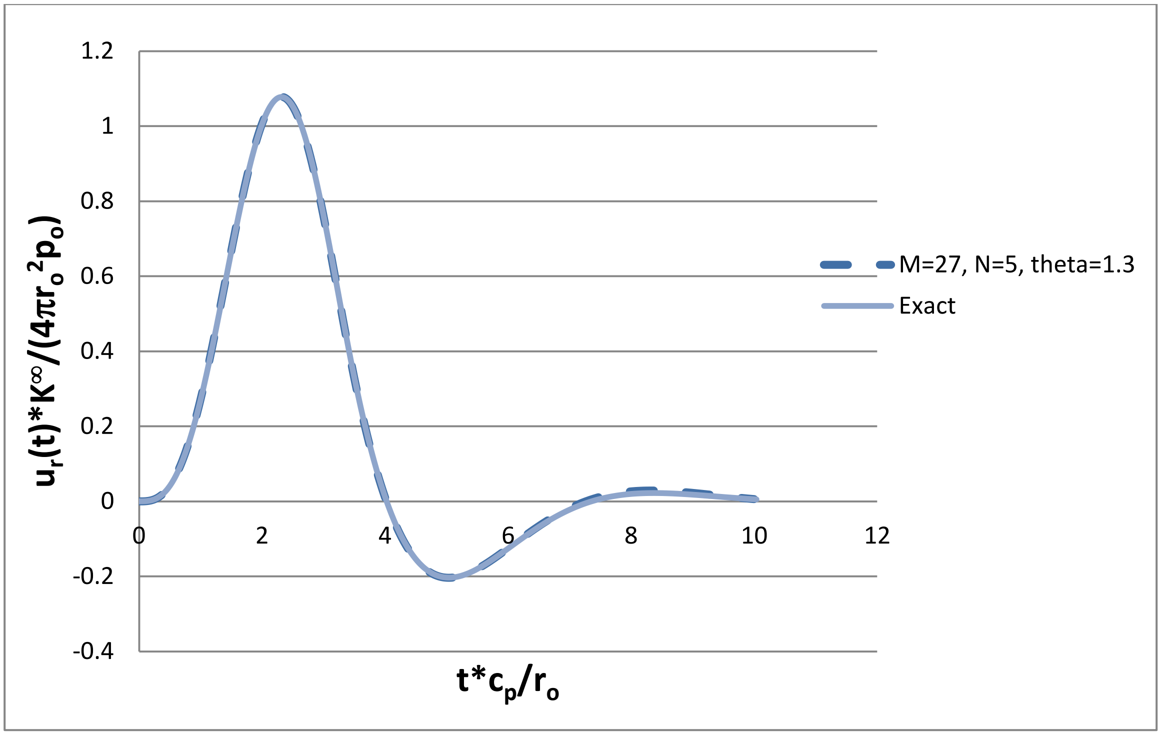


Fig.(4.7): The radial displacement of the spherical cavity for N=5 and M=27

4.3. In-plane motion of circular cavity subjected to radial load

In this problem the in plane motion of a circular cavity subjected to uniform radial load is discussed ([2], [18] and [19]), first the dynamic stiffness is calculated by introducing an in plane uniform radial displacement as shown in fig(4.8) then using the dynamic stiffness displacement is calculated for any given load (Fig(4.8) is in the x-y plane).

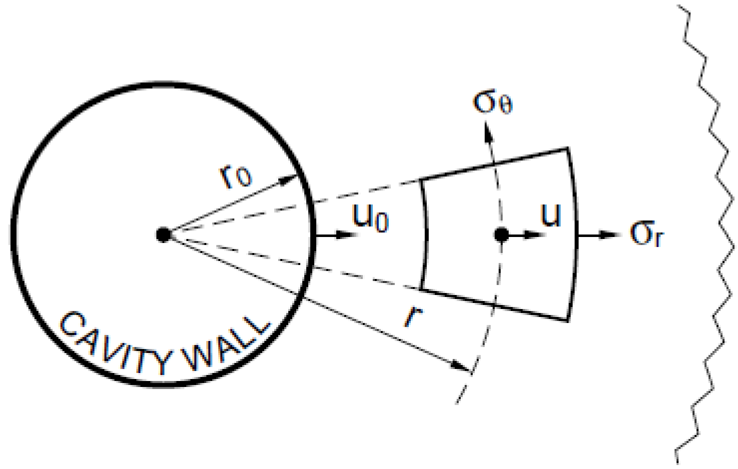


Fig.(4.8): Circular cavity subjected to uniform radial load

The stress strain relationship in polar coordinates will be:

$$\begin{Bmatrix} \sigma_r \\ \sigma_\theta \\ \tau_{r\theta} \end{Bmatrix} = \frac{2G}{(1-2\nu)} \begin{bmatrix} (1-\nu) & \nu & 0 \\ \nu & (1-\nu) & 0 \\ 0 & 0 & (1-2\nu)/2 \end{bmatrix} \begin{Bmatrix} \varepsilon_r \\ \varepsilon_\theta \\ \gamma_{r\theta} \end{Bmatrix} \tag{4.18}$$

where, $\varepsilon_r = u_{r,r}$ (4.19)

$$\varepsilon_\theta = \frac{1}{r} u_r + \frac{1}{r} u_{\theta,\theta} \tag{4.20}$$

and, $\gamma_{r\theta} = \frac{1}{r} u_{r,\theta} + u_{\theta,r} - \frac{1}{r} u_\theta$ (4.21)

Where, u_r and u_θ are the displacement in the radial and transversal directions.

The equations of motion (in time domain) are:

$$\sigma_{r,r} + \frac{1}{r}\tau_{r\theta,\theta} + \frac{1}{r}(\sigma_r - \sigma_\theta) = \rho \ddot{u}_r \quad \text{in r direction} \tag{4.22}$$

$$\tau_{r\theta,r} + \frac{1}{r}\sigma_{\theta,\theta} + \frac{2}{r}\tau_{r\theta} = \rho \ddot{u}_\theta \quad \text{in } \theta \text{ direction} \tag{4.23}$$

Since the displacement (or load) is uniform therefore: $\gamma_{r\theta} = 0$ and $u_\theta = 0$

Substituting by equation (4.19), (4.20) and (4.21) in (4.18) then substitute by the result in (4.22) and (4.23) after transforming them in the frequency domain, and take in consideration the above condition so the result is:

$$r^2 u_{r,rr} + r u_{r,r} + \left(\left(\frac{\omega r}{c_p} \right)^2 - 1 \right) u_r = 0 \tag{4.24}$$

Where, the boundary conditions are:

$$u_r(r = r_o) = u_o,\ u_r(t = 0) = 0 \text{ and } \dot{u}_r(t = 0) = 0 \tag{4.25}$$

$$u_r(r = \infty) = 0 \tag{4.26}$$

Let, $a = \dfrac{\omega r}{c_p}$ (dimensionless frequency) (4.27)

Substitute by (4.27) in (4.24) so the equation will be Bessel differential equation

$$a^2 u_{r,aa} + a u_{r,a} + \left(a^2 - 1\right) u_r = 0 \tag{4.28}$$

Therefore the solution is:

$$u_r = c_1 H_1^{(1)}(a) + c_2 H_1^{(2)}(a) \tag{4.29}$$

where, $H_1^{(1)}(a)$ is Hankel function of 1st kind 1st order, and $H_1^{(2)}(a)$ is Hankel function of 2nd kind 1st order

Using the boundary condition (4.26) (radiation condition)

Since, the asymptotic expansion of the Hankel function of 1st and 2nd kind as $r \to \infty$ is:

$$H_1^{(1)}(a) \approx \sqrt{\frac{2}{\pi a}} \exp i\left(a - \frac{3\pi}{4}\right) \tag{4.30}$$

$$H_1^{(2)}(a) \approx \sqrt{\frac{2}{\pi a}} \exp - i\left(a - \frac{3\pi}{4}\right) \tag{4.31}$$

So, at $u_r(r = \infty) = 0 \quad a \to \infty$, so c_1 must be zero

Therefore,

$$u_r = c H_1^{(2)}(a) \text{ , where } c = c_2 \tag{4.32}$$

Using the boundary condition (4.25), we get c, therefore:

$$u_r = \frac{u_o}{H_1^{(2)}(a_o)} H_1^{(2)}(a) \tag{4.33}$$

Where, $a_o = \dfrac{\omega r_o}{c_p}$ (4.34)

Using the force displacement relationship in frequency domain equation (4.10) at $r = r_o$ the dynamic stiffness is obtained:

$$R(\omega, r = r_o) = -\sigma_r(\omega, r = r_o) 2\pi r \tag{4.35}$$

Substitute by equations (4.33) in (4.19) and (4.20) then substitute by the result in (4.18) then (4.35) and using equation (4.10) the dynamic stiffness is obtained:

$$S^{\infty}(a_o) = \frac{2\pi E(1-\nu)}{(1+\nu)(1-2\nu)}\left[\frac{(1-2\nu)}{(1-\nu)} - a_o \frac{H_0^{(2)}(a_o)}{H_1^{(2)}(a_o)}\right] \tag{4.36}$$

To get the static stiffness substitute by $\omega = 0$ in (4.36), so using the asymptotic expansion for $\omega \to 0$ for the Hankel functions:

$$H_0^{(2)}(a_o) \approx -i\frac{2}{\pi}\ln\left(\frac{a_o}{2}\right) \tag{4.37}$$

$$H_1^{(2)}(a_o) \approx i\frac{2}{\pi a_o} \tag{4.38}$$

Substituting by (4.37) and (4.38) in (4.36) and substitute by $a_o = 0$

$$K^{\infty} = S^{\infty}(0) = \frac{2\pi E}{(1+\nu)} = 4\pi G \tag{4.39}$$

Also to get the damping coefficient, substitute by $a_o = \infty$ in (4.36) so we will use the asymptotic expansion for the Hankel function for $\omega \to \infty$:

$$H_o^{(2)}(a_o) \approx \sqrt{\frac{2}{\pi a_o}}\left(1 - \frac{1}{i8a_o}\right)\exp - i\left(a_o - \frac{\pi}{4}\right) \tag{4.40}$$

$$H_1^{(2)}(a_o) \approx \sqrt{\frac{2}{\pi a_o}}\left(1 + \frac{3}{i8a_o}\right)\exp - i\left(a_o - \frac{3\pi}{4}\right) \tag{4.41}$$

Since, $Lim_{a_o \to \infty} S^{\infty}(a_o) = i a_o C_{\infty} + K_{\infty}$ where C_{∞} and K_{∞} are the damping and static stiffness (at $\omega \to \infty$) coefficients, therefore:

$$C_{\infty} = 2\pi r_o \rho c_p \tag{4.42}$$

$$K_{\infty} = \frac{2\pi(1-3\nu)}{(1-2\nu)}G \tag{4.43}$$

Now applying the SBFEM the problem is discretized using four 3-noded line elements taking in consideration the symmetry of the problem as shown in fig.(4.9), the time step equals $0.182 r_o / c_p$. Fig.(4.10) shows the unit impulse response function at different values of θ (1, 1.5, 2) at N=5 and the total time is used with no truncation i.e. M=total, compared with the solution of program SIMILAR (a SBFE program made by Prof. J. Wolf and Prof. Chongmin Song).

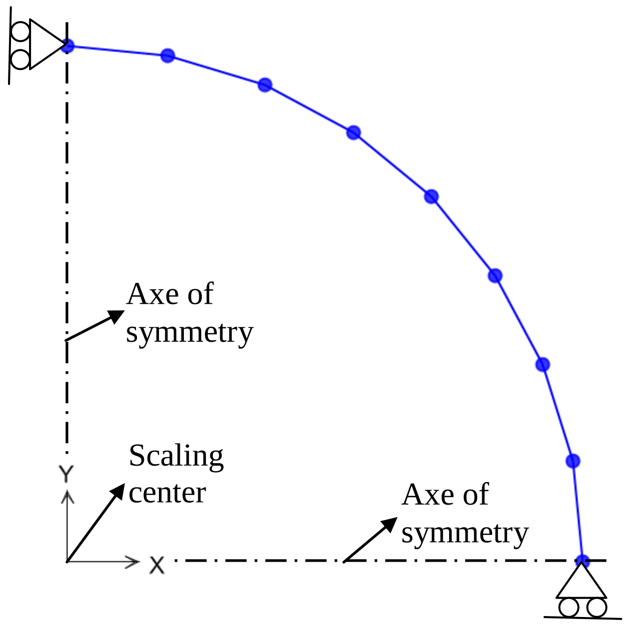


Fig.(4.9): The SBFE model of one arc of a circular cavity

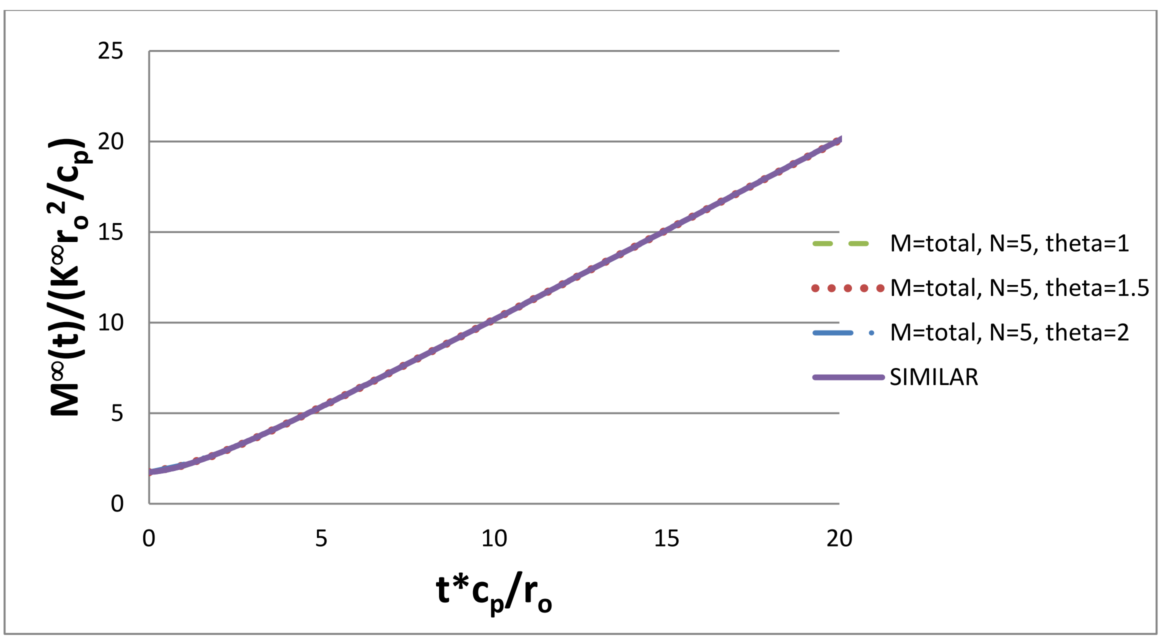


Fig.(4.10): The unit impulse response function due to radial uniform load for different values of theta

And here is a table comparing the values of the static stiffness K^{∞} with the exact solution for different values of θ and N:

$(K^{\infty}(approx.) - K^{\infty}(exact)) / \min(K^{\infty}(approx.), K^{\infty}(exact))$ %			
N \ θ	1	1.5	2
2		0.451782	0.451784
4	21.84889	0.496216	0.496098
5	1.3804713	0.482063	0.481781

For C_{∞} the ratio $(C_{\infty}(approx.) - C_{\infty}(exact) / \min(C_{\infty}(approx.), C_{\infty}(exact))$ =0.006863%

It is clear from the above chart that excellent agreement is obtained compared to the SIMILAR solution. Also the above table shows excellent agreement, except for the case of θ=1 at N=4 and also at N=2 (where the ratio is not written as the solution is unstable)

To see the effect of truncating the time of calculation of the unit impulse the problem is solved again using N=5, θ=1.5 and M=10, the result is shown in fig.(4.11) with the solution at M=total (111), where the two results agree.

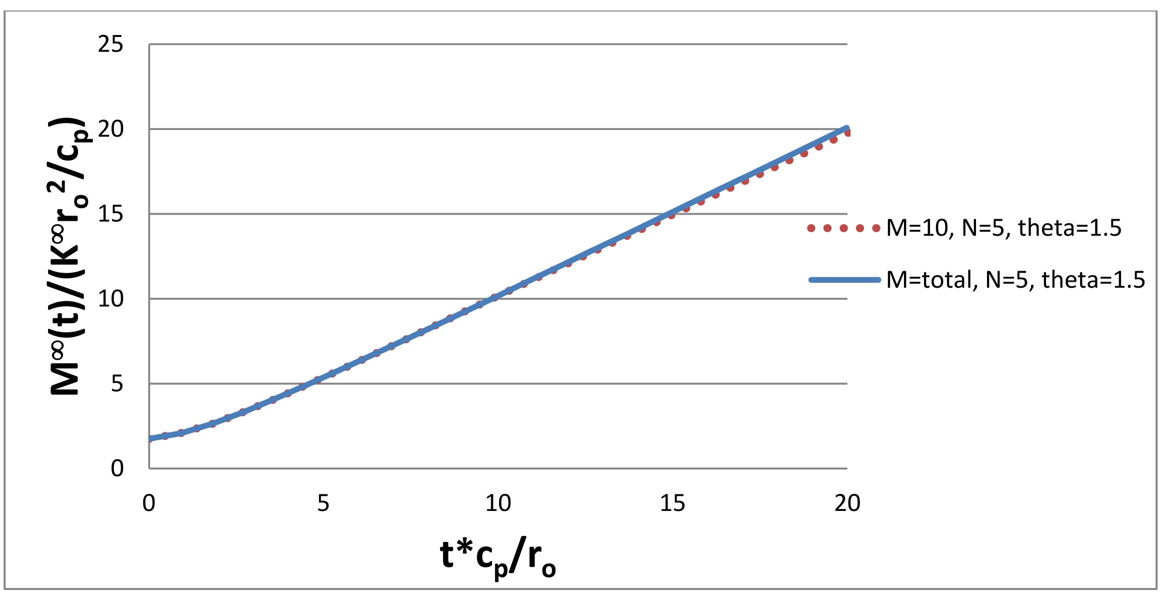


Fig.(4.11): The unit impulse response function due to radial uniform load

Now apply uniform radial triangular impulse load fig.(4.12) to the interface, the displacement for different values of theta (1, 1.5, 2) at N=5 and M=total are shown in fig.(4.13), where the results is compared with that of SIMILAR, also good agreement is achieved.

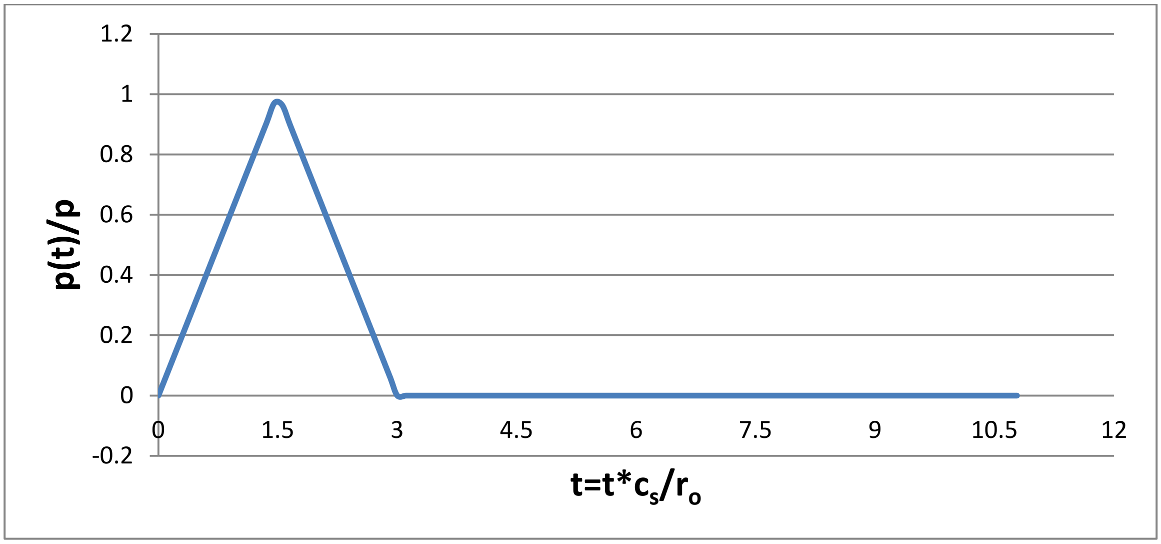


Fig.(4.12): Triangular impulse load

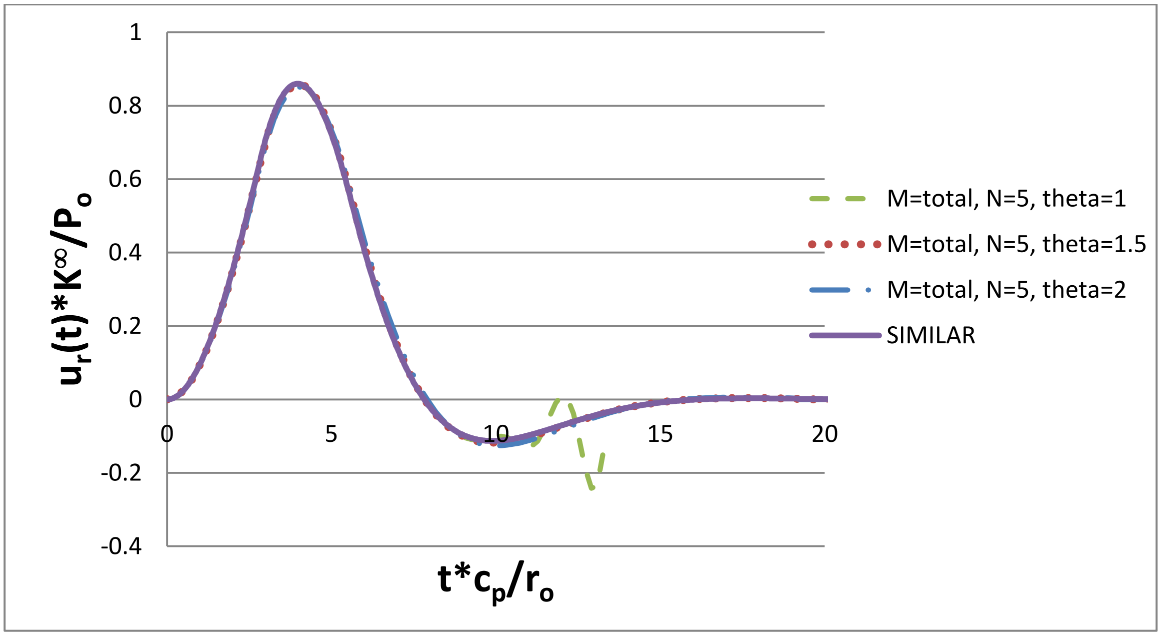


Fig.(4.13): Radial displacement due to uniform radial triangular impulse

To see the effect of truncating the time of calculation of the unit impulse the displacement for the case where, N=5, theta=1.5 and M=10 is shown in fig.(4.14) with the solution at M=total (111), where the two results agree.

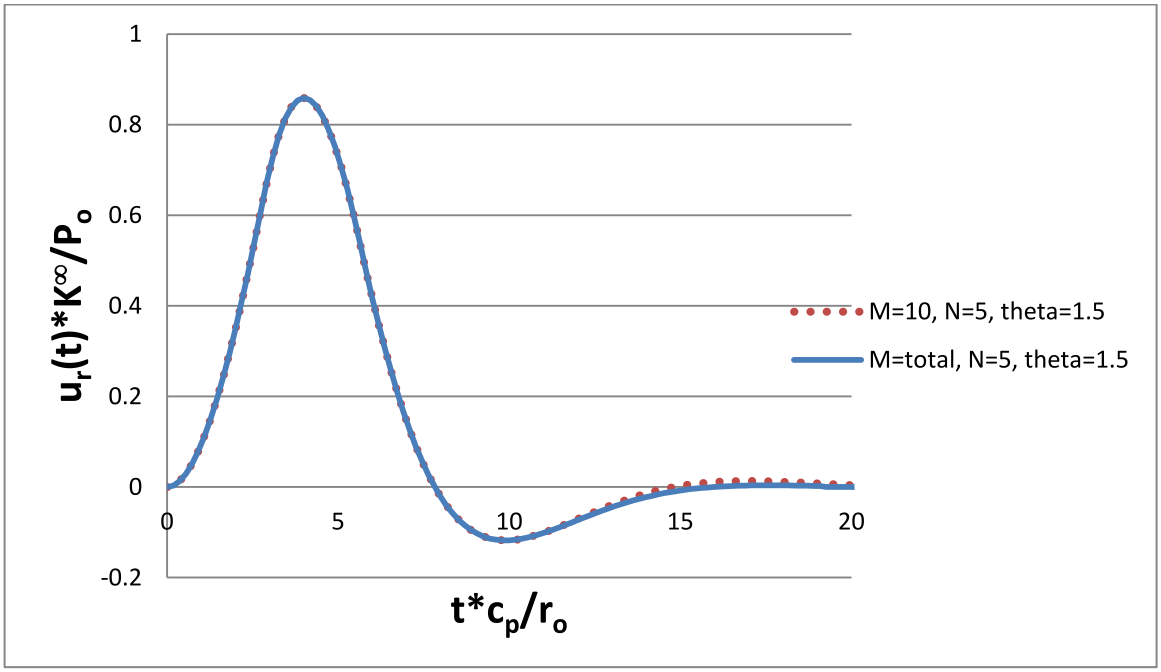


Fig.(4.14): Radial displacement due to uniform radial triangular impulse

4.4. Elastic half space subjected to vertical load (Lamb`s problem)

Here the in-plane motion of elastic half space subjected to vertical load is studied [21]. The elastic half space is shown in fig.(4.15) where, point B is the observation point, the load is a sinusoidal impulse see fig(4.16) (b is the length of the domain used in numerical calculation from the loading point, fig.(4.17)), where p(t) is given by

$p(t)=\begin{cases} p_o \sin^2(\pi t) & t \le t_o \\ 0 & t > t_o \end{cases}$, where in the analysis $p_o=10^4$N and $t_o=1$ sec

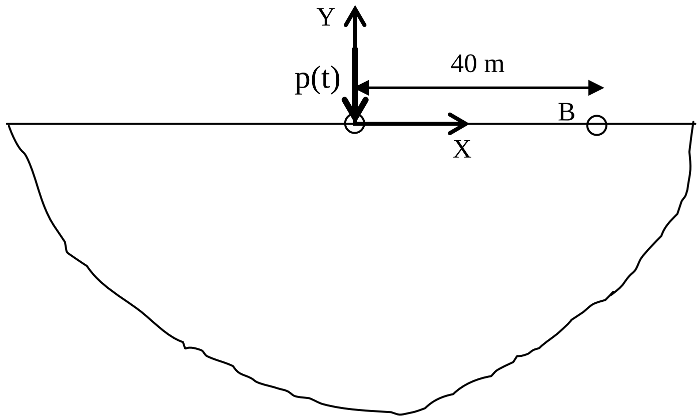


Fig.(4.15): An elastic half space subjected to vertical load

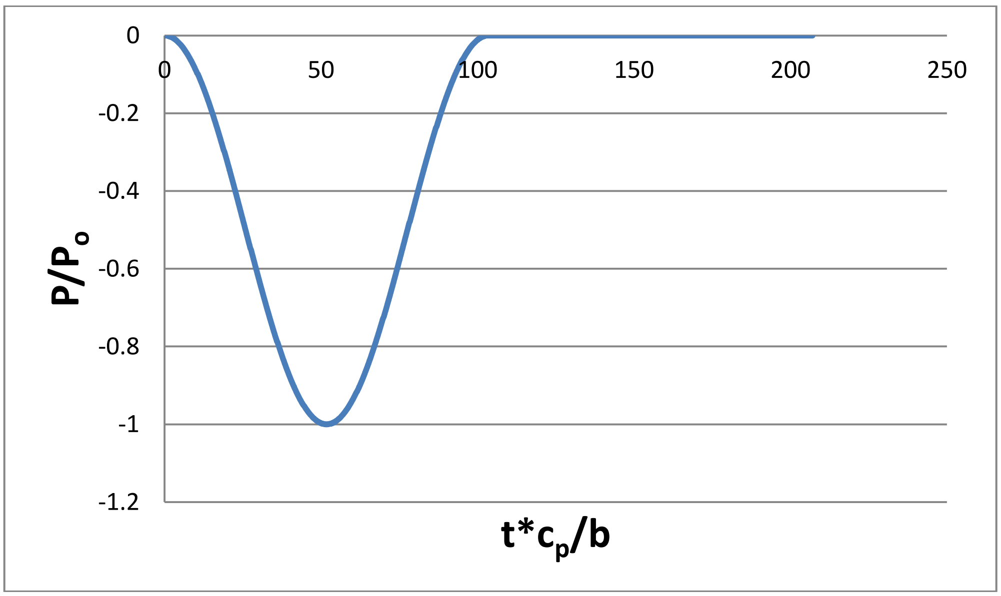


Fig.(4.16): Sinusoidal impulsive load

The equations of motion are:

$$(\lambda + G)\frac{\partial \Delta}{\partial x} + G\nabla^2 u_x = \rho \ddot{u}_x \qquad (4.44)$$

$$(\lambda + G)\frac{\partial \Delta}{\partial x} + G\nabla^2 u_y = \rho \ddot{u}_y \qquad (4.45)$$

Where, $\Delta = \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y}$, $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ and $\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}$

The boundary conditions of the problem are:

$$\sigma_{xy}(y=0) = 0 \qquad (4.46)$$

$$\sigma_{yy}(y=0) = -p(t)\delta(x) \tag{4.47}$$

The solution of the above equation in case of $p(t) = p_o\delta(t)$ is according to [20] at y=0:

$$u_y(t) = \frac{p_o c_s}{\pi G x}\begin{cases} 0 & \tau \le \chi \\ \dfrac{\left(1-2\tau^2\right)^2\sqrt{\tau^2-\chi^2}}{\left(1-2\tau^2\right)^4 + 16\tau^4\left(\tau^2-\chi^2\right)\left(1-\tau^2\right)} & \chi < \tau < 1 \\ \dfrac{\sqrt{\tau^2-\chi^2}}{\left(1-2\tau^2\right)^2 - 4\tau^2\sqrt{\left(\tau^2-\chi^2\right)\left(\tau^2-1\right)}} & \tau \ge 1 \end{cases} \tag{4.48}$$

Where, $\tau = c_s t/x$ and $\chi = c_s / c_p$

In order to get the solution for any force function $p(t)$ the convolution integral can be used as follows:

$\bar{u}_y(t) = \int_0^t p(\tau`)u_y(t-\tau`)d\tau`$, where $\bar{u}_y(t)$ is the displacement due to the force function $p(t)$

Here I have to thank Dr. Tianyun Liu (Department of Hydraulic Engineering, Tsinghua University, Beijing, China), who provided me with a FORTRAN code [25] which gives the analytical solution of Lamb`s problem

Now the numerical solution is done as follows: part of the unbounded domain is modeled using FEM (For an introduction to FEM refer to [3]) and the rest is modeled using SBFEM, the advantage of symmetry of the problem is taken in consideration so only half of the domain is modeled. As shown in fig. (4.17) a part of a variable length and depth is modeled using FEM and the rest of the unbounded domain is modeled using SBFEM. For the FE model four-noded isoparametric element is used and for the SBFE two-noded line elements is used. Here different types of mesh dimensions with different element sizes are used (the elements are square in shape), where the time duration is taken so that the wave length is equal to the element size.

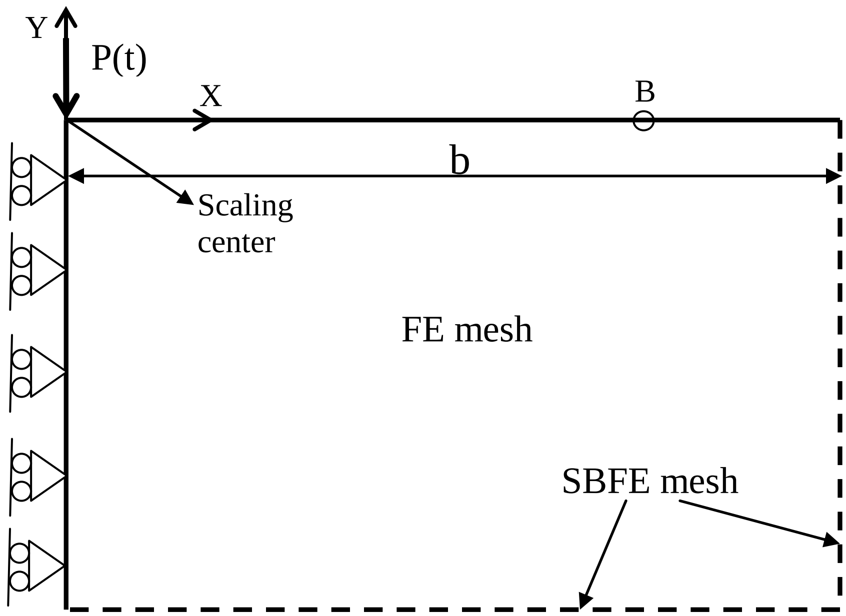


Fig.(4.17): Coupled FE/SBFE model for the elastic half space

In figures (4.18) through (4.23) below the solutions at point B (at x=40 m) for the vertical and horizontal displacement for mesh dimensions 40X20 m, 40X10 m and 40X5 m for different element sizes (at θ=2, N=2 and M=total) are shown with the exact solution (in case of vertical), where h is the element dimension. The difference between the maximum values of the displacement for the calculated cases is shown in the tables below:

Mesh dimension 40X20 m					
element size (m)	u_y*G/P	diff. %	error %	u_x*G/P	diff. %
10X10	-6.670E-01		2.817	-7.578E-02	
5X5	-6.824E-01	-2.259	0.571	-8.138E-02	-6.876
2.5X2.5	-6.867E-01	-0.627	-0.056	-8.287E-02	-1.802

Mesh dimensions 40X10 m					
element size (m)	u_y*G/P	diff. %	error %	u_x*G/P	diff. %
10X10	-6.114E-01		10.916	-6.048E-02	
5X5	-6.634E-01	-7.829	3.349	-7.526E-02	-19.640
2.5X2.5	-6.818E-01	-2.700	0.667	-8.122E-02	-7.334

Mesh dimensions 40X5 m					
element size (m)	u_y*G/P	diff. %	error %	u_x*G/P	diff. %
5X5	-6.003E-01		12.538	-5.912E-02	
2.5X2.5	-6.609E-01	-9.168	3.710	-7.490E-02	-21.065
1.25X1.25	-6.811E-01	-2.968	0.765	-8.112E-02	-7.670

Where, diff. %=(present result-previous result)*100/minimum of the two results, and error %=(present result-exact result)*100/minimum of the two results. (Exact uy*G/P=-6.86E-01)

In figures (4.24) through (4.29) below the solutions at point B (at x=40 m) for the vertical and horizontal displacement for element sizes 10X10 m, 5X5 m and 2.5X2.5 m for different mesh dimensions (at θ=2, N=2 and M=total) are shown with the exact solution (in case of vertical). The difference between the maximum values of the displacement for the calculated cases is shown in the tables below:

Element size 10X10 m					
Mesh dim. (m)	u_y*G/P	diff. %	error %	u_x*G/P	diff. %
40X10	-6.114E-01		10.916	-6.048E-02	
40X20	-6.670E-01	-8.333	2.817	-7.578E-02	-20.194
40X40	-6.820E-01	-2.200	0.632	-8.162E-02	-7.152

Element size 5X5 m					
Mesh dim. (m)	u_y*G/P	diff. %	error %	u_x*G/P	diff. %
40X5	-6.003E-01		12.538	-5.912E-02	
40X10	-6.634E-01	-9.507	3.349	-7.526E-02	-21.447
40X20	-6.824E-01	-2.794	0.571	-8.138E-02	-7.519
40X40	-6.87E-01	-0.596	-0.025	-8.296E-02	-1.899
30X40	-6.86E-01	0.140	0.116	-8.259E-02	0.439
40X20	-6.82E-01		0.571	-8.14E-02	

60X20	-6.82E-01	0.123	0.694	-8.11E-02	0.383
80X20	-6.81E-01	0.088	0.781	-8.10E-02	0.025

Element size 2.5X2.5 m					
Mesh dim. (m)	uy*G/P	diff. %	error %	ux*G/P	diff. %
40X5	-6.609E-01		3.710	-7.490E-02	
40X10	-6.818E-01	-3.063	0.667	-8.122E-02	-7.782
40X15	-6.855E-01	-0.548	0.120	-8.244E-02	-1.483
40X20	-6.87E-01	-0.176	-0.056	-8.287E-02	-0.523

It is clear from the charts below and from the tables above that:

1. as the depth of the mesh increases the results converges
2. the horizontal displacement are affected by the mesh dimensions and element size more than the vertical displacement.
3. increasing the dimensions in x direction does not affects the results.
4. A good agreement is achieved in case of vertical displacement

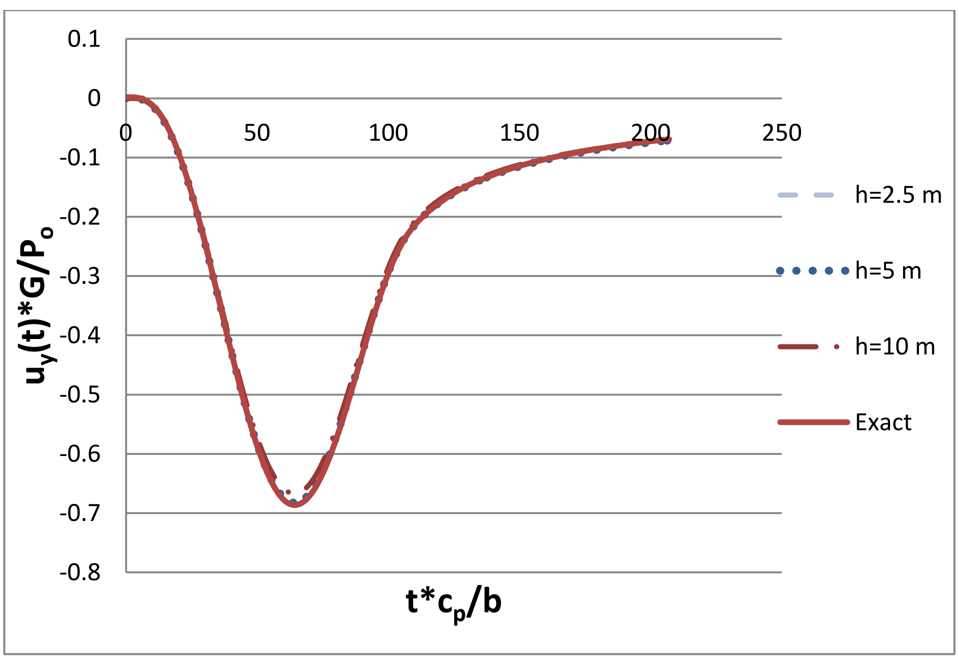


Fig.(4.18): Vertical displacement of point B for mesh dimensions of 40X20 m

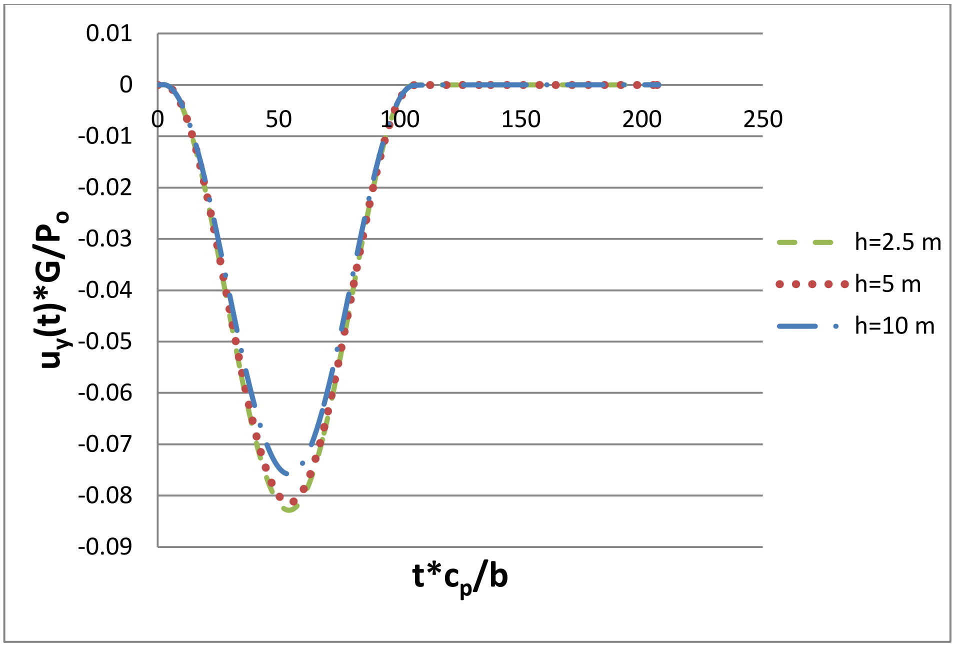


Fig.(4.19): Horizontal displacement of point B for mesh dimensions of 40X20m

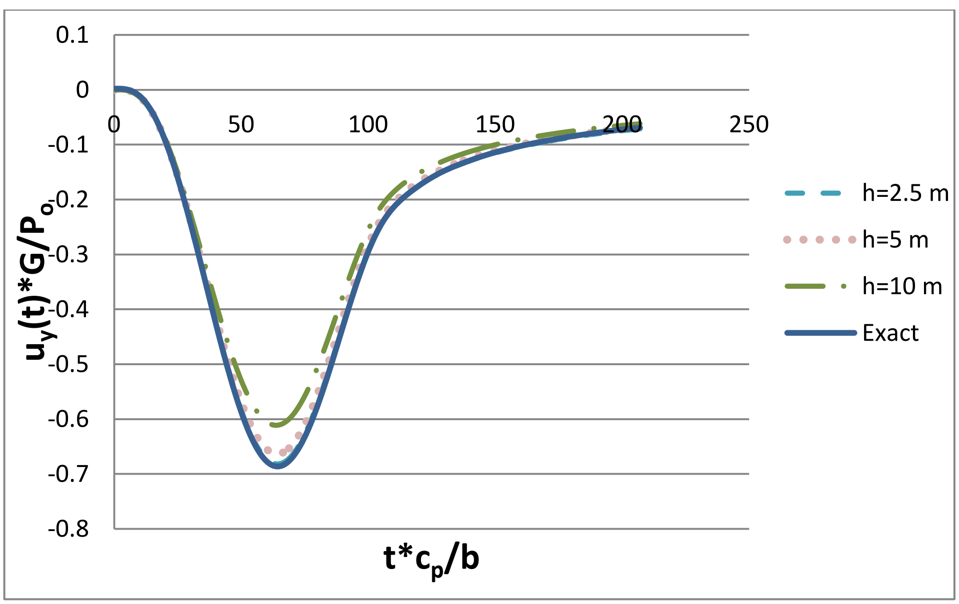


Fig.(4.20): Vertical displacement of point B for mesh dimensions of 40X10m

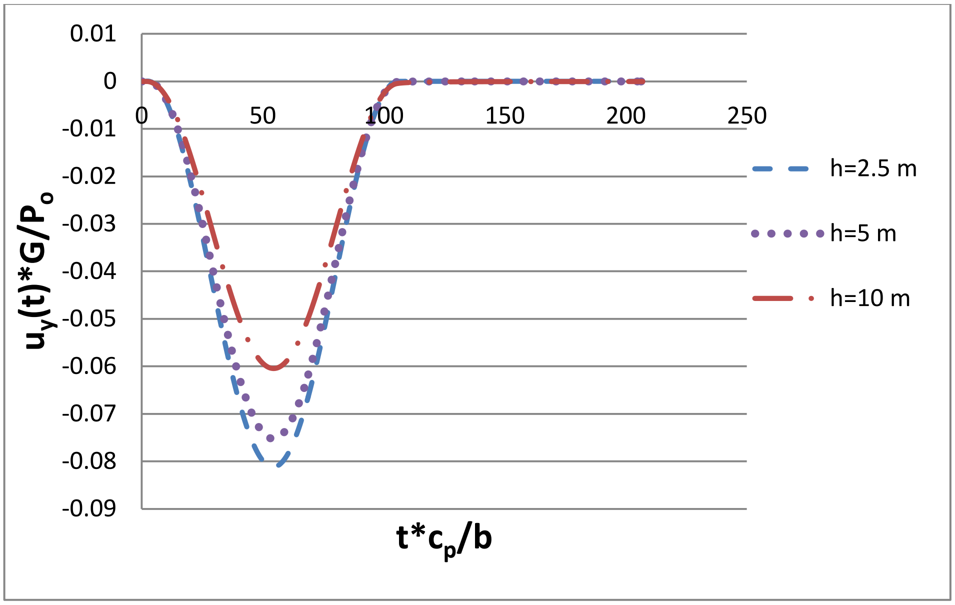


Fig.(4.21): Horizontal displacement of point B for mesh dimensions of 40X10m

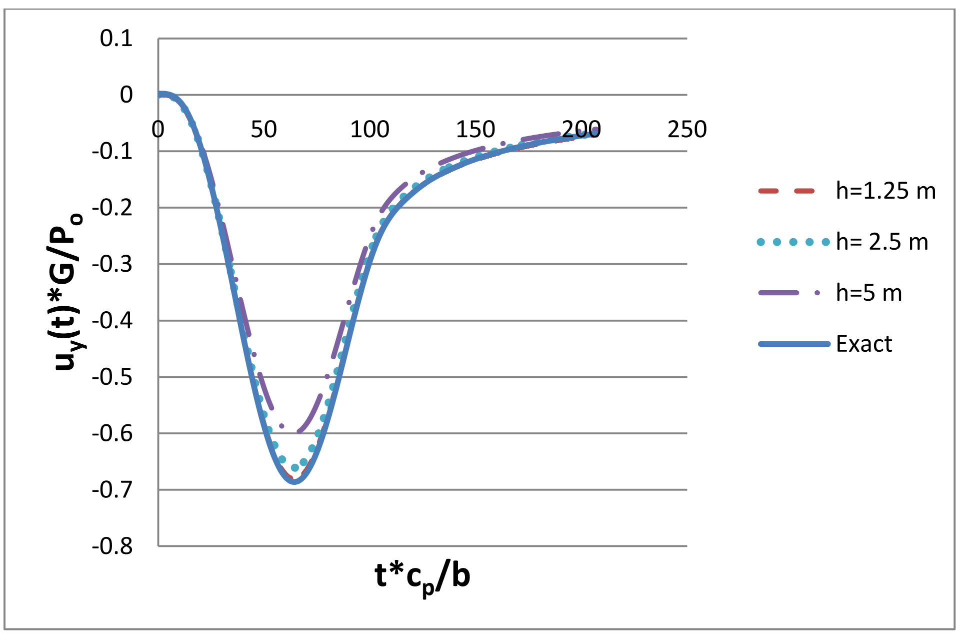


Fig.(4.22): Vertical displacement of point B for mesh dimensions of 40X5m

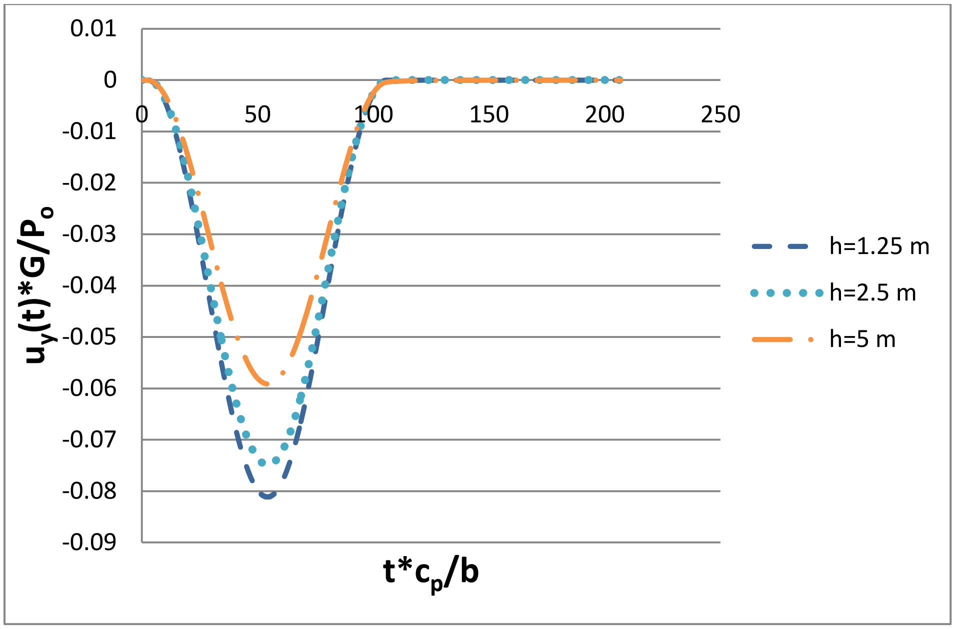


Fig.(4.23): Horizontal displacement of point B for mesh dimensions of 40X5m

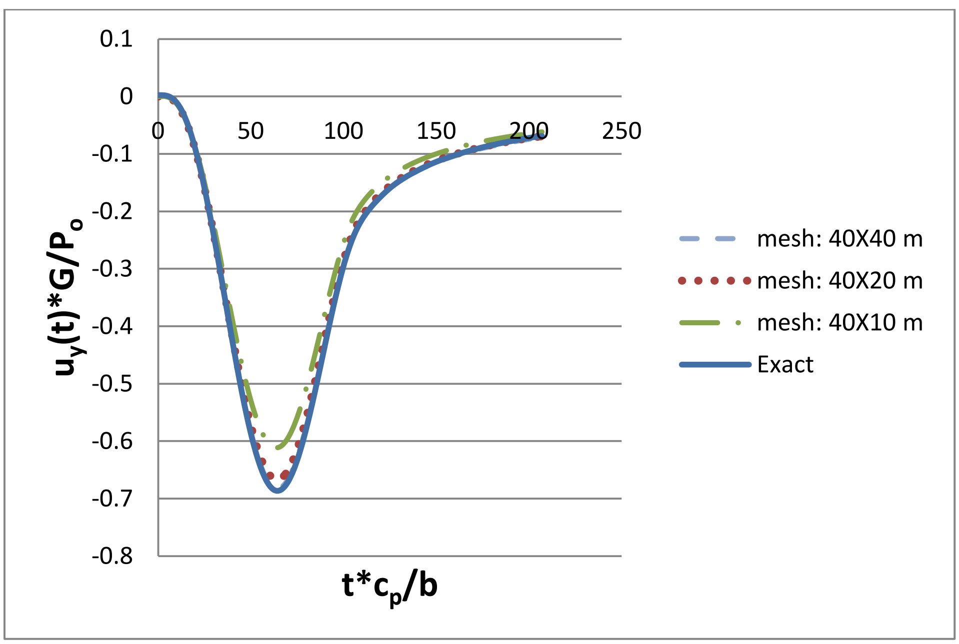


Fig.(4.24): Vertical displacement of point B for element size of 10 m

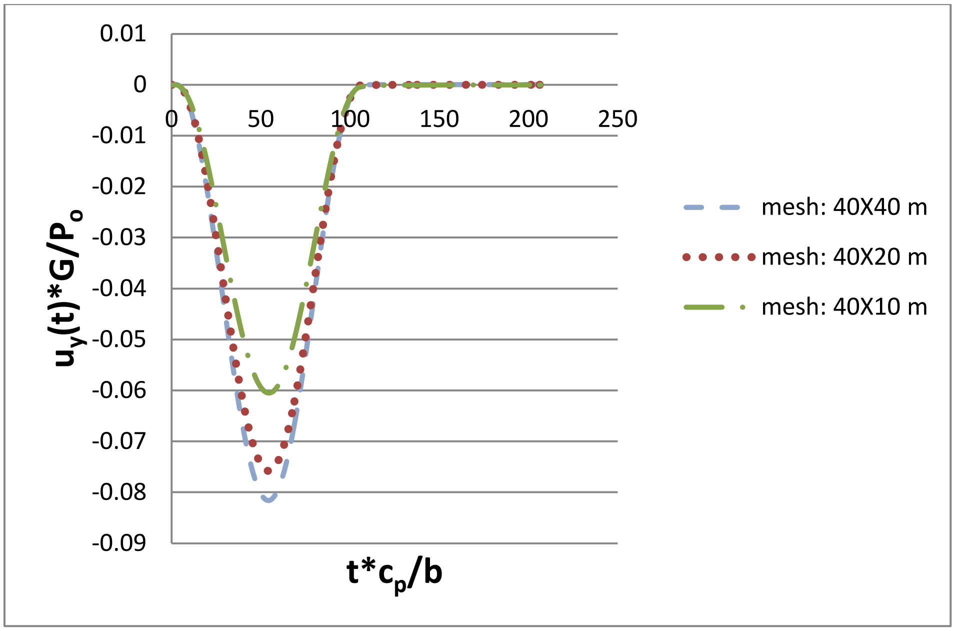


Fig.(4.25): Horizontal displacement of point B for element size of 10 m

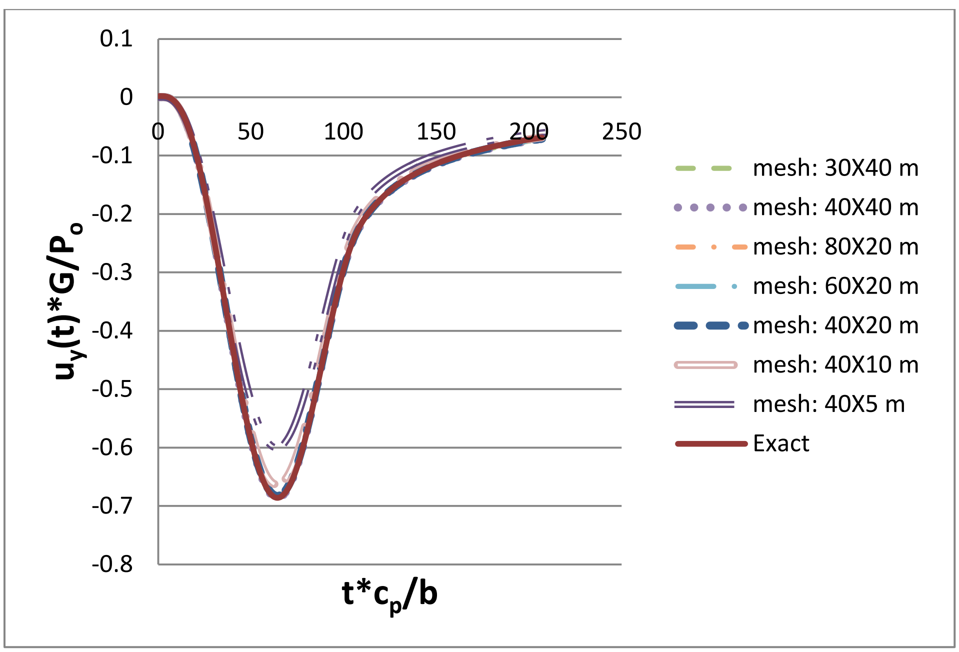


Fig.(4.26): Vertical displacement of point B for element size of 5 m

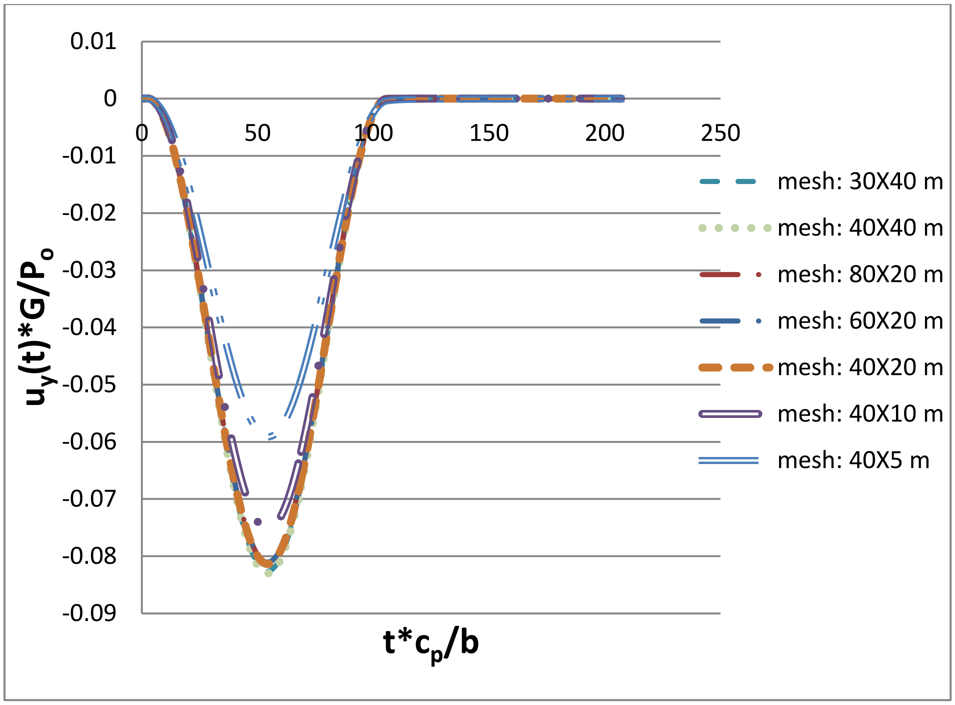


Fig.(4.27): Horizontal displacement of point B for element size of 5 m

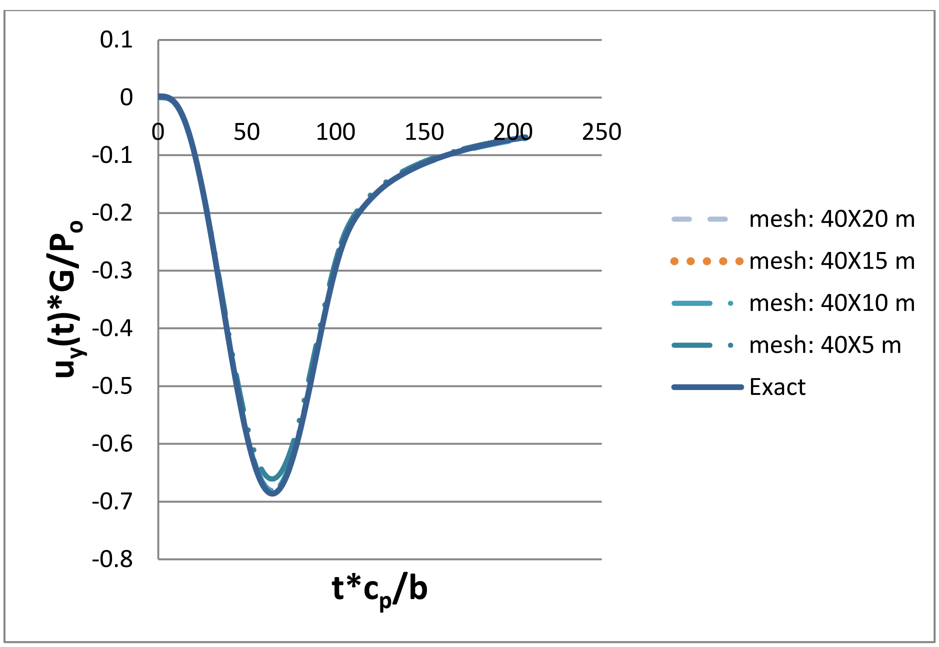


Fig.(4.28): Vertical displacement of point B for element size of 2.5 m

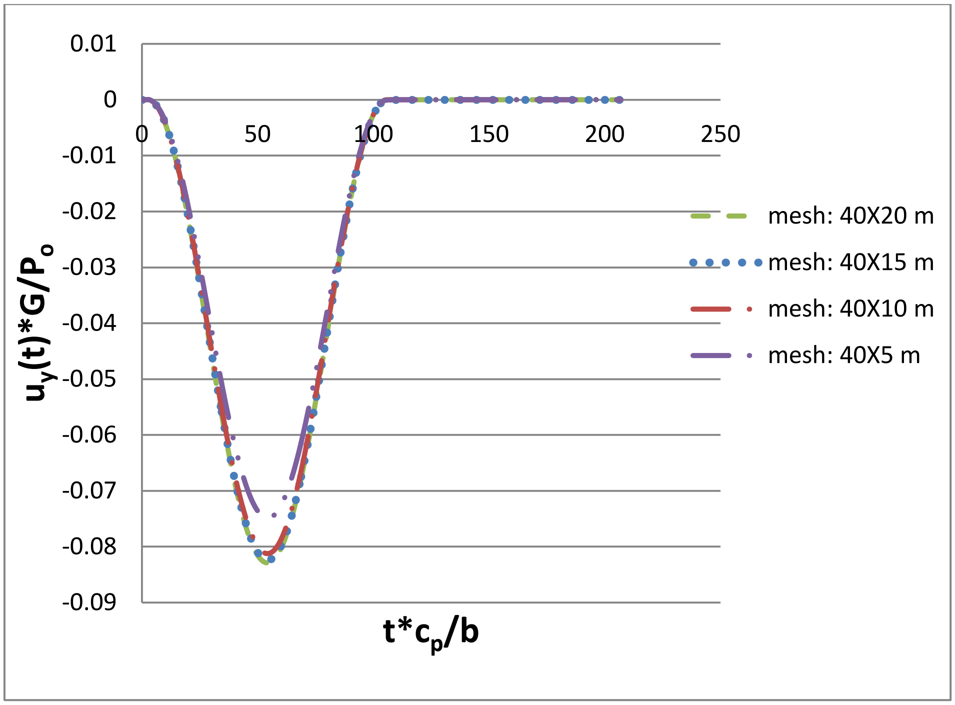


Fig.(4.29): Horizontal displacement of point B for element size of 2.5 m

To test the efficiency of the method in case of large N the problem is solved in case of mesh dimensions 40X20 m and element size 5 m for N=5, M=total and different θ (2, 1.5 and 1), and the results is compared to the case of N=2, M=total and θ=2, the results for the vertical and horizontal displacement of point B are shown below in fig.(4.30) and fig.(4.31), respectively.

It is clear from the figures below that there is an agreement with the case of N=2, except for the case of θ=1 where the solution is unstable.

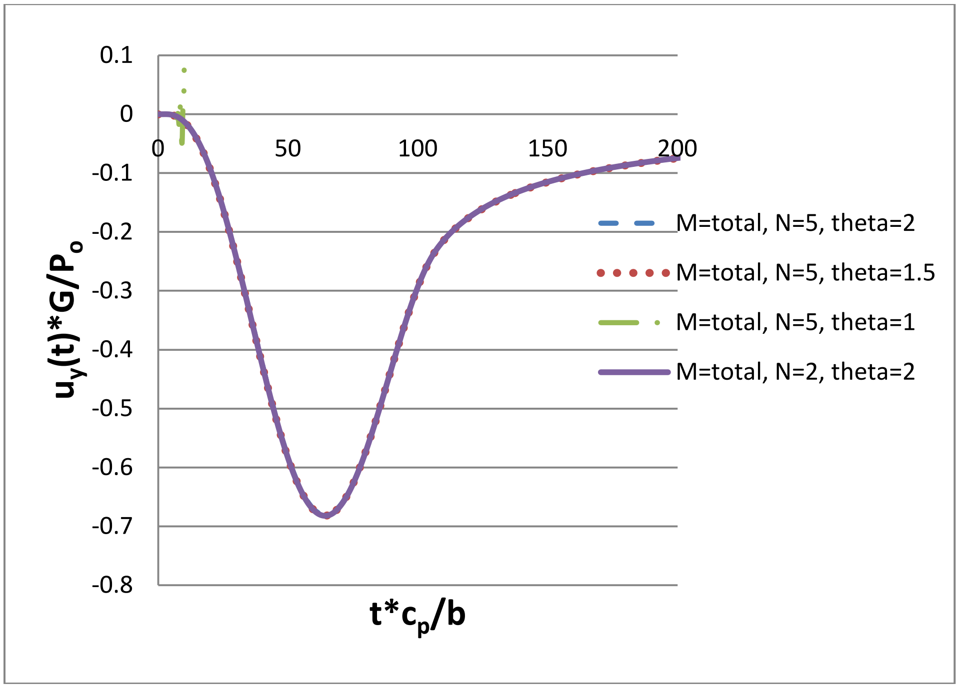


Fig.(4.30): Vertical displacement of point B for mesh dimension 40X20 m and element size of 5 m

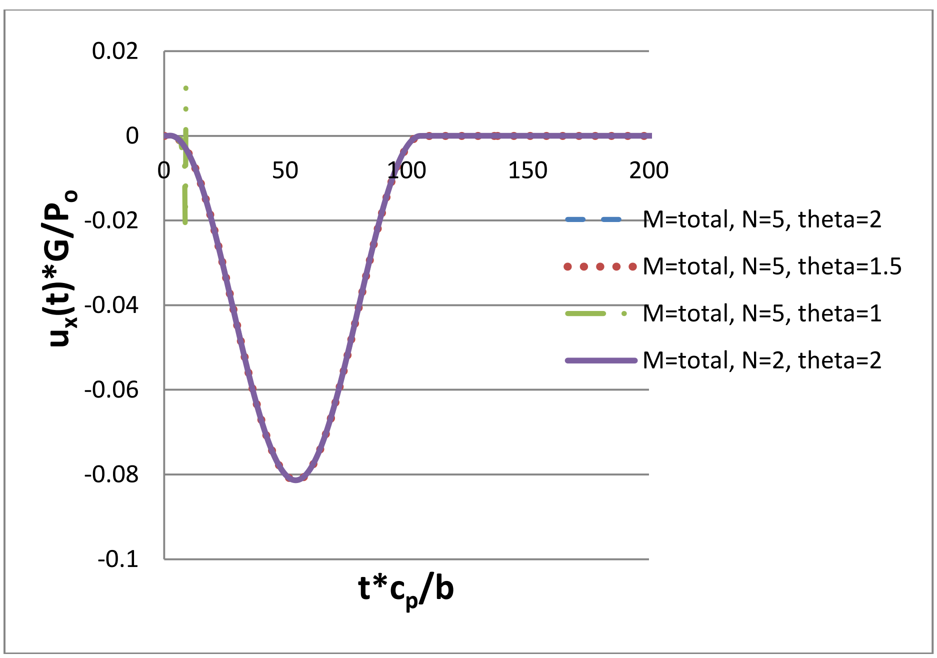


Fig.(4.31): Horizontal displacement of point B for mesh dimension 40X20 m and element size of 5 m

To see the effect of the truncation of the time of calculation of the unit impulse the problem is solved in case of mesh dimensions 40X20 m and element size 5 m for N=5, M=270 and θ=1.5, and the results is compared to the case of N=2, M=total (2000) and θ=1.5, the results for the vertical and horizontal displacement of point B are shown below in fig. (4.32) and fig.(4.33), respectively, where good agreement is achieved.

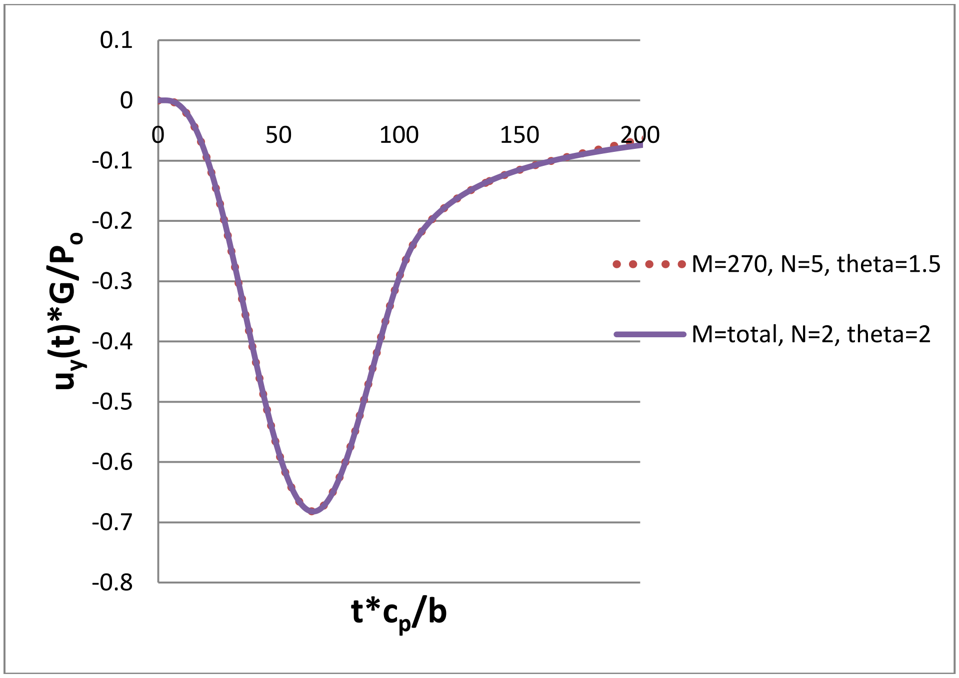


Fig.(4.32): Vertical displacement of point B for mesh dimension 40X20 m and element size of 5 m

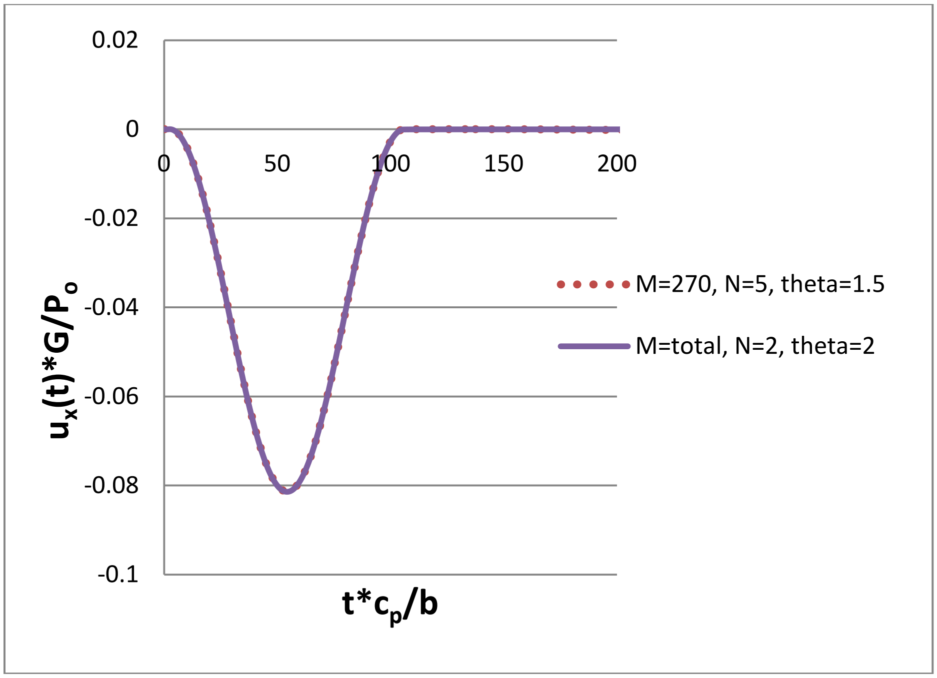


Fig.(4.33): Horizontal displacement of point B for mesh dimension 40X20 m and element size of 5 m

Chapter 5: Practical problems

5.1 Introduction

In this chapter some practical problems are solved, where the unbounded part is solved using SBFEM and the bounded part is solved using FEM. The validation of the results is done using the FE program SAP2000 ver.16 [24], where an extended FE mesh is used (For an introduction to FEM refer to [3]).

5.2 A concrete dam rests on elastic half space subjected to vertical load

Here the in plane response of a concrete dam rests on soil is studied [21]. The geometry of the dam is shown in fig. (5.1). A concentrated load (see fig.(4.16), where c_s is the shear wave velocity of the soil)) is applied at the top of the dam as shown in the figure, where:

$p(t) = \begin{cases} p_o \sin^2(\pi t) & t \le t_o \\ 0 & t > t_o \end{cases}$, in the analysis $p_o = 10^4$ N/m and $t_o = 1$ sec

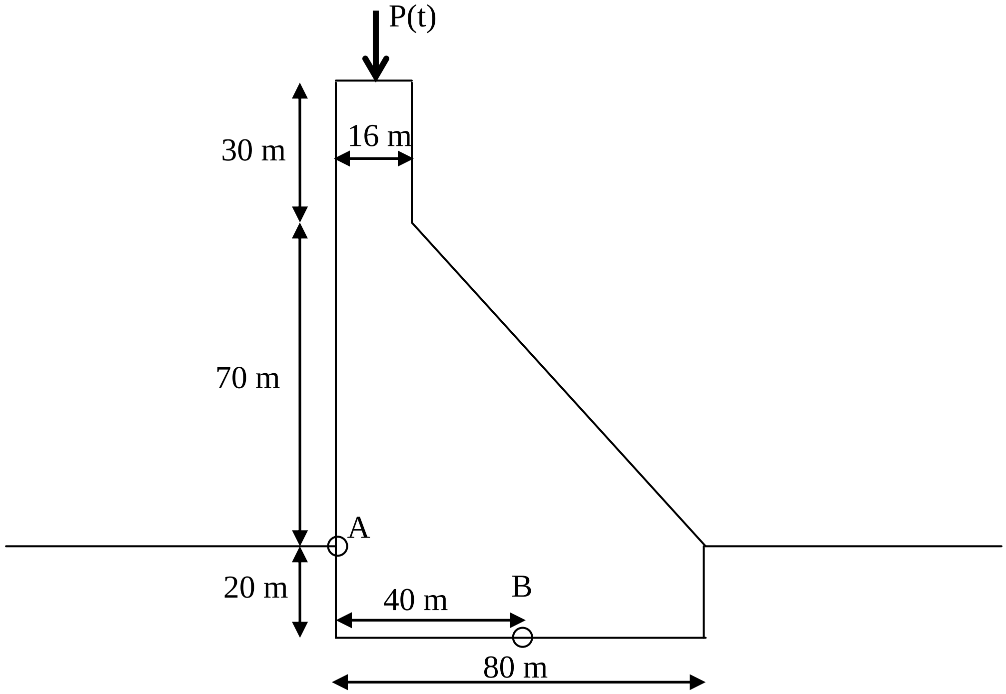


Fig.(5.1): The concrete dam geometry

The material properties are:

For soil:

$E=2.4*10^{10}$ N/m^2, $\nu=1/3$ and $\rho=2100$ kg/m^3

For the concrete tunnel:

$E=4.956*10^{10}$ N/m^2, $\nu=0.18$ and $\rho=2400$ kg/m^3

The problem here is a plane strain problem, it is modeled as shown in fig.(5.2), where the concrete dam is modeled using FE (280 four-noded isoparametric element) and for the soil part of it is modeled using FE (four-noded isoparametric element) and the rest using SBFE (two-noded line element). The analysis is done with time step equals $0.031 b_o / c_s$, where c_s is the shear wave velocity of the soil and the FE size is 10X10 m.

The problem is solved for five different dimensions of soil domain:

1. b=40 m and d=20 m
2. b=40 m and d=40 m

3. b=60 m and d=40 m
4. b=80 m and d=60 m
5. b=100 m and d=80 m

where , for all the above cases the M=total (i.e., the total time is used to calculate the unit impulse function), N=2 and θ=2.

The results are shown for the vertical and horizontal displacements of point A and B in figures (5.4) through (5.7), where b_o=40 m (for all the charts the wave velocities and the elastic constants are that of the soil).

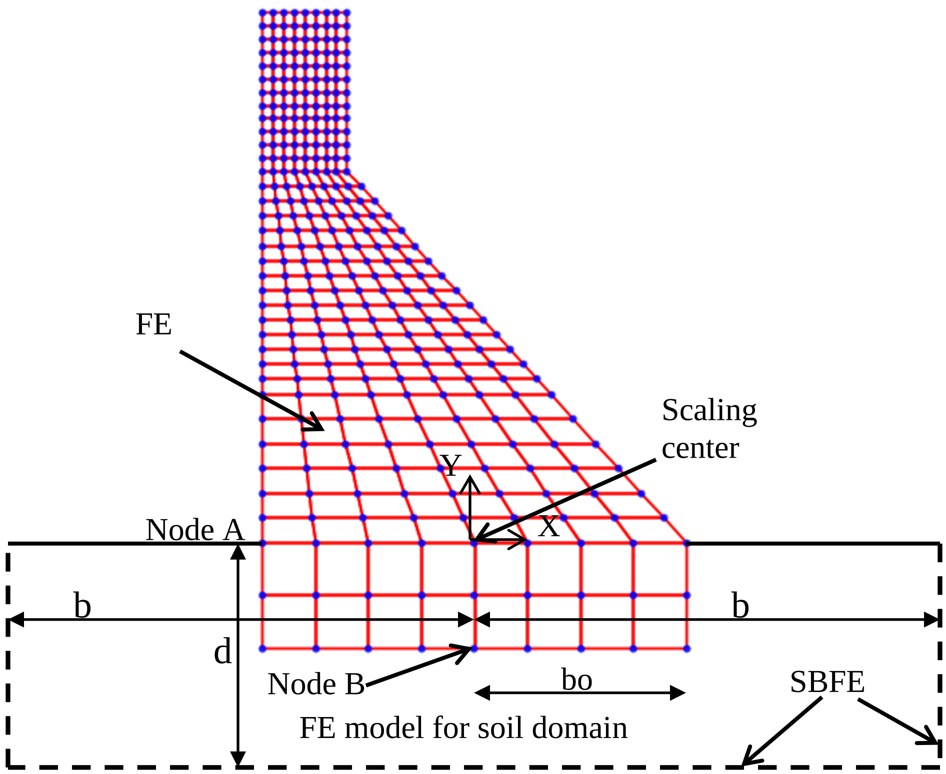


Fig.(5.2): The concrete dam model

To validate the answer, since there is no exact solution for this problem a numerical model is done using extended FE mesh (using SAP2000 program ver.16) with variable element dimensions as shown in fig.(5.3), the mesh dimensions is 5500X2960 m, the boundary is restraint in x and y directions with rigid supports. The output is also shown in figures (5.4) through (5.7)

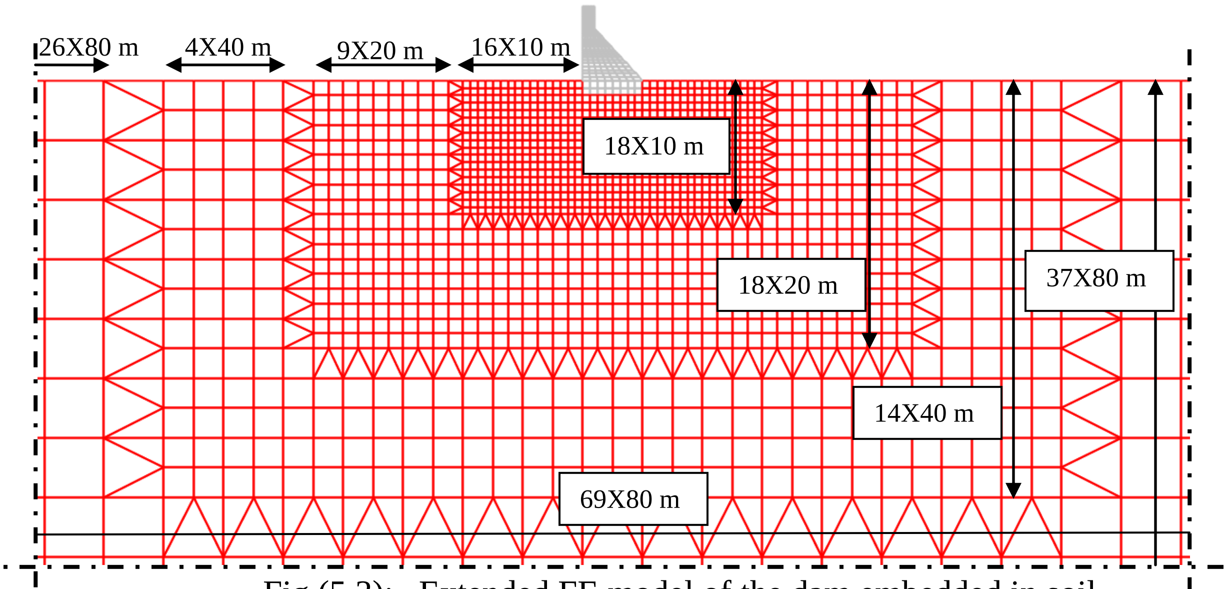


Fig.(5.3): Extended FE model of the dam embedded in soil

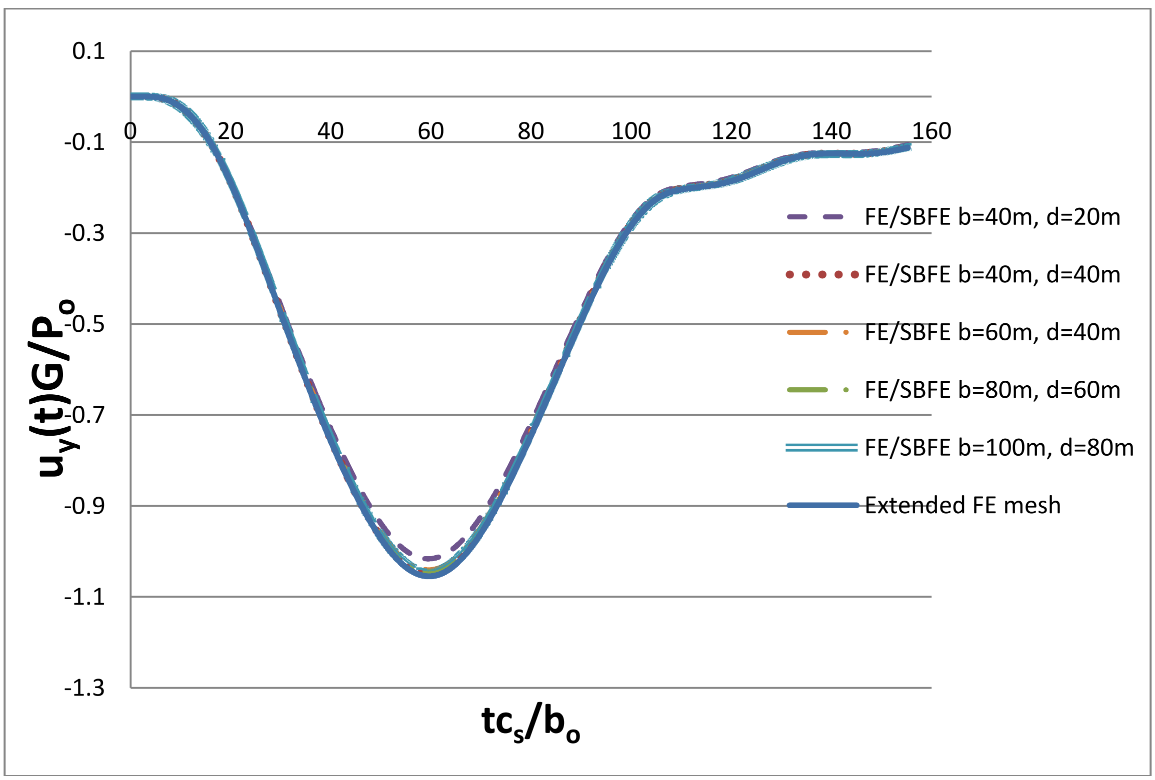


Fig.(5.4): Vertical displacement of point A

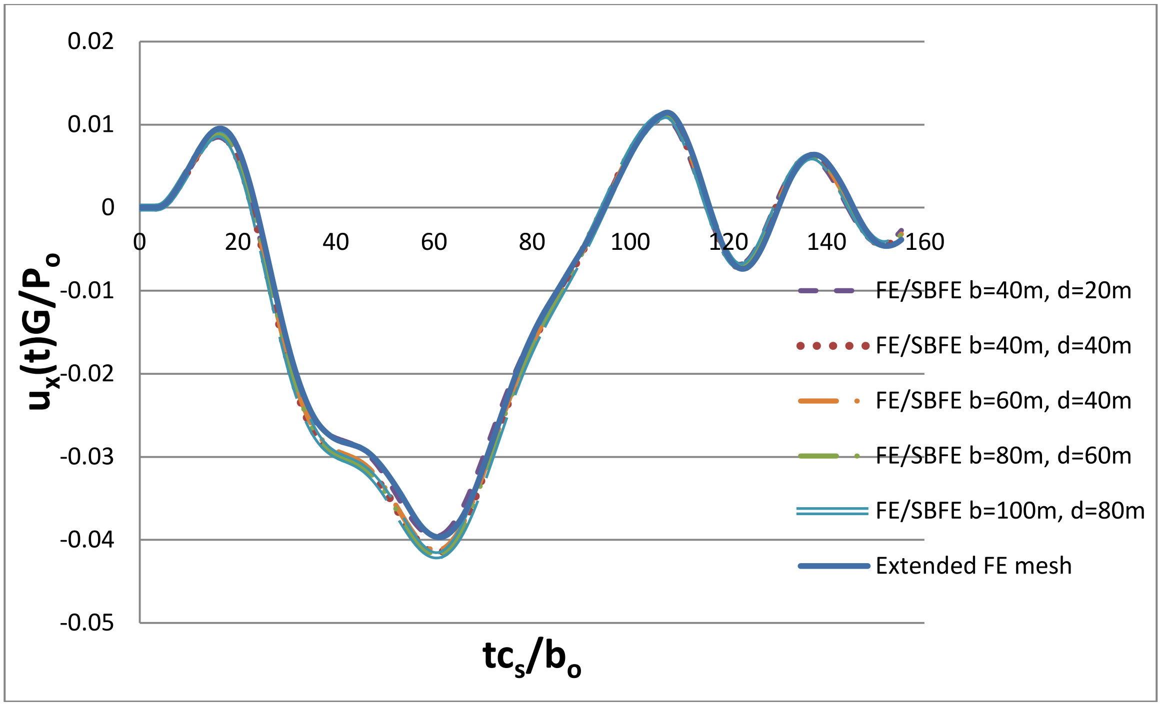


Fig.(5.5): Horizontal displacement of point A

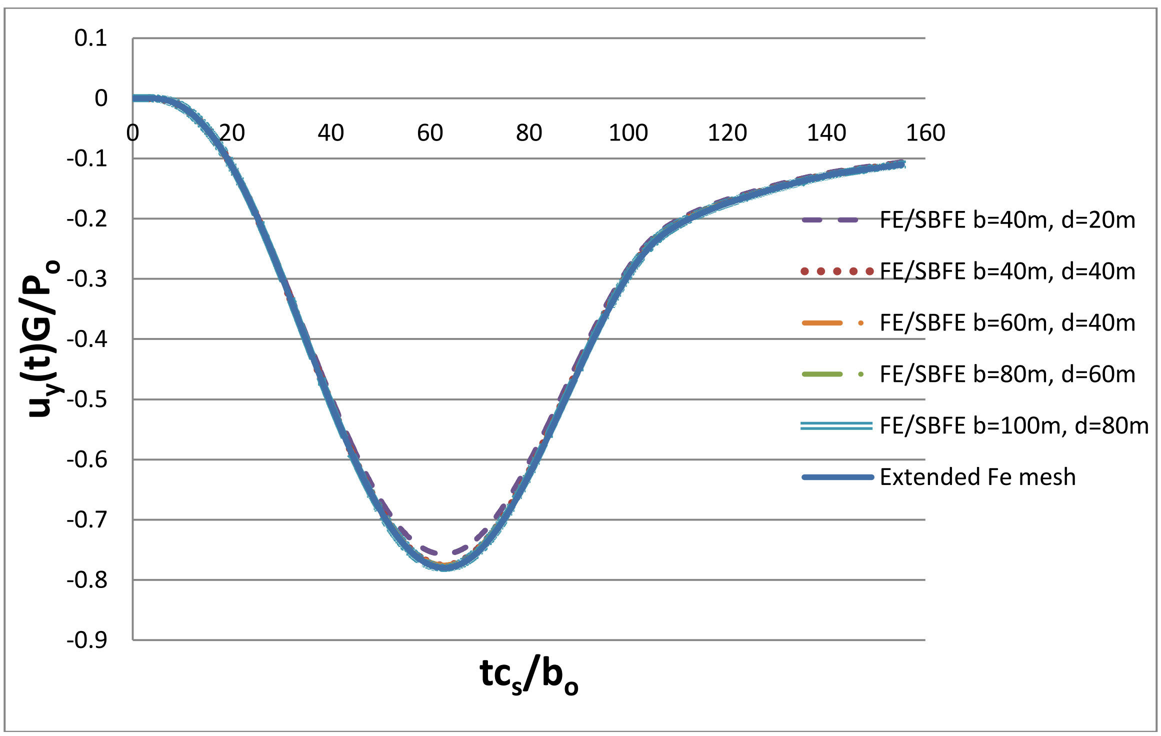


Fig.(5.6): Vertical displacement of point B

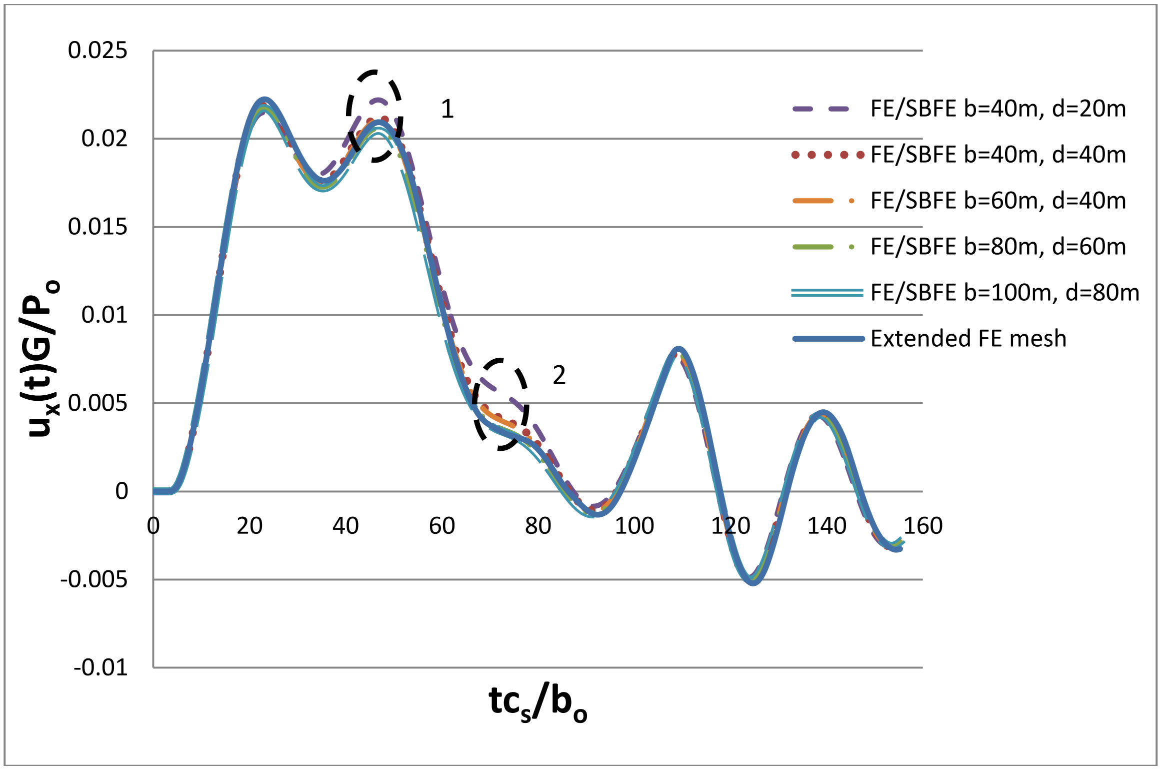


Fig.(5.7): Horizontal displacement of point B

Here is a comparison between the maximum values in each case:

For point A

b and d (m)	u_y*G/P	diff.%	diff.% w.r.t. FE	u_x*G/P	diff.%	diff.% w.r.t. FE
40 and 20	-1.016		3.779	-0.040		0.177
40 and 40	-1.050	-3.286	0.477	-0.042	-4.995	-4.810
60 and 40	-1.041	0.877	1.358	-0.041	0.821	-3.956
80 and 60	-1.045	-0.434	0.920	-0.042	-1.116	-5.115
100 and 80	-1.047	-0.127	0.792	-0.042	-0.344	-5.477

Where, for the FE model uy*G/P= -1.055 and ux*G/P= -0.040

For point B (vertical displacement):

b and d (m)	u_y*G/P	diff.%	diff.% w.r.t. FE
40 and 20	-0.757		2.999
40 and 40	-0.776	-2.509	0.478
60 and 40	-0.775	0.093	0.571
80 and 60	-0.779	-0.430	0.141
100 and 80	-0.780	-0.141	0.000

Where, for the FE model uy*G/P= -0.780

For the horizontal displacement of point B (at the maximum differences in the chart 1 and 2):

b and d (m)	u_x*G/P (1)	diff.%	diff.% w.r.t. FE	u_x*G/P (2)	diff.%	diff.% w.r.t. FE
40 and 20	0.022		6.021	0.012		*11.665*
40 and 40	0.021	-7.820	-1.697	0.010	*-16.224*	-4.082
60 and 40	0.020	-0.732	-2.441	0.010	-1.592	-5.739
80 and 60	0.021	2.825	0.375	0.010	6.013	0.258
100 and 80	0.021	0.857	1.236	0.011	2.353	2.617

Where, for the FE model u_x*G/P (1) = 0.021 and u_x*G/P (2) = -0.010

It is clear from the figures above and the results in the tables that:

1. Increasing the depth of the FE affects the convergence of the results than increasing the width.
2. The difference between the use of the SBFE only to model the soil and the use of the coupled FE/SBFE to model it is not big. (For the differences in italics in the above table it occurs at small displacements i.e. of less importance)
3. There is also a good agreement between the results of the extended FE mesh and the coupled FE/SBFE mesh.
4. For the value named (2) in fig.(5.7) where the difference from the finite element is 11.7% the displacement value is not significant since it is small.

In order to test the efficiency of the method for large N, the problem is solved for N=5, M=total and for different values of θ (1, 1.5 and 2), the case solved will be for b=40 m and d=20 m. The results are shown in figures (5.8) through (5.11) for the vertical and horizontal displacements for points A and B, respectively, compared with the results of N=2, M=total and θ=2.

Good agreement between the results can be observed, except for the case of θ=1 where the solution is unstable.

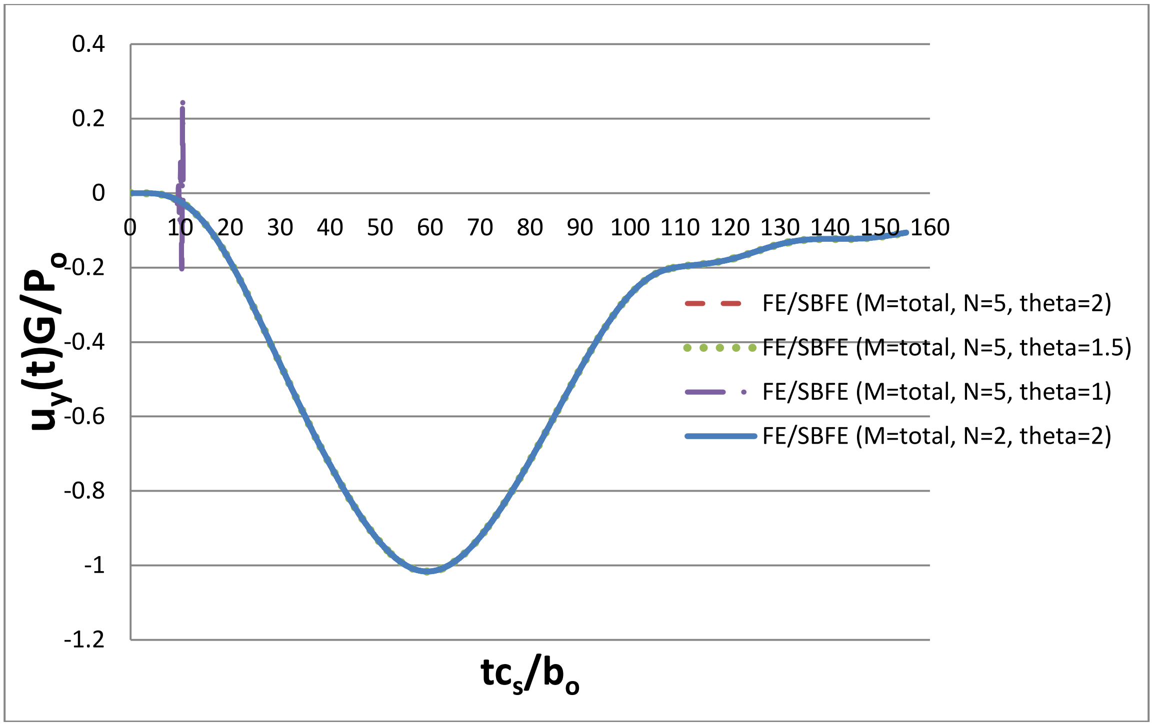


Fig.(5.8): Vertical displacement of point A for N=5 and N=2

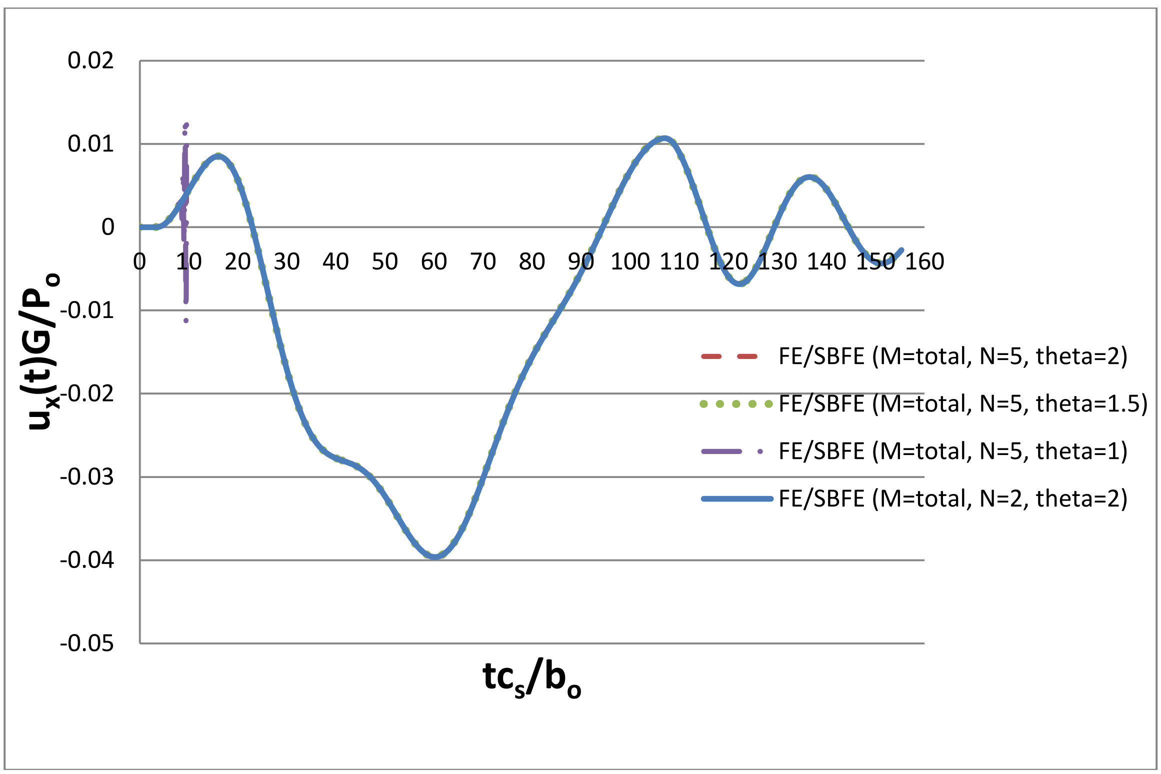


Fig.(5.9): Horizontal displacement of point A for N=5 and N=2

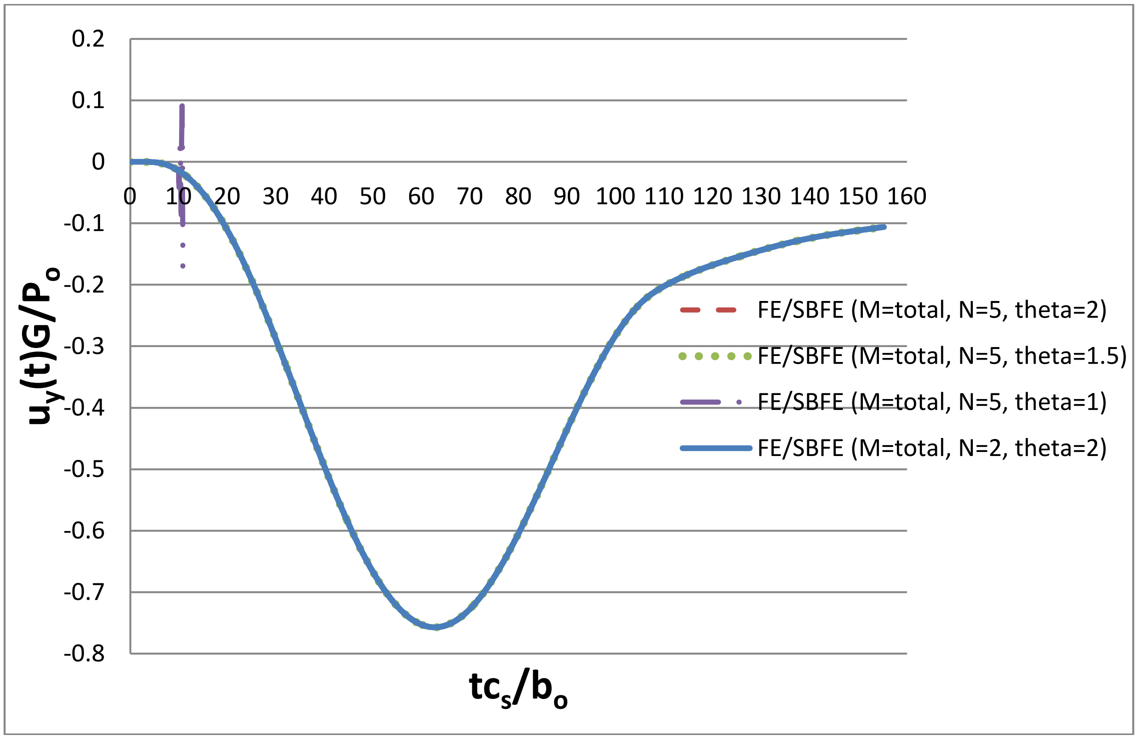


Fig.(5.10): Vertical displacement of point B for N=5 and N=2

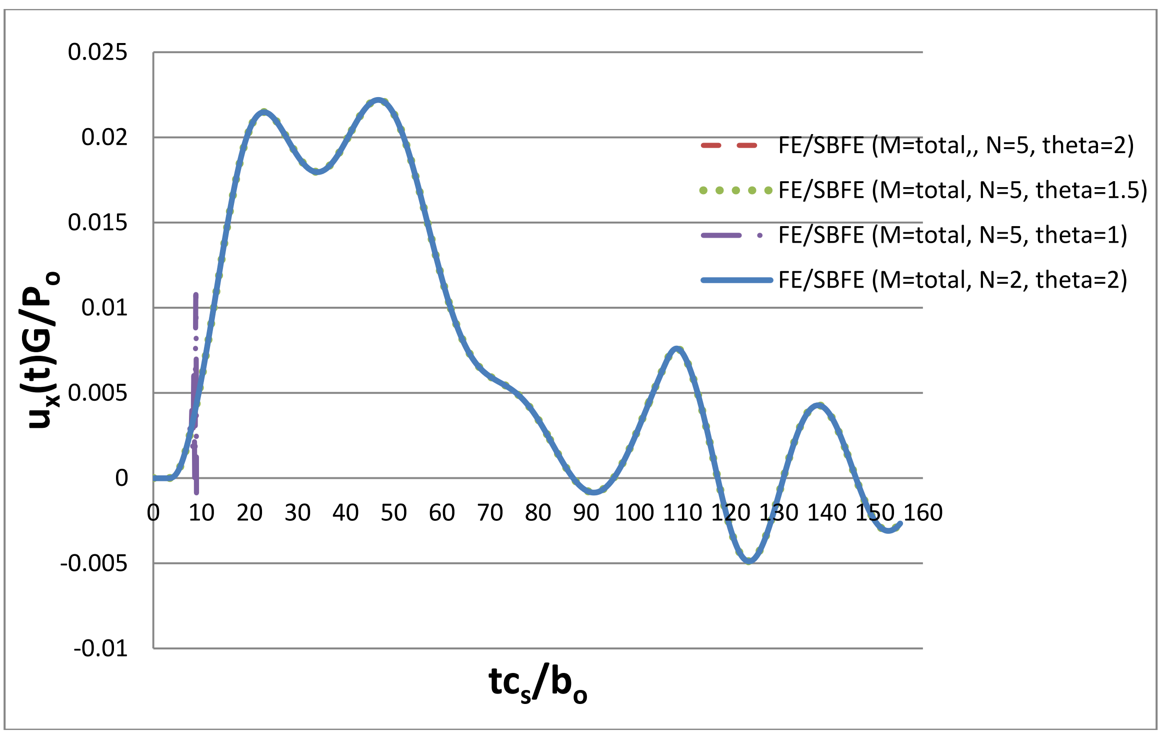


Fig.(5.11): Horizontal displacement of point B for N=5 and N=2

To see the effect of the truncation of the time of calculation of the unit impulse the problem is solved for M=650, N=5, and for different values of θ=1.5, the case solved will be for b=40 m and d=20 m. The results are shown in figures (5.12) through (5.15) for the vertical and horizontal displacements for points A and B, respectively, compared with the results of N=2, M=total (2500) and θ=2, good agreement can be observed.

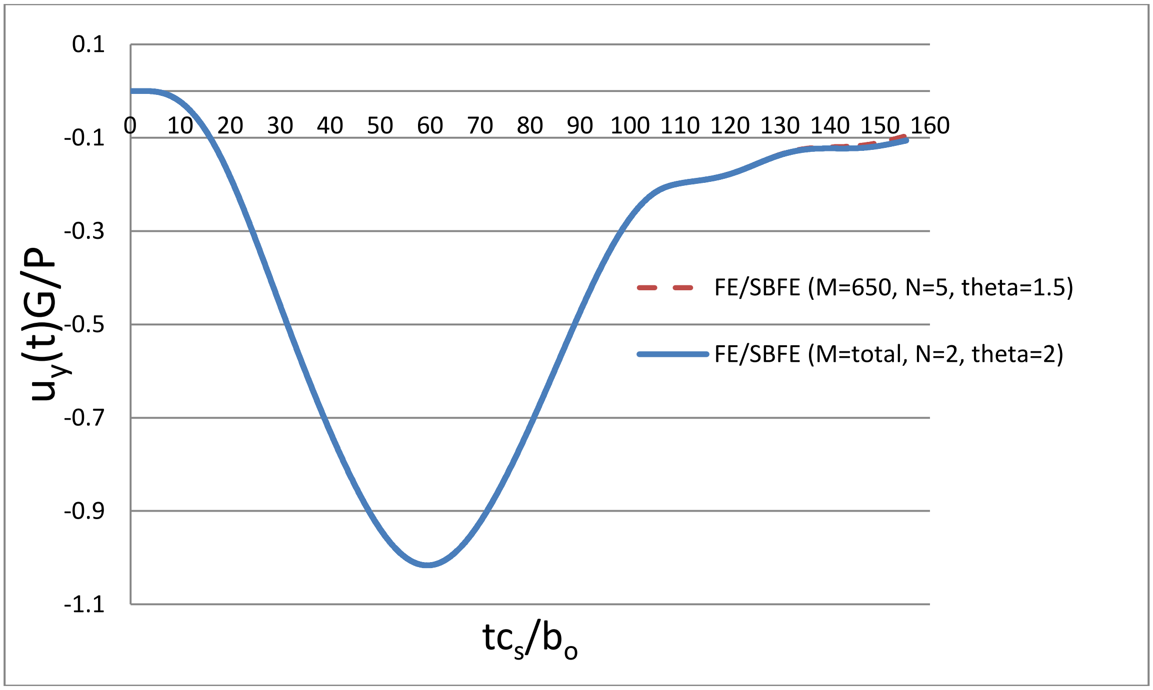


Fig.(5.12): Vertical displacement of point A for N=5, M=650 and N=2, M=total

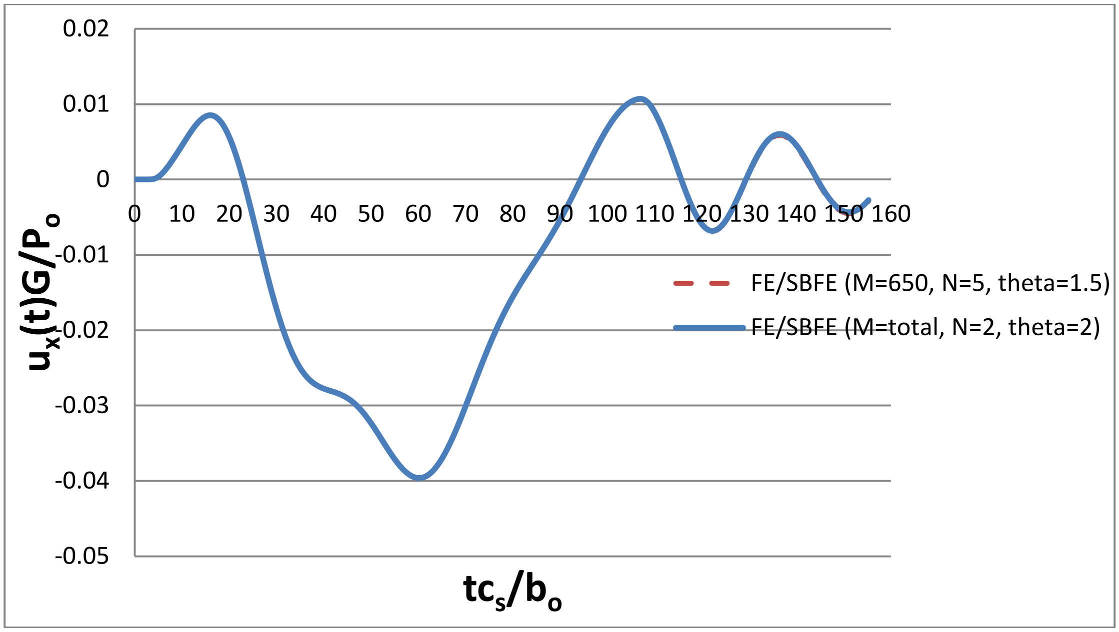


Fig.(5.13): Horizontal displacement of point A for N=5, M=650 and N=2, M=total

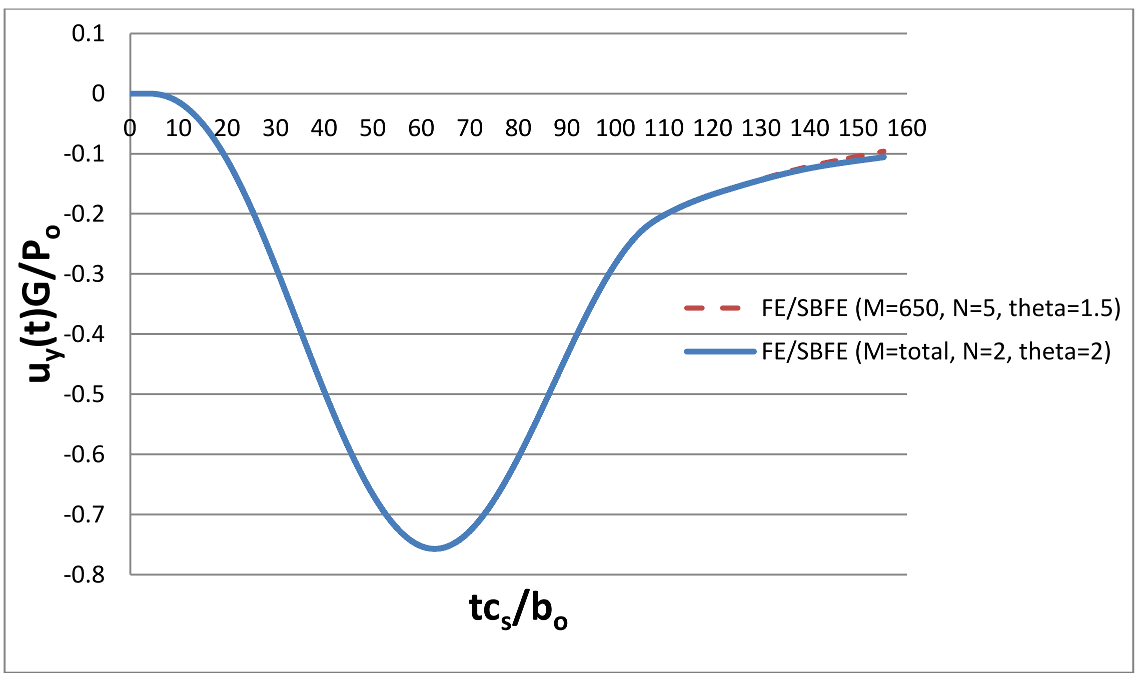


Fig.(5.14): Vertical displacement of point B for N=5, M=650 and N=2, M=total

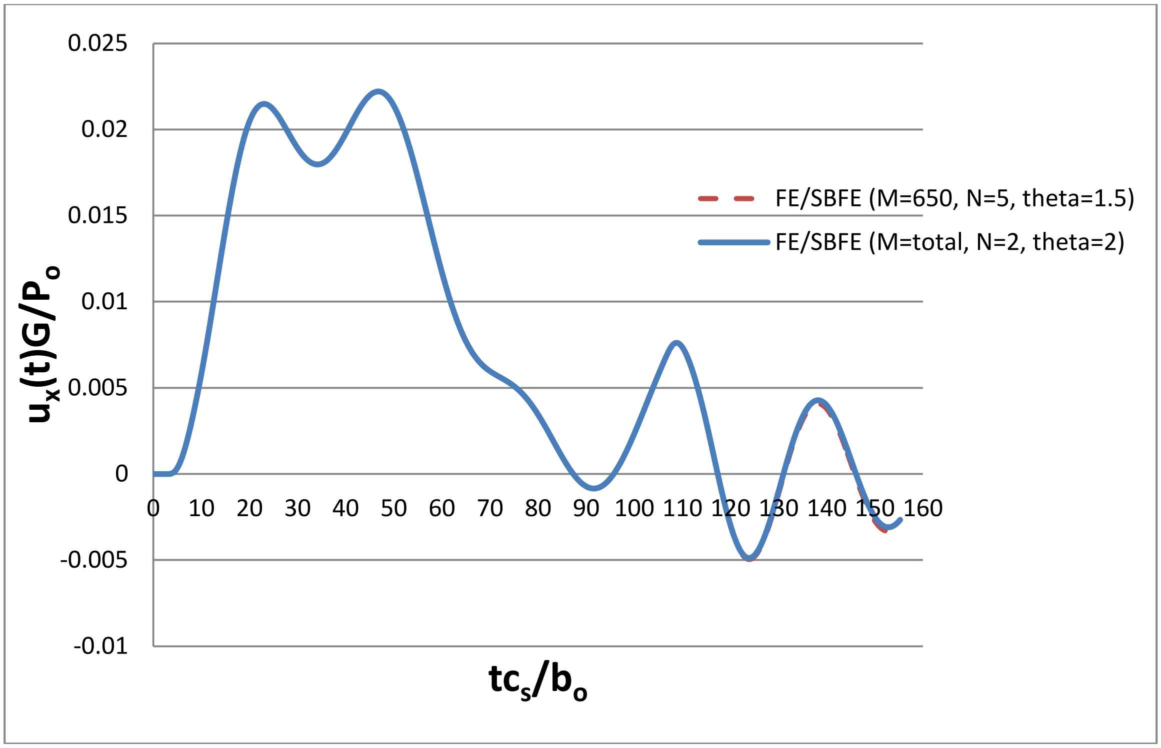


Fig.(5.15): Horizontal displacement of point B for N=5, M=650 and N=2, M=total

5.3 A strip footing in elastic half space with a trench

Here the in plane response of a strip footing embedded in a homogeneous isotropic soil is studied where a trench is near it [23], the trench is extended in the out of plane direction. The geometry of the problem is shown in fig.(5.16). The strip footing is subjected to a uniform double triangular impulse load as shown in the fig. (5.17) (for all the charts the wave velocities and the elastic constants are that of the soil), p_o is the maximum value, where:

$$p(t)=\begin{cases} 0.5p_o\dfrac{c_s}{e}t & 0\le t\le 0.5d/c_s \\ 2p_o\left(1-\dfrac{c_s}{e}t\right) & 0.5d/c_s\le t\le 1.5d/c_s \\ 2p_o\left(\dfrac{c_s}{e}t-2\right) & 1.5d/c_s\le t\le 2d/c_s \end{cases}$$

, in the analysis p_o=10000N/m^2 and t_o=1 sec, e is the embedment depth (e is taken in the analysis 2m)

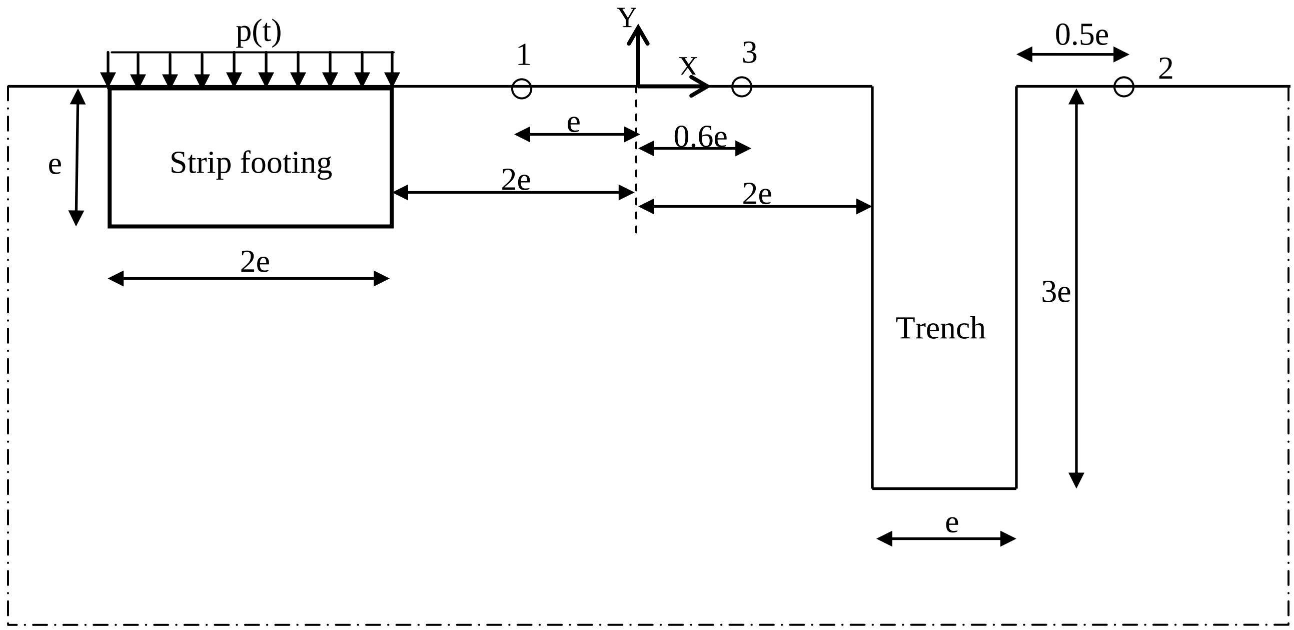


Fig.(5.16): Geometry of the problem

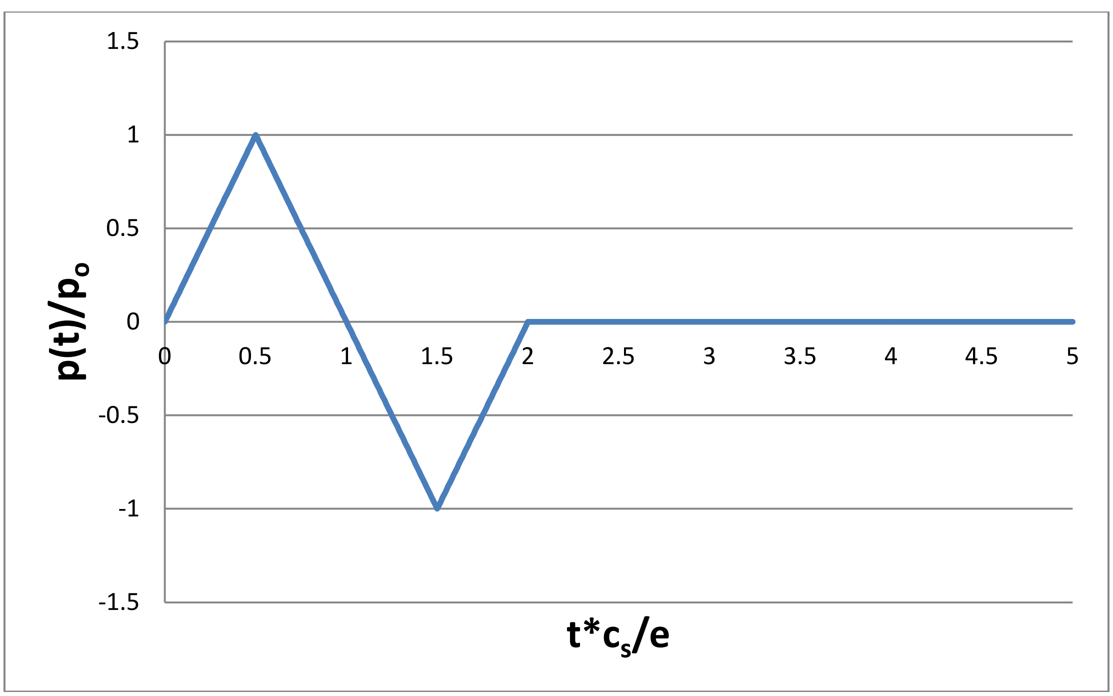


Fig.(5.17): Applied load

The material properties are:

For soil:

$E=3.25*10^8$ N/m^2, $\nu=0.3$ and $\rho=2300$ kg/m^3

For the concrete foundation:

$E=1.68*10^{10}$ N/m^2, $\nu=0.2$ and $\rho=2500$ kg/m^3

The problem here is a plane strain problem; it is modeled as shown in fig.(5.18), where part of the soil (including the strip footing and the trench) is modeled using FEM and the rest using SBFEM. The analysis is done using different domain dimensions and element sizes, as follows (Where, c_s is the shear wave velocity of the soil):

1. dimensions: 8eX4e (a=0, b=e and d=e) with 4-noded plane FE and 2-noded line SBFE (element size 0.5e, time step $0.03e/c_s$)
2. dimensions: 8eX4e (a=0, b=e and d=e) with 4-noded plane FE and 2-noded line SBFE (element size 0.25e, time step $0.015e/c_s$)
3. dimensions: 8eX4e (a=0, b=e and d=e) with 4-noded plane FE and 2-noded line SBFE (element size 0.125e, time step $0.01e/c_s$)
4. dimensions: 8eX4e (a=0, b=e and d=e) with 8-noded plane FE and 3-noded line SBFE (element size 0.125e, time step $0.01e/c_s$)
5. dimensions:8eX5e (a=0, b=e and d=2e) with 4-noded plane FE and 2-noded line SBFE (element size 0.125e, time step $0.01e/c_s$)
6. dimensions: 9eX5e (a=0, b=2e and d=2e) with 4-noded plane FE and 2-noded line SBFE (element size 0.125e, time step $0.01e/c_s$)
7. dimensions: 10eX5e (a=e, b=2e and d=2e) with 4-noded plane FE and 2-noded line SBFE (element size 0.125e, time step $0.01e/c_s$)
8. dimensions: 9eX6e (a=0, b=2e and d=3e) with 4-noded plane FE and 2-noded line SBFE (element size 0.125e, time step $0.01e/c_s$)

Cases 3, 5, 6, 7 and 8 are repeated with refining the part at the trench with width 0.375e (as shown in fig.(5.19)).

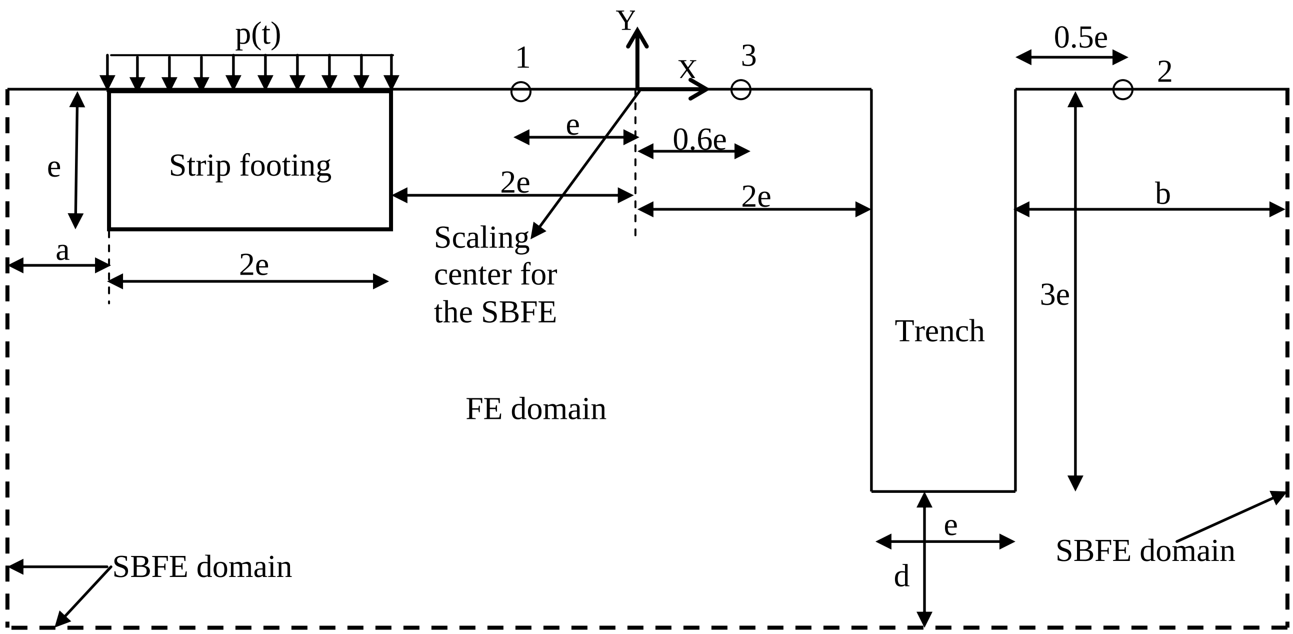


Fig.(5.18): The problem model

All the above cases are solved with N=2, for cases 1, 2 and 3 the unit impulse response function is not truncated (M=1500) but for the other cases it is truncated at M=350 step. The calculated displacement at points 1, 2 and 3 are shown in figures (5.20) through (5.25), where (2) is placed beside the name of the refined model ($P_o=p_o*2e$).

For the solution to be validated the problem is solved again using extended FE mesh with dimensions 40eX20e as shown in fig.(5.19) using SAP2000 FE program ,the model is supported with rigid hinges on the boundaries (the dimensions are taken so that the wave will not be reflected again in the domain within the time of study), the displacement for the points 1,2 and 3 are plotted with the other displacements (figures (5.20) through (5.25)).

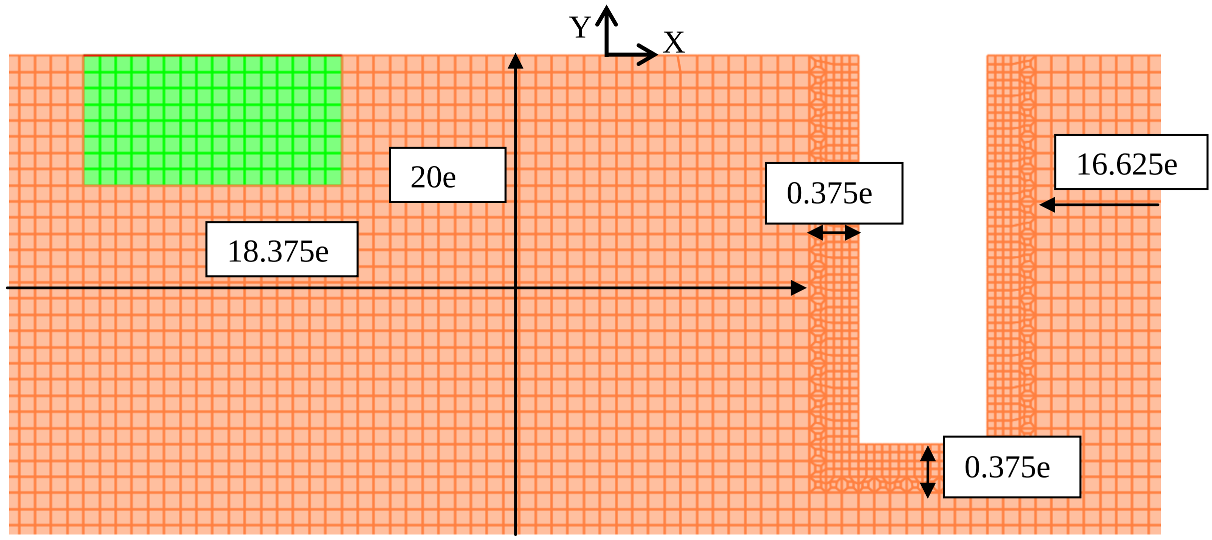


Fig.(5.19): The extended FE model

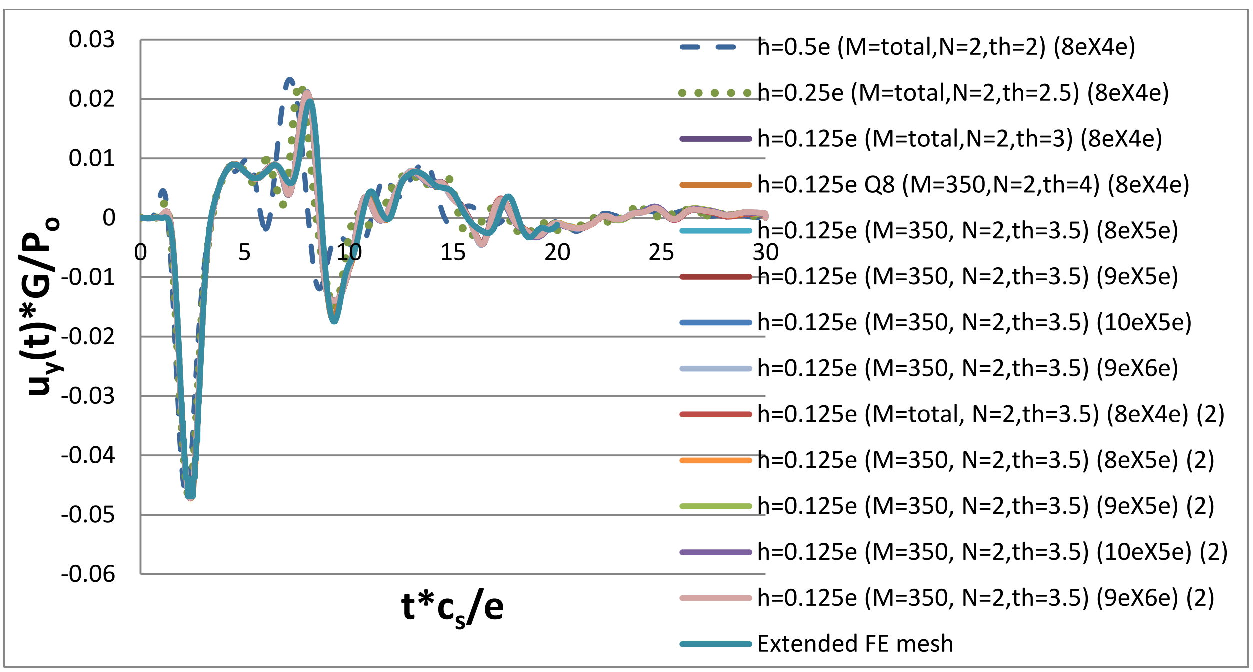


Fig.(5.20): Vertical displacement of point 1

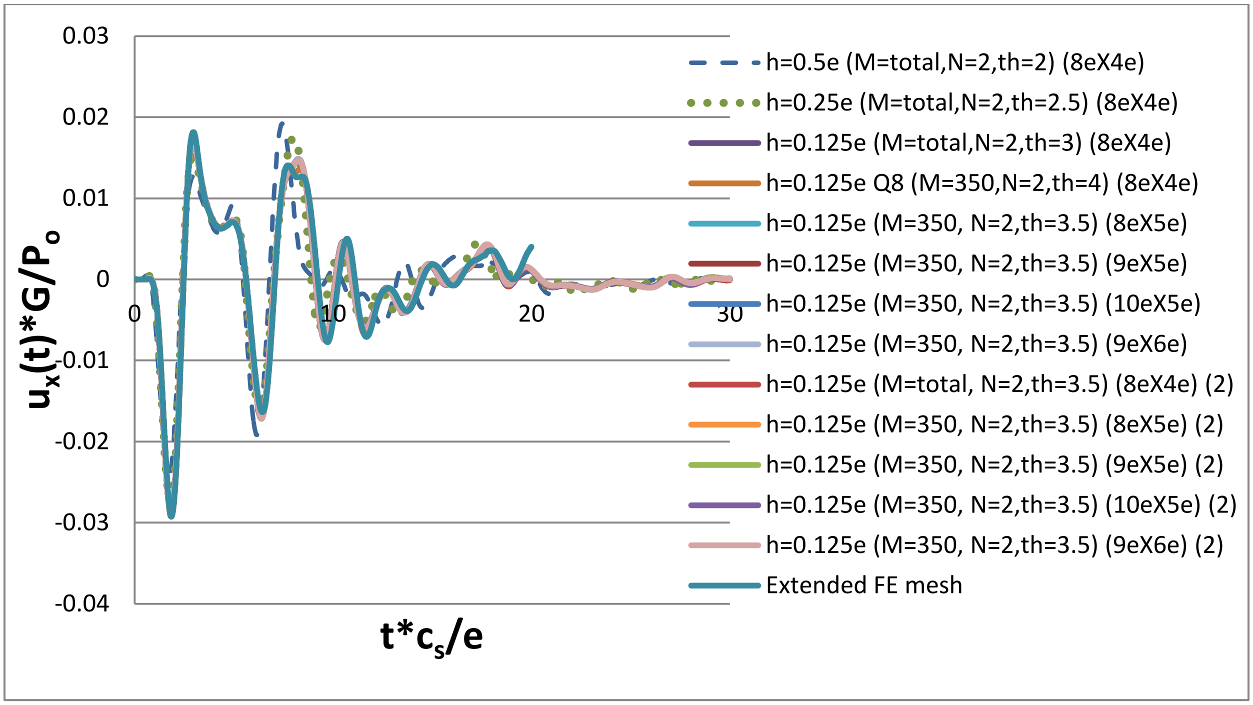


Fig.(5.21): Horizontal displacement of point 1

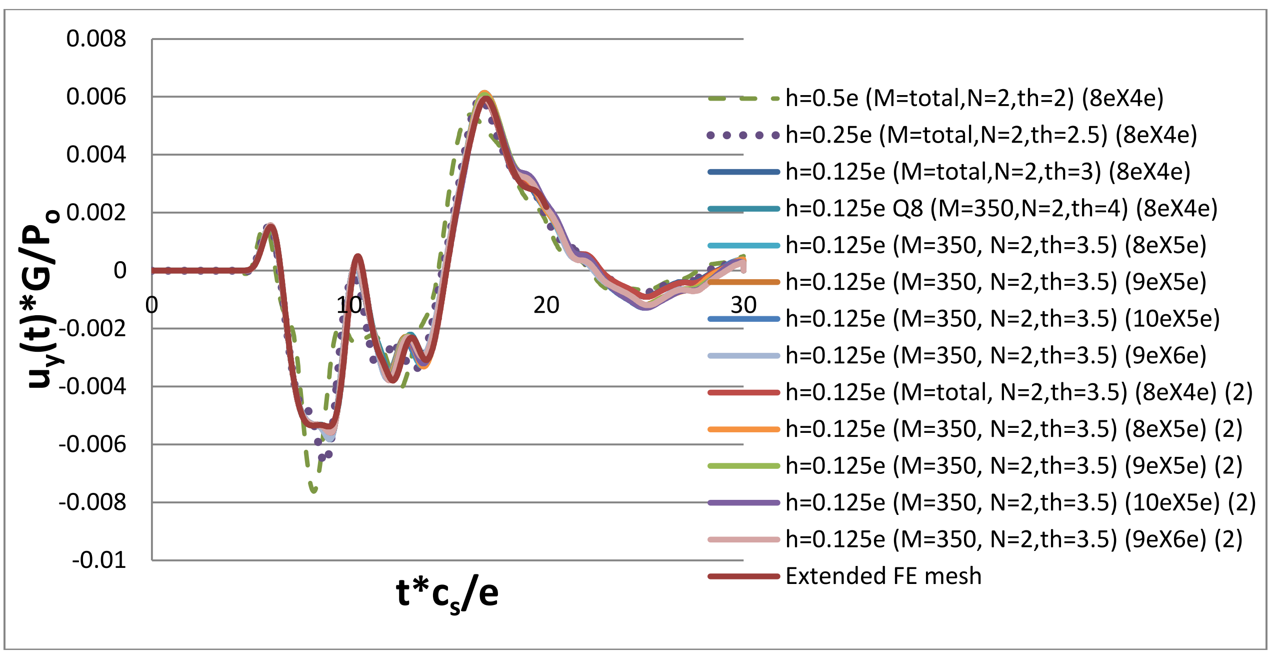


Fig.(5.22): Vertical displacement of point 2

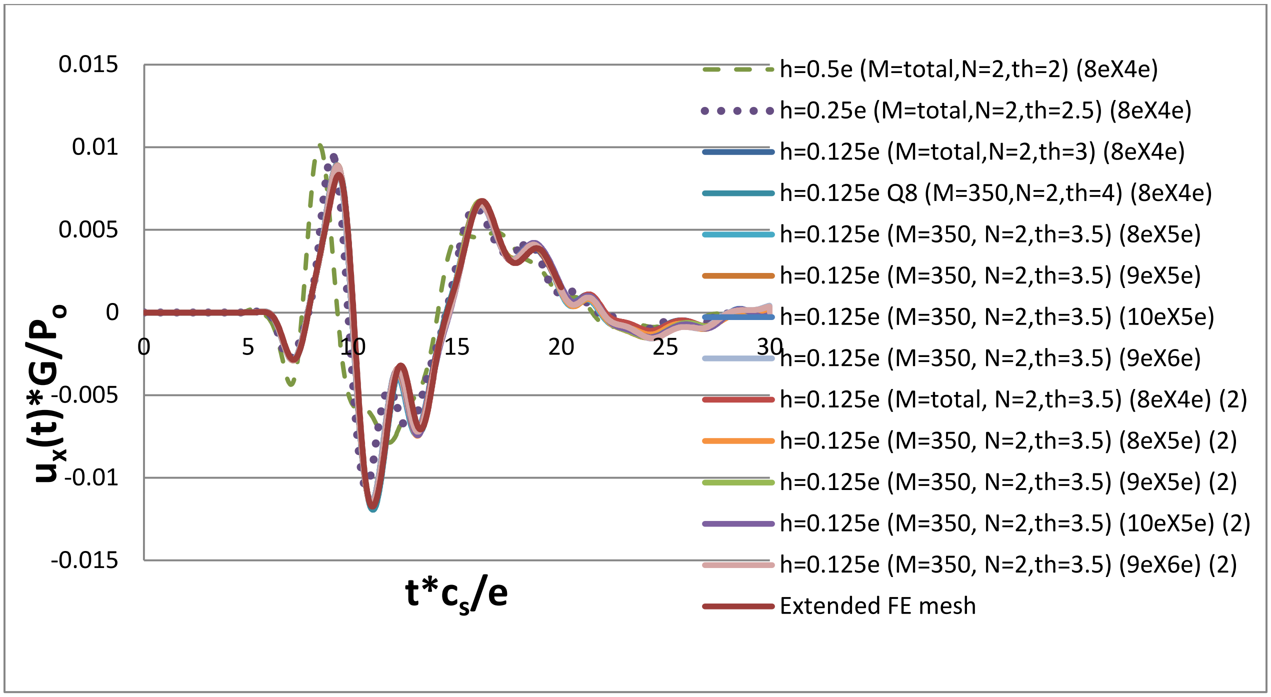


Fig.(5.23): Horizontal displacement of point 2

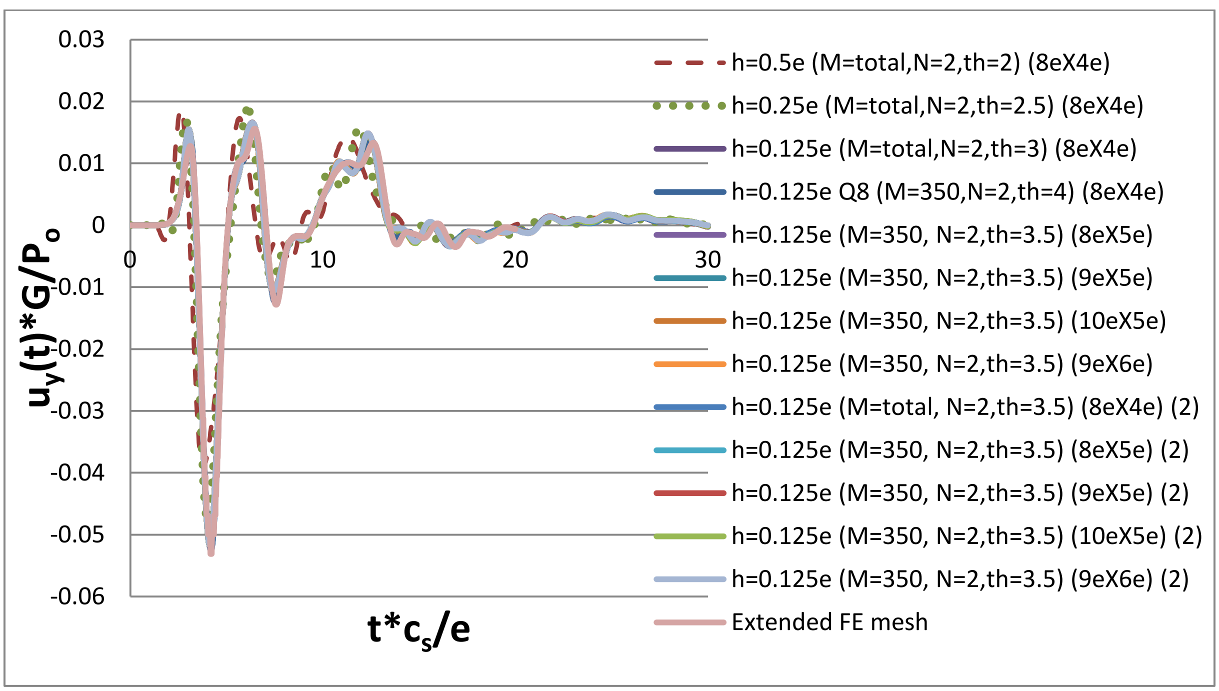


Fig.(5.24): Vertical displacement of point 3

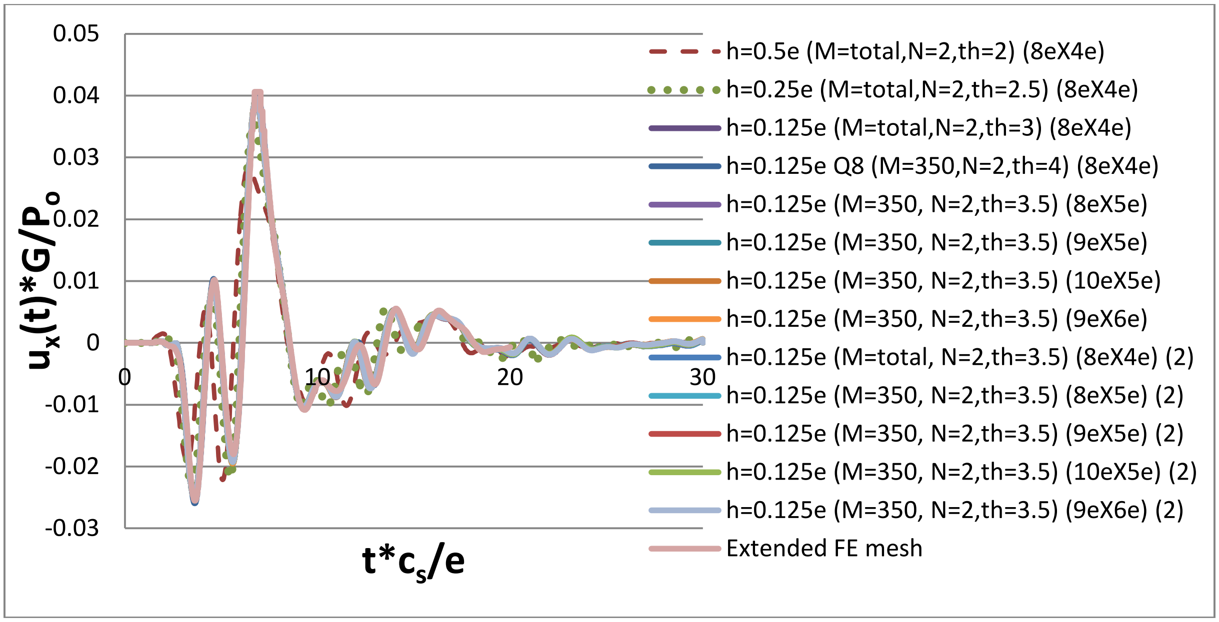


Fig.(5.25): Horizontal displacement of point 3

It is clear from above that all the models except the first 2 nearly agree with each other and with the extended FE mesh model, this is also the case for the models where the time of the unit impulse is truncated

Here I am going to compare the results of the displacements of the 3 points under consideration when using the total time and the truncated time in calculating the unit impulse response function also a large N is used. Figures (5.26) through (5.31) shows the displacements of the 3 points for the 3rd model in case of total time (M=total (1500), N=2 and θ=3) and in case of truncated time (M=140, N=5 and θ=3).

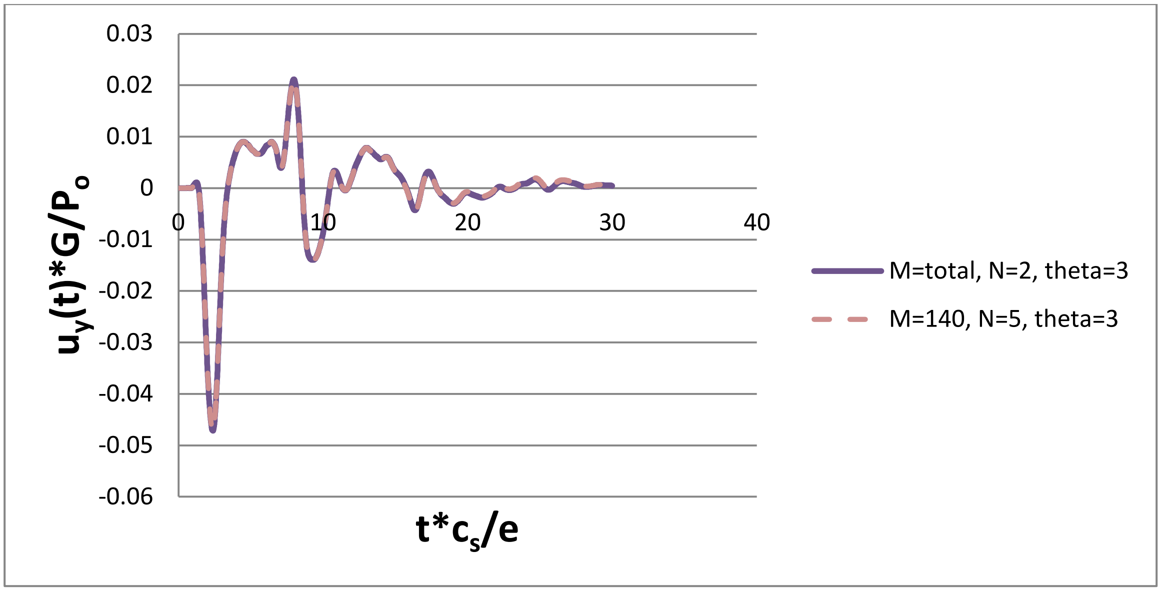


Fig.(5.26): Vertical displacement of point 1 for N=5, M=140 and N=2, M=total

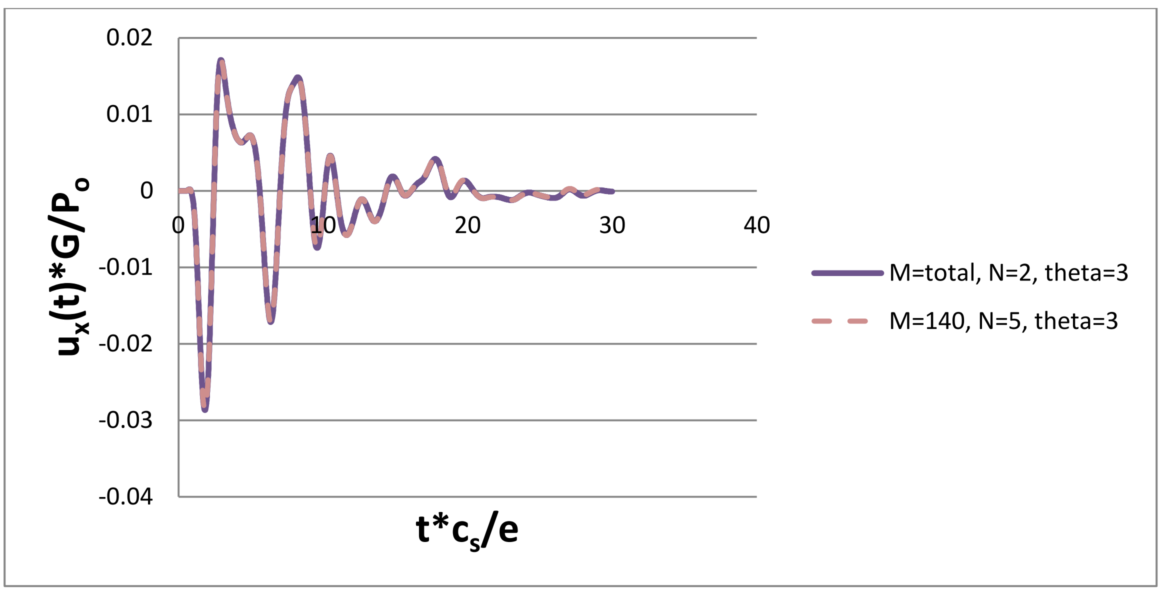


Fig.(5.27): Horizontal displacement of point 1 for N=5, M=140 and N=2, M=total

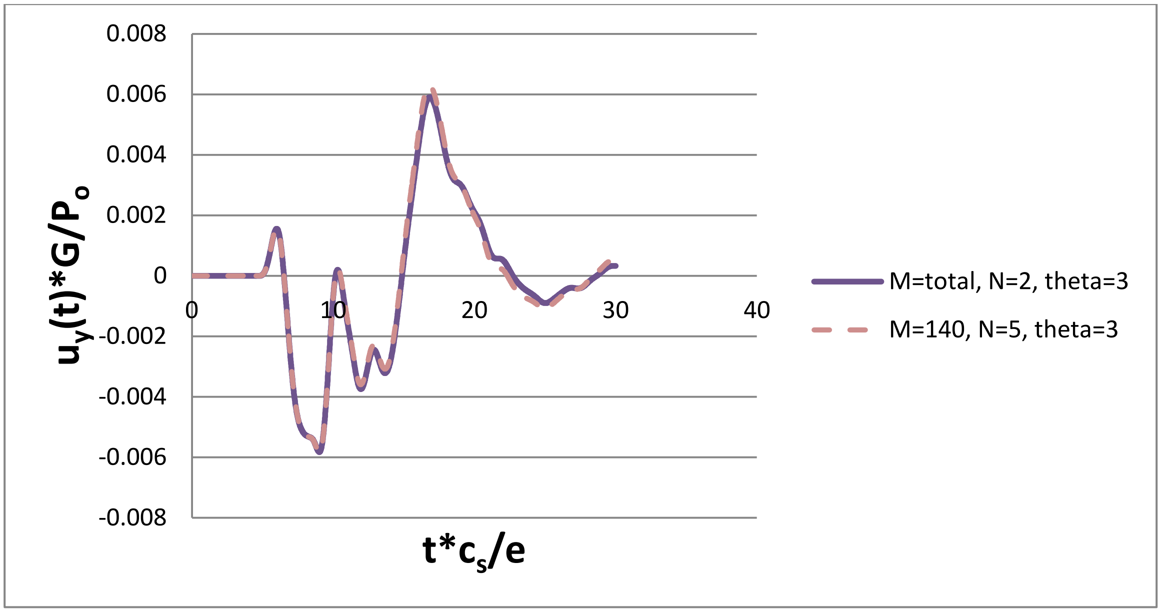


Fig.(5.28): Vertical displacement of point 2 for N=5, M=140 and N=2, M=total

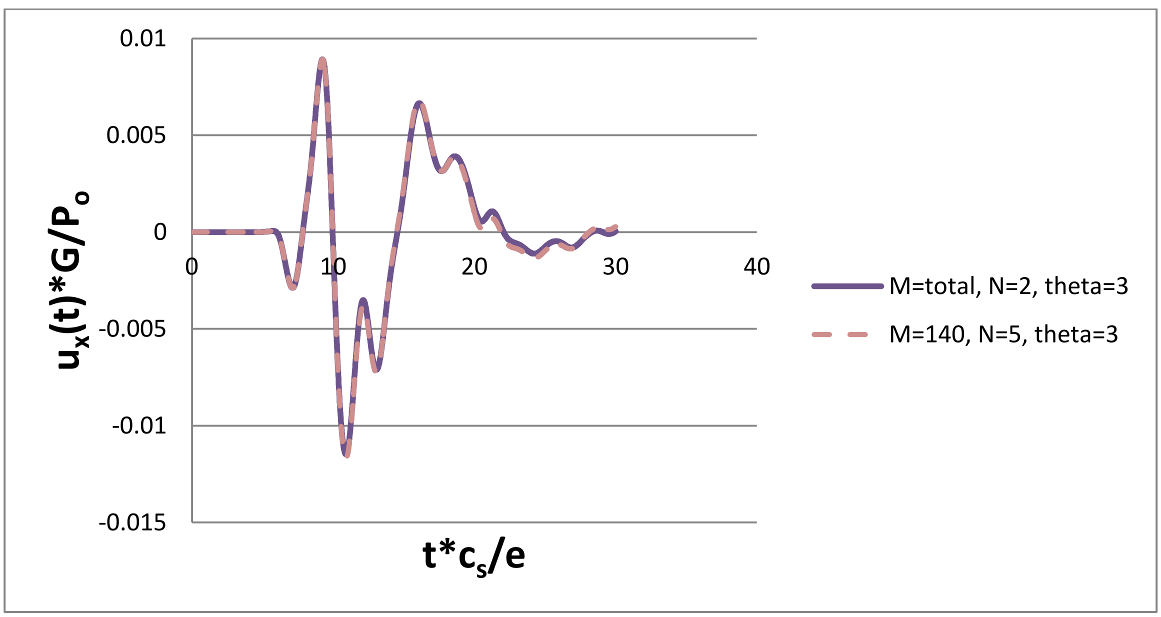


Fig.(5.29): Horizontal displacement of point 2 for N=5, M=140 and N=2, M=total

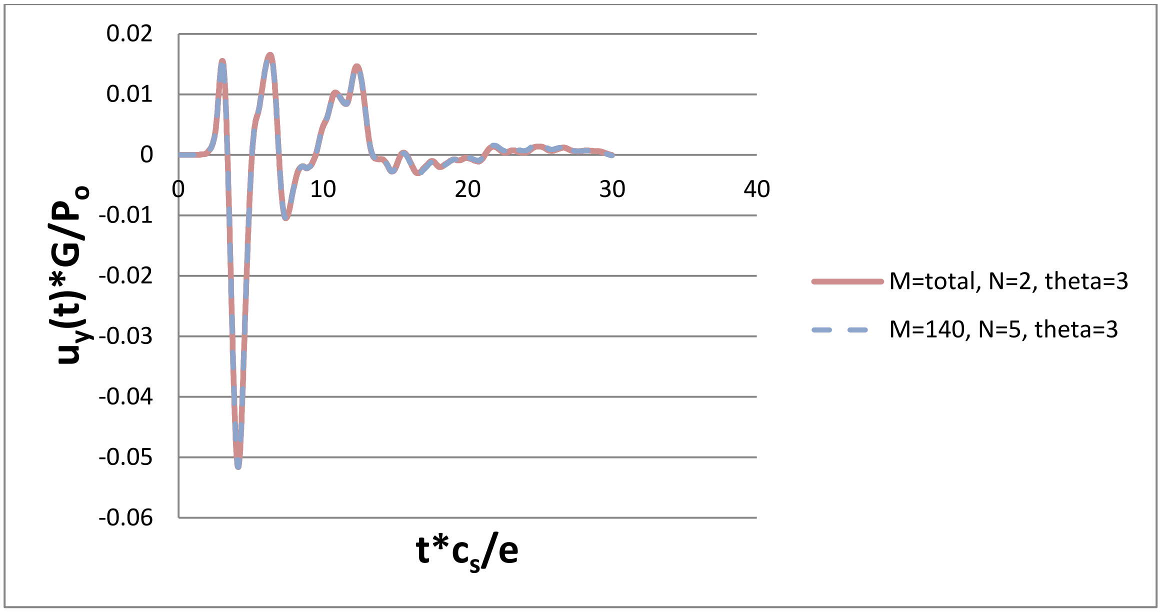


Fig.(5.30): Vertical displacement of point 3 for N=5, M=140 and N=2, M=total

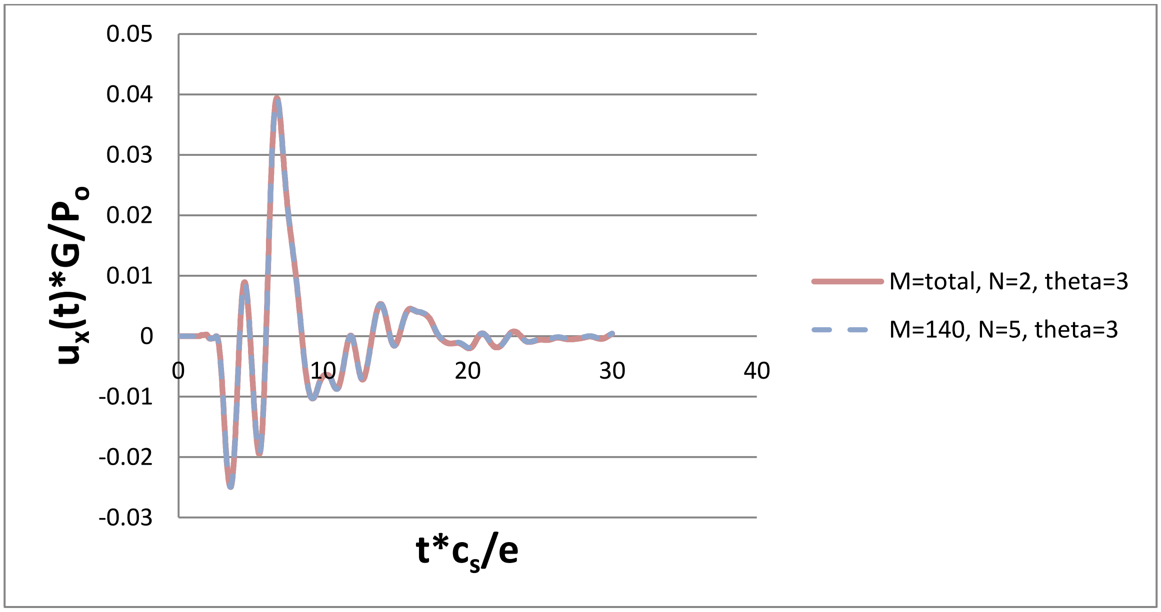


Fig.(5.31): Horizontal displacement of point 3 for N=5, M=140 and N=2, M=total

It is clear from above that the two results agree with each other, except for point 2 where small difference occurs.

5.4 Tunnel in elastic half space

Here the in-plane motion of a concrete tunnel in elastic half space subjected to two uniform line loadings is studied. The geometry of the tunnel is shown in fig.(5.32) below, where points G, C, F and A are the observation points. The load applied is a sinusoidal load as shown in fig. (5.33) (for all the charts the wave velocities and the elastic constants are that of the soil).

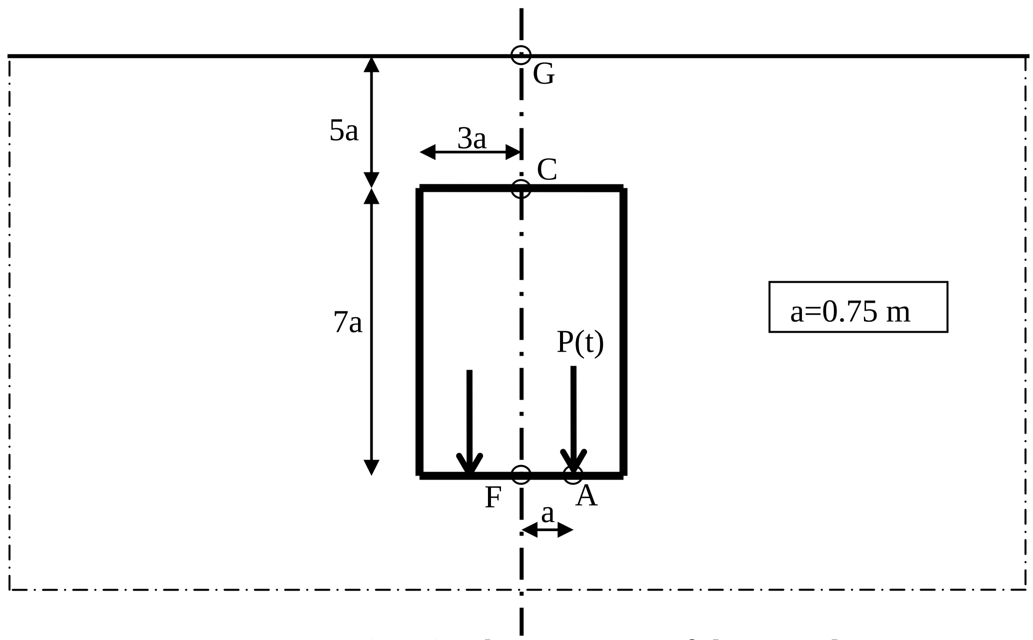


Fig.(5.32): The geometry of the tunnel

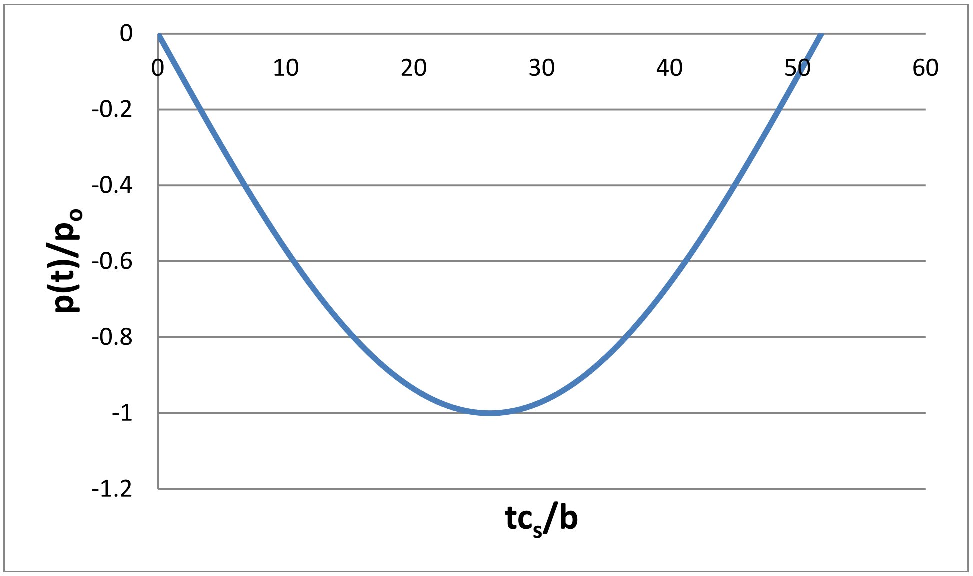


Fig.(5.33): The sinusoidal load

The material properties are:

For soil:

E=667*10^4 N/m^2, ν=1/3 and ρ=183.7 kg/m^3

For the concrete tunnel:

E=22*10^9 N/m^2, ν=0.2 and ρ=2500 kg/m^3

The tunnel section is of depth 0.75 m and breadth 1 m

As previous problems part of the unbounded domain is modeled using FE and the rest using SBFE as shown in fig.(5.34). Due to symmetry of the problem only half of it is analyzed, for the FE part four-noded isoparametric quadratic element is used and for SBFE part two-noded isparametric line element is used, and the tunnel itself is modeled using two-noded frame element. The problem is solved using different mesh size with the element size equal “a", where the time step equals $0.01037 b_o / c_s$ (c_s is the shear velocity of the soil):

1. b=a and d=a
2. b=a and d=3a
3. b=3a and d=a
4. b=3a and d=3a

5. b=3a and d=5a
6. b=5a and d=3a
7. b=5a and d=5a

where for all of the above M=total (i.e. the total time is used in calculation the unit impulse), N=2 and θ=2

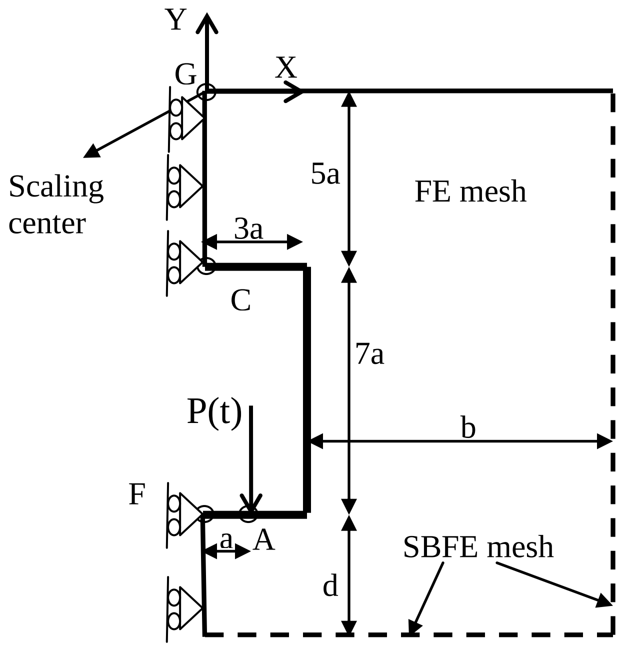


Fig.(5.34): The coupled FE/SBFE model

The results for the vertical displacement of point G, C, F and A are shown in figures (5.36) through (5.40)

To validate the answer, since there is no exact solution for this problem a numerical model is done using extended FE mesh (using SAP2000 program ver.16) as shown in fig.(5.35) with dimensions 93X195 m using uniform element size as used in the coupled FE/SBFE model.

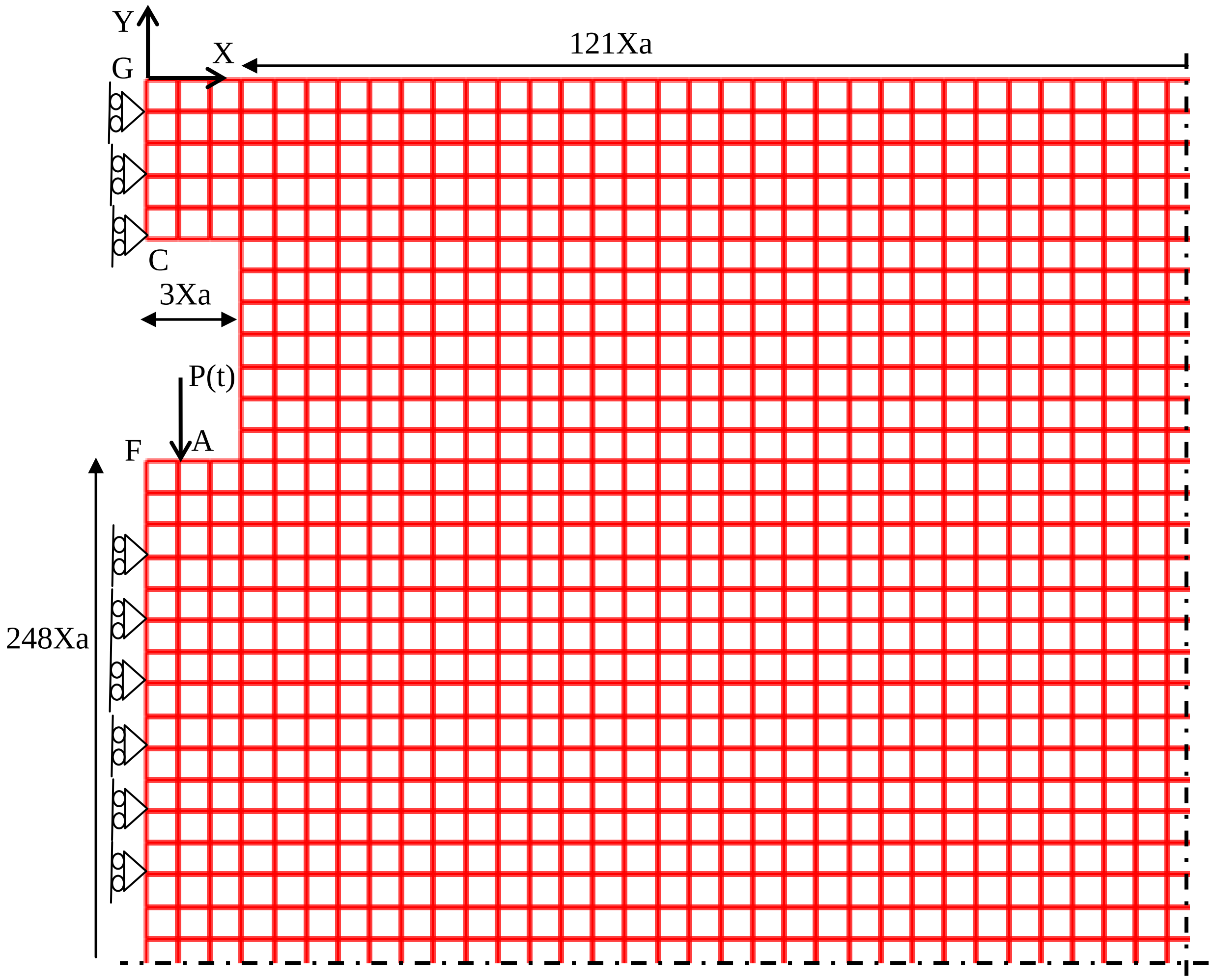


Fig.(5.35): Extended FE mesh model

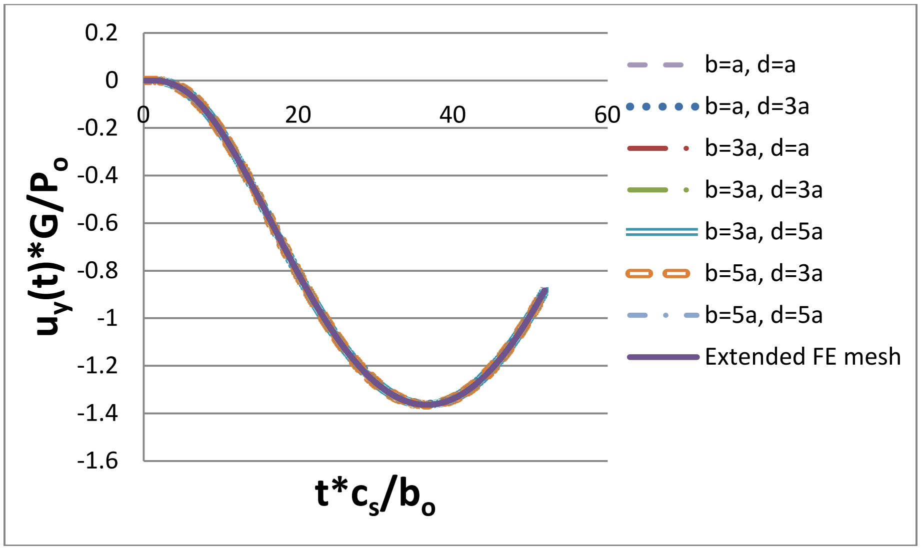


Fig.(5.36): Vertical displacement of point G

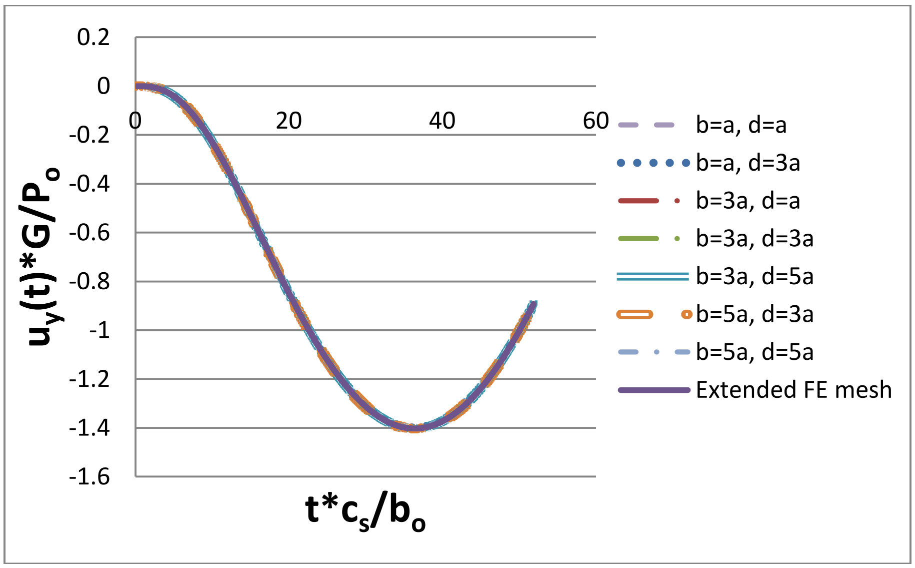


Fig.(5.37): Vertical displacement of point C

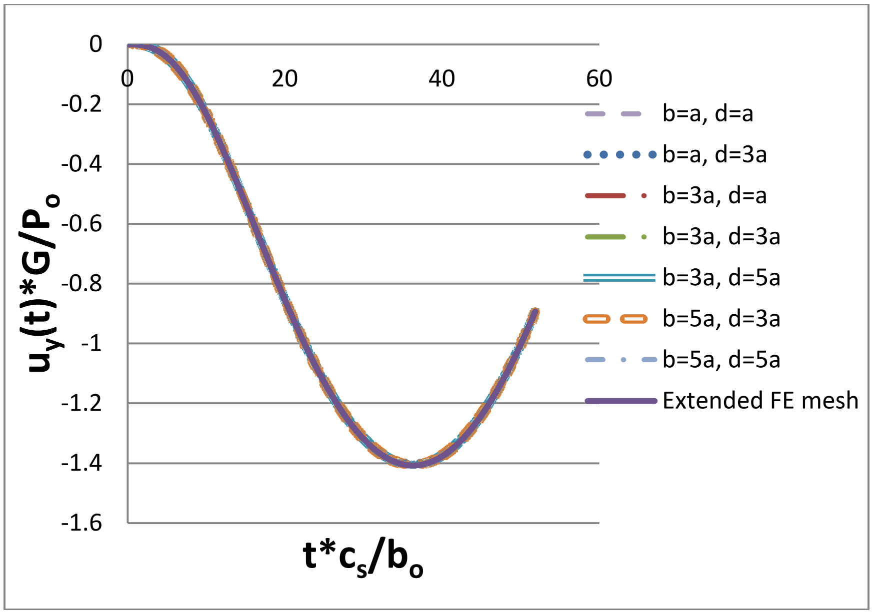


Fig.(5.38): Vertical displacement of point F

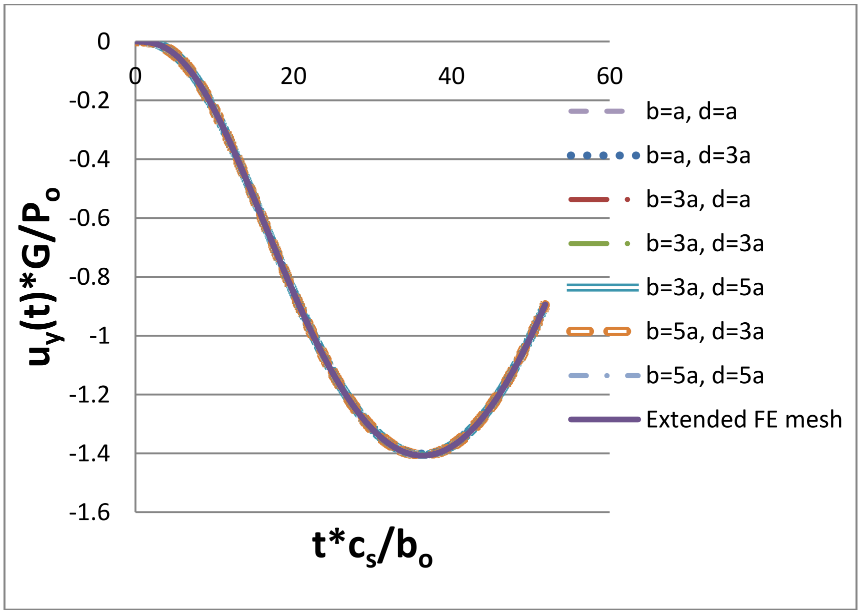


Fig.(5.39): Vertical displacement of point A

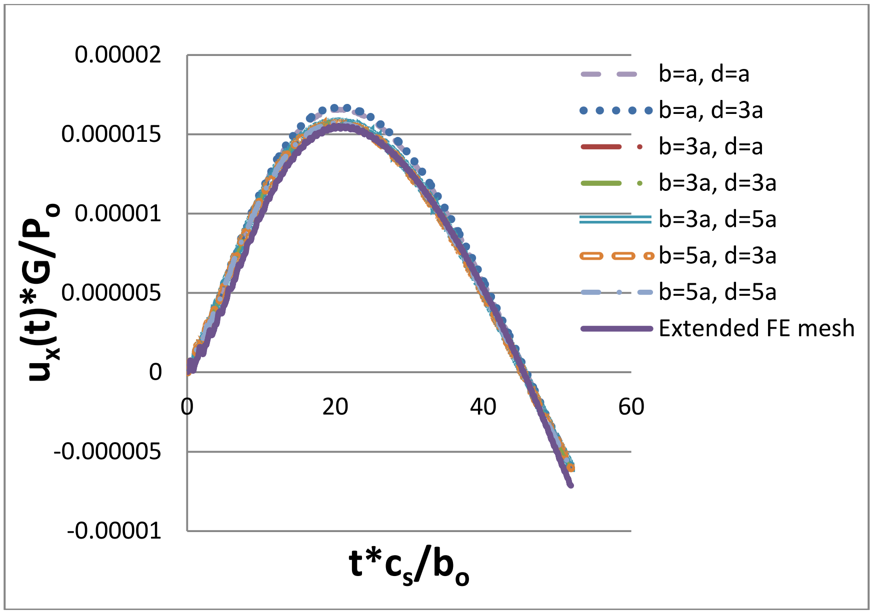


Fig.(5.40): Horizontal displacement of point A

The maximum displacements in the above figures are shown in the tables below for comparison:

For point G

Dimensions	u_y*G/P	diff.%	diff.% w.r.t. FE
b=a, d=a	-1.359		-0.375
b=a, d=3a	-1.359	0.002	-0.373
b=3a, d=a	-1.363	0.264	-0.110
b=3a, d=3a	-1.363	0.008	-0.102
b=3a, d=5a	-1.363	0.006	-0.096
b=5a, d=3a	-1.364	0.073	-0.023
b=5a, d=5a	-1.364	0.008	-0.015

Where, for FE model uy*G/P= -1.364

For point C

Dimensions	u_y*G/P	diff.%	diff.% w.r.t. FE
b=a, d=a	-1.398		-0.336
b=a, d=3a	-1.398	0.002	-0.333
b=3a, d=a	-1.402	0.276	-0.058
b=3a, d=3a	-1.402	0.008	-0.049

b=3a, d=5a	-1.402	0.006	-0.043
b=5a, d=3a	-1.403	0.075	0.032
b=5a, d=5a	-1.403	0.008	0.039

Where, for FE model u_y*G/P= -1.403

For point F

Dimensions	u_y*G/P	diff.%	diff.% w.r.t. FE
b=a, d=a	-1.401		-0.495
b=a, d=3a	-1.401	0.002	-0.493
b=3a, d=a	-1.405	0.277	-0.217
b=3a, d=3a	-1.405	0.008	-0.209
b=3a, d=5a	-1.405	0.006	-0.203
b=5a, d=3a	-1.406	0.075	-0.128
b=5a, d=5a	-1.406	0.008	-0.121

Where, for FE model u_y*G/P= -1.408

For point A

Dimensions	u_y*G/P	diff.%	diff.% w.r.t. FE	u_x*G/P	diff.%	diff.% w.r.t. FE
b=a, d=a	-1.400		-0.496	1.657E-05		6.640
b=a, d=3a	-1.400	0.002	-0.494	1.673E-05	0.971	7.685
b=3a, d=a	-1.404	0.277	-0.218	1.572E-05	-6.041	1.180
b=3a, d=3a	-1.404	0.008	-0.210	1.581E-05	0.528	1.717
b=3a, d=5a	-1.404	0.006	-0.204	1.581E-05	0.047	1.766
b=5a, d=3a	-1.405	0.075	-0.129	1.571E-05	-0.622	1.133
b=5a, d=5a	-1.406	0.008	-0.122	1.572E-05	0.009	1.142

Where, for FE model u_y*G/P= -1.407, and u_x*G/P= 1.554E-05

Here also is the Bending moment diagram of the frame element at the maximum load as shown in fig.(5.41) for all the cases above (where, the dots represents points F, C and the corners of the tunnel).

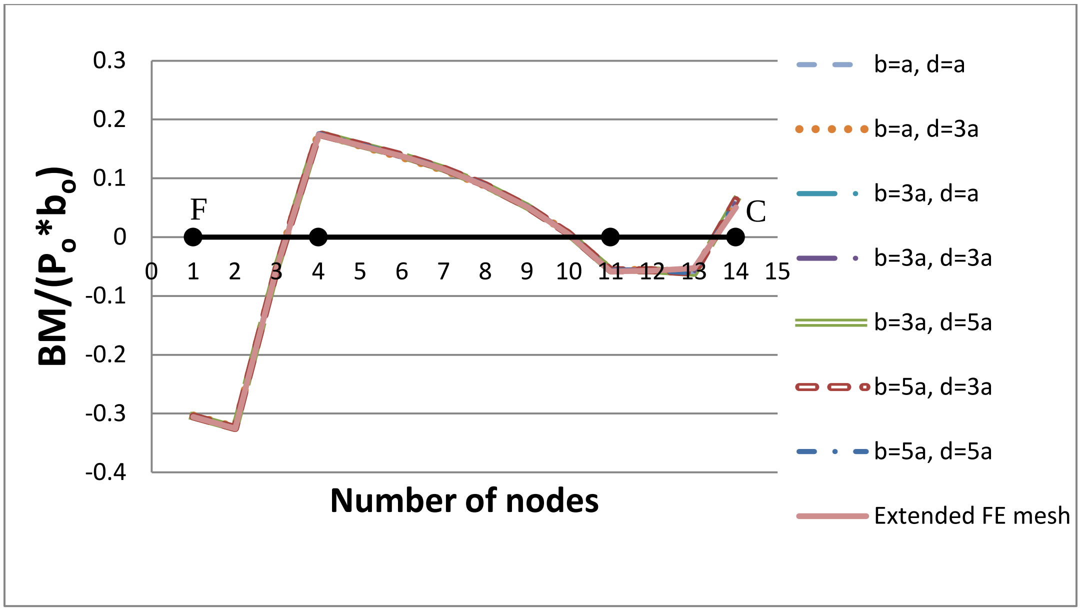


Fig.(5.41): Bending moment of the tunnel at maximum load

It is clear from above that:

1. the difference between the FE model and the coupled FE/SBFE model is small
2. increasing the dimension "b" affect the results more than increasing the dimension "d"

In order to test the effect of decreasing the element size the problem is solved for element size equal 0.5a and also for N=2, M=total and theta=6, the case solved is b=d=3a. The vertical displacement of points G, C, F and A and the horizontal displacement of A in comparison with the results from the case M=total, N=2 and θ=2 are shown below (figures 5.42 through 5.46), where there is a small difference between the results, the largest difference is in the horizontal of point A where the difference is 6% which is also small.

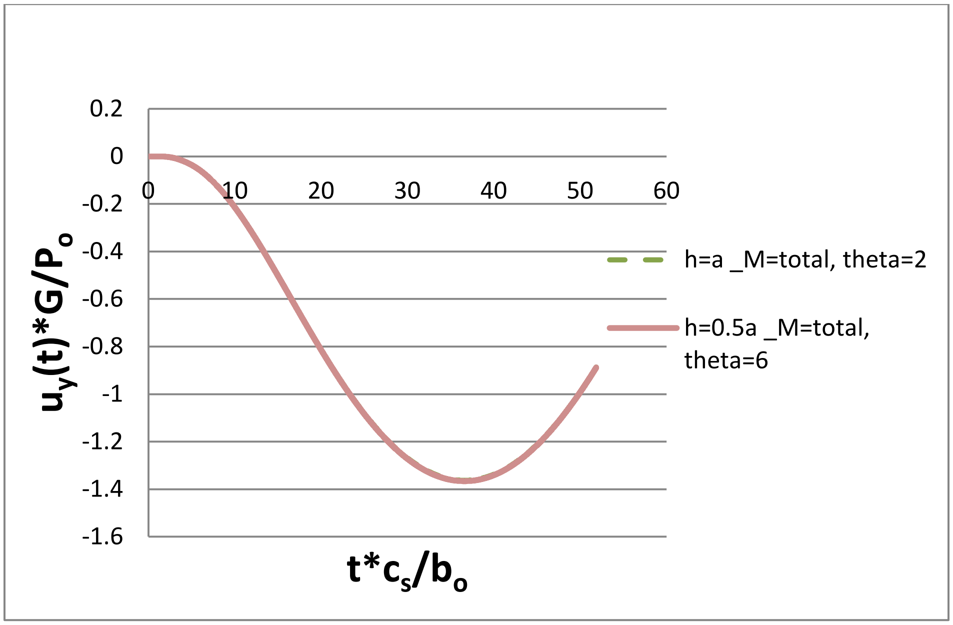


Fig.(5.42): Vertical displacement of point G for element size “a” and “0.5a”

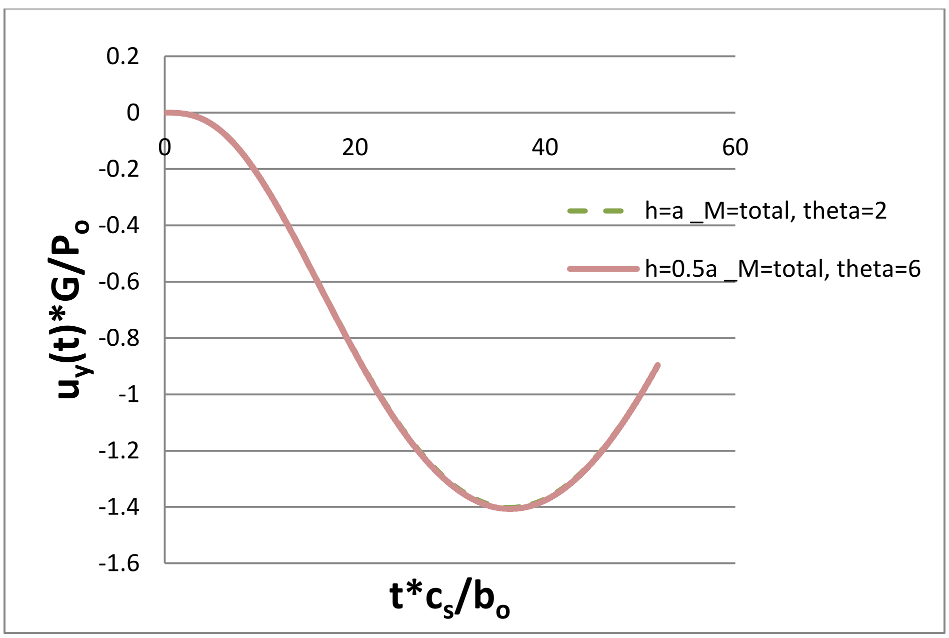


Fig.(5.43): Vertical displacement of point C for element size “a” and “0.5a”

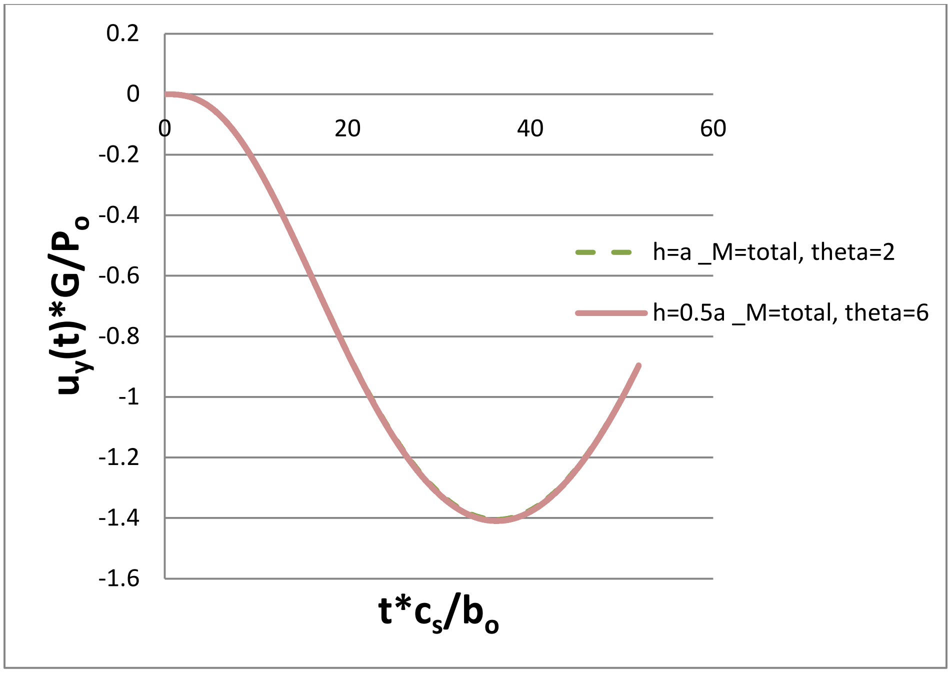


Fig.(5.44): Vertical displacement of point F for element size “a” and “0.5a”

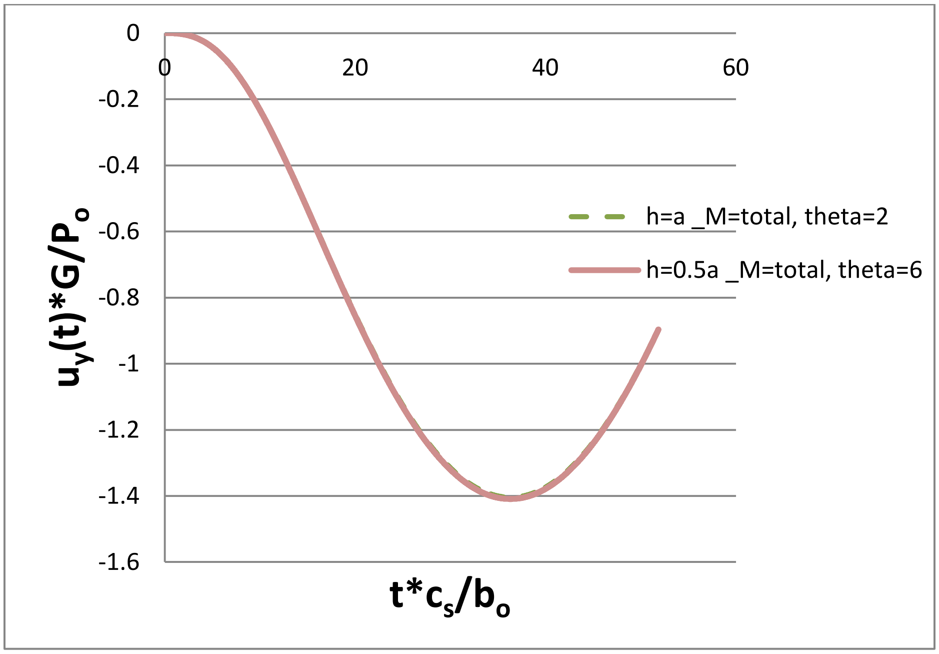


Fig.(5.45): Vertical displacement of point A for element size “a” and “0.5a”

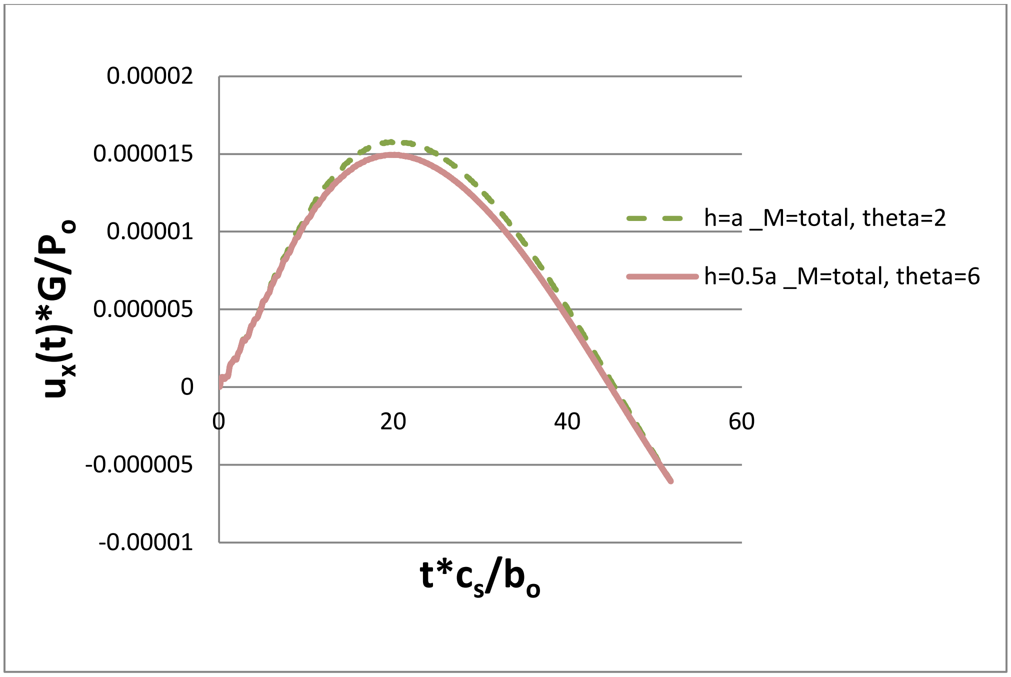


Fig.(5.46): Horizontal displacement of point A for element size "a" and "0.5a"

In order to test the efficiency of the method in case of large N, the problem is solved for N=5, M=total and with different θ (2, 1.9 and 1.5) for the case of b=d=3a. The vertical displacement of points G, C, F and A and the horizontal displacement of A in comparison with the results from the case M=total, N=2 and θ=2 are shown below (figures 5.47 through 5.51), where the element size used is "a".

A good agreement between the results is shown except for θ=1.5 where the method becomes unstable.

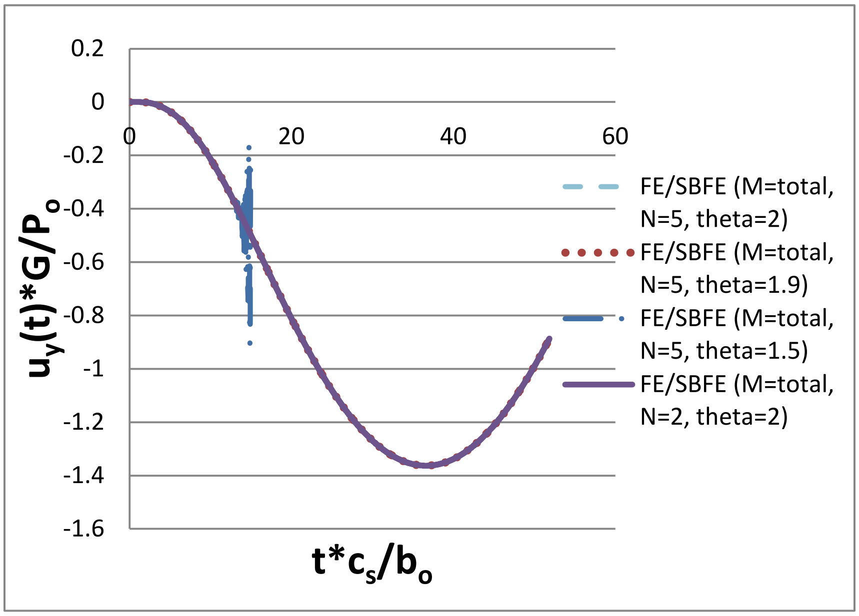


Fig.(5.47): Vertical displacement of point G for N=5 and N=2

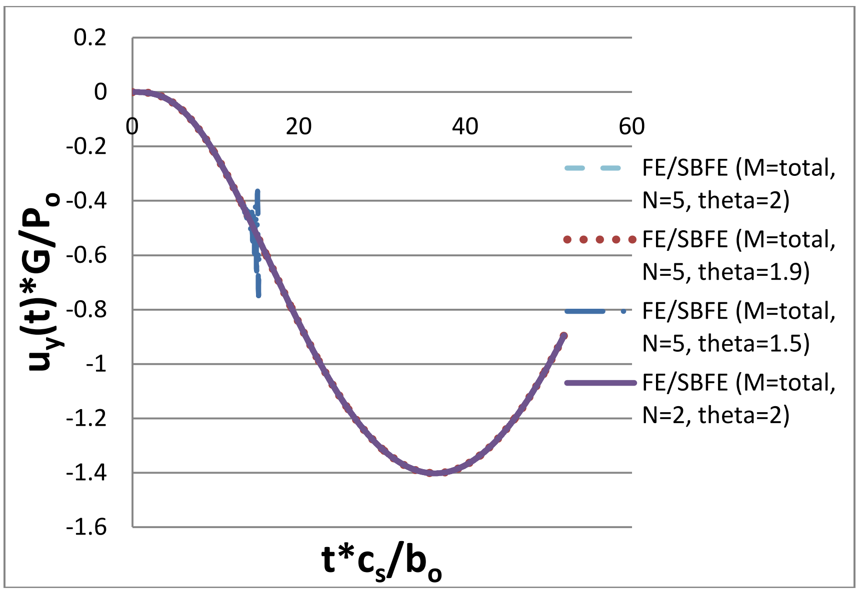


Fig.(5.48): Vertical displacement of point C for N=5 and N=2

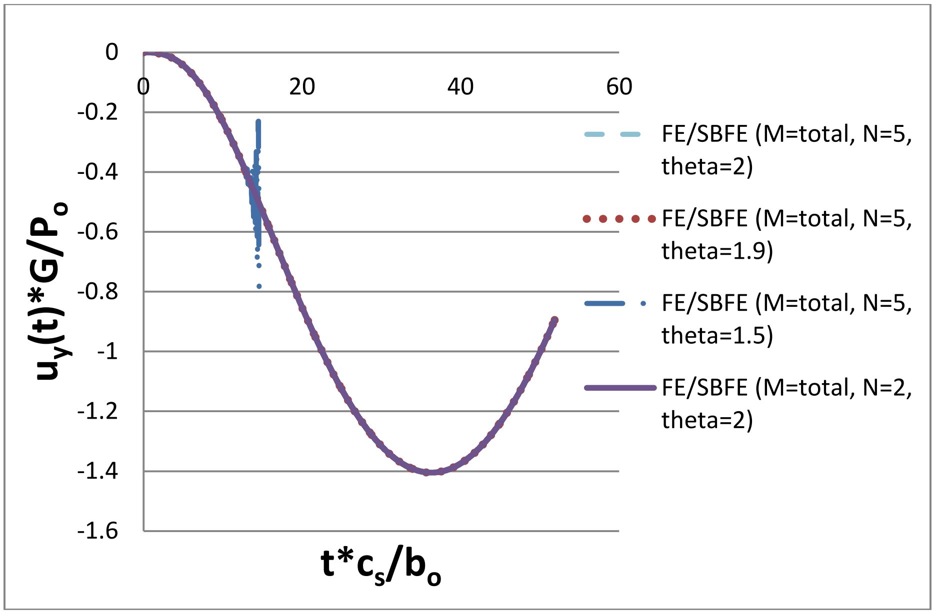


Fig.(5.49): Vertical displacement of point F for N=5 and N=2

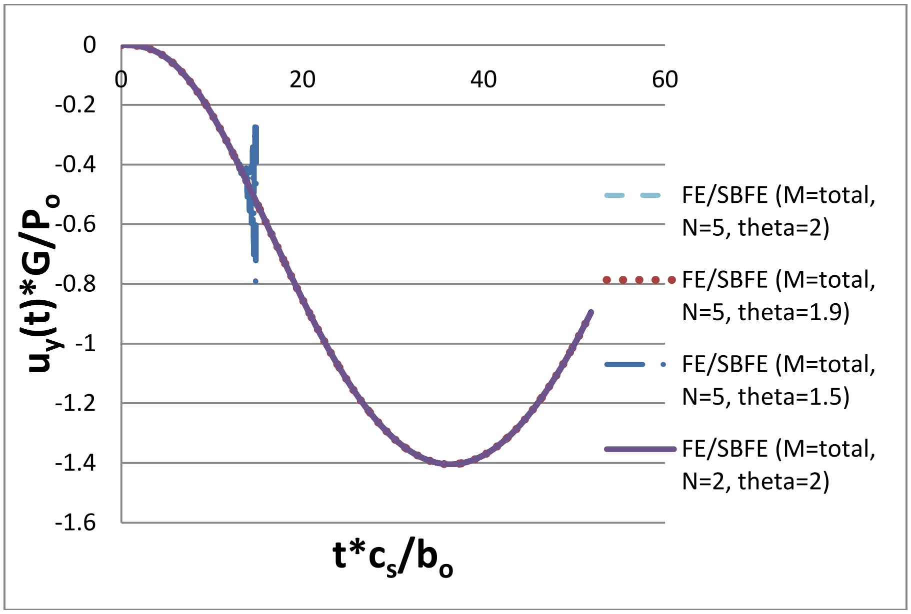


Fig.(5.50): Vertical displacement of point A for N=5 and N=2

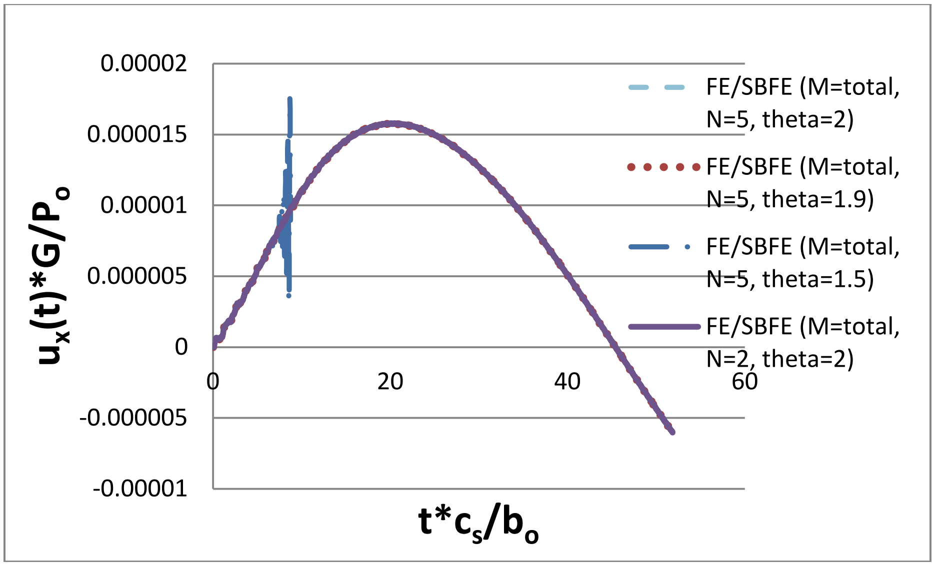


Fig.(5.51): Horizontal displacement of point A for N=5 and N=2

To see the effect of the truncation of the time of calculation of the unit impulse the problem is solved for M=350, N=5 with θ=2 for the case of b=d=3a. The vertical displacement of points G, C, F and A and the horizontal displacement of A in comparison with the results from the case M=total (2500), N=2 and θ=2, with element size “a”. (Figures 5.52 through 5.56), also the bending moment of the tunnel is shown in fig.(5.57) at the maximum load, where good agreement between the results.

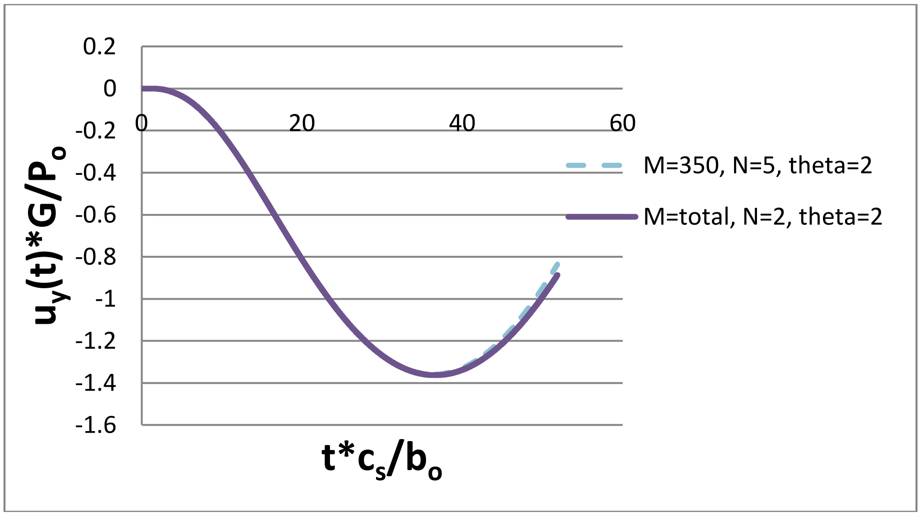


Fig.(5.52): Vertical displacement of point G for M=350, N=5 and M=total, N=2

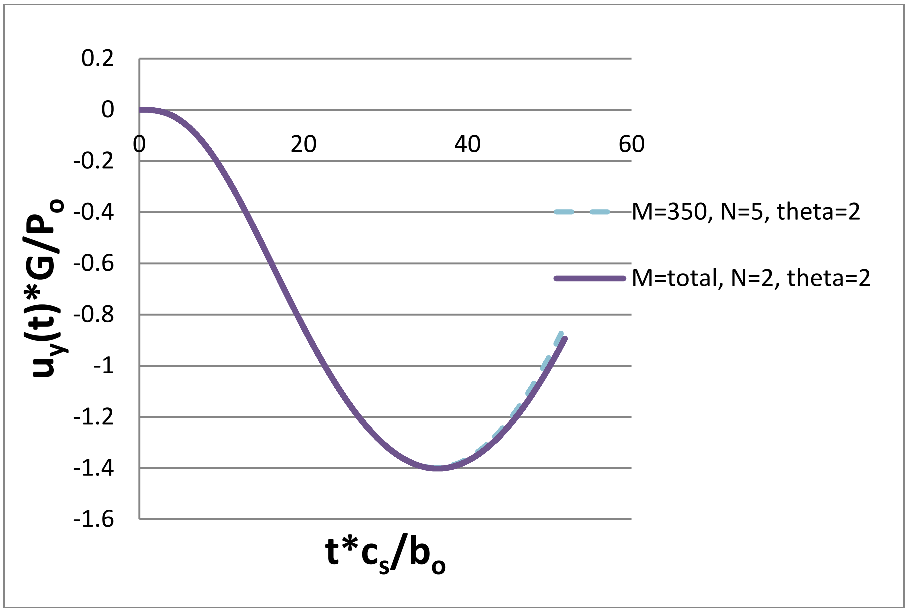


Fig.(5.53): Vertical displacement of point C for M=350, N=5 and M=total, N=2

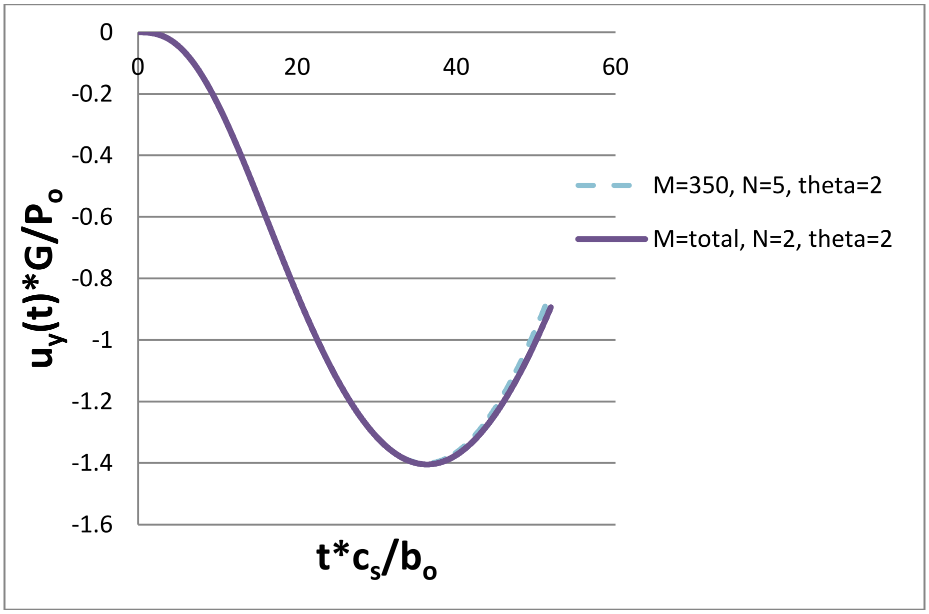


Fig.(5.54): Vertical displacement of point F for M=350, N=5 and M=total, N=2

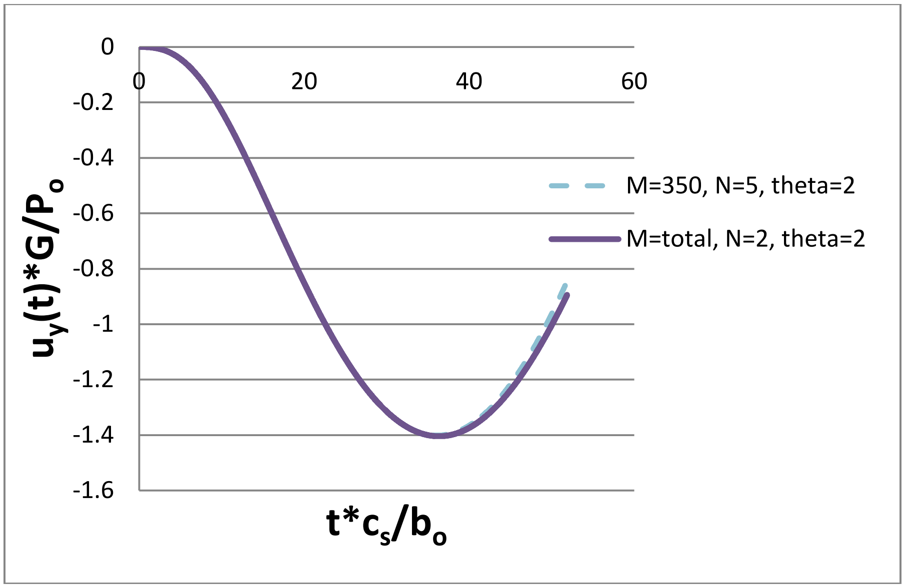


Fig.(5.55): Vertical displacement of point A for M=350, N=5 and M=total, N=2

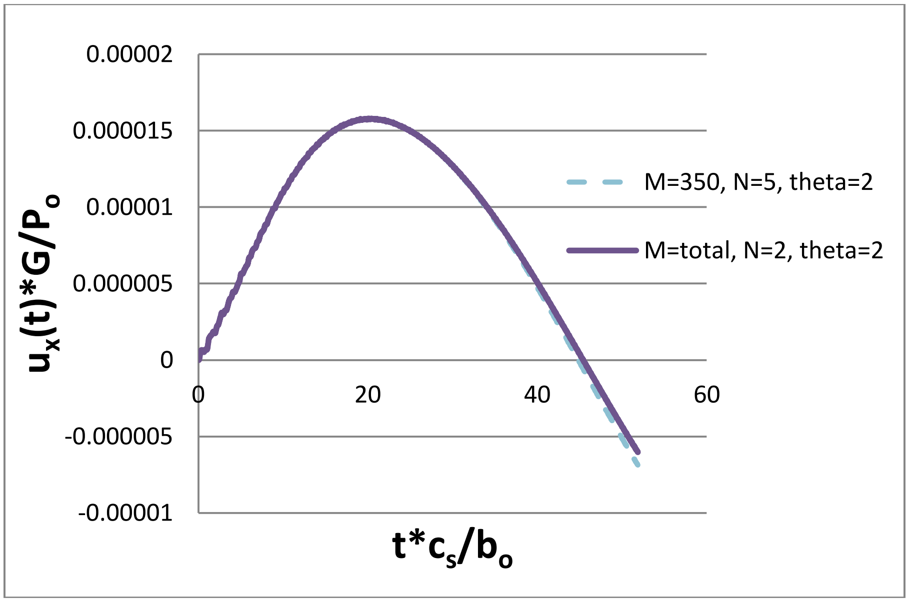


Fig.(5.56): Horizontal displacement of point A for M=350, N=5 and M=total, N=2

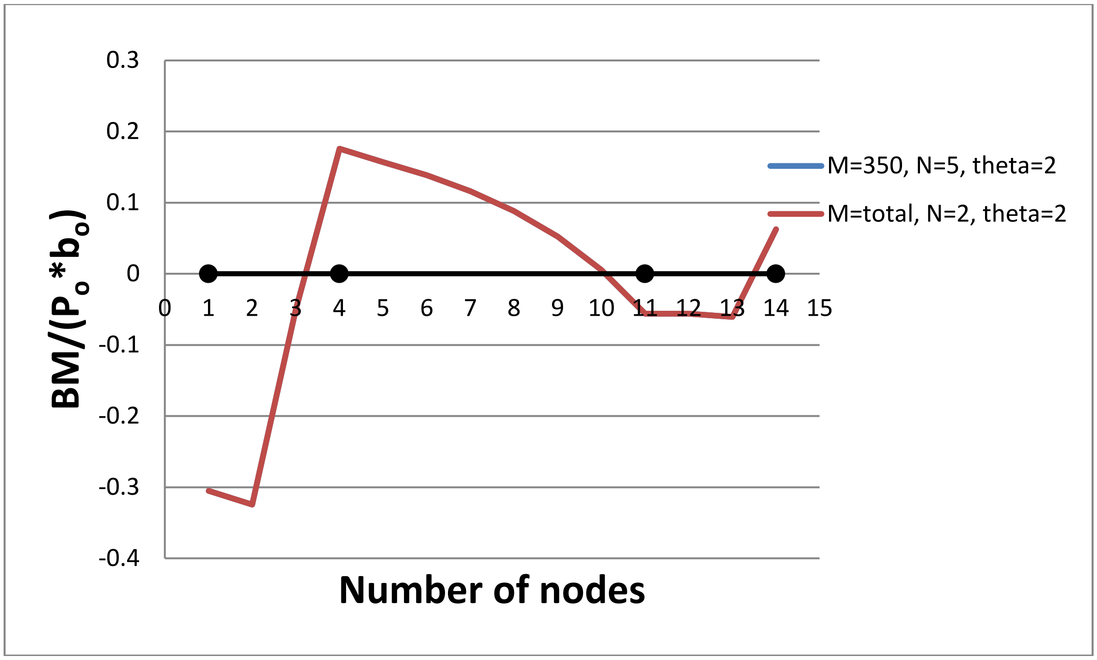


Fig.(5.57): Bending moment of the tunnel for M=350, N=5 and M=total, N=2

Chapter 6: Conclusions

6.1 Summary

In this thesis the SBFE method is derived in details and a numerical method is stated to solve the unit impulse equation and to get the domain response (chapter 2). The computer program used to solve the equations is described in chapter 3. In chapter 4 some benchmark problems are described for validation of the program, and in chapter 5 some practical problems are solved to see the application of the method in real life problems.

6.2 Conclusion

It can be concluded from the work in this thesis that the SBFEM is efficient in modeling problems of unbounded domain, where a good agreement with the exact solution of the benchmark problems and a good agreement with the results of the extended FE mesh in case of absence of the exact solution is achieved.

Also the method shows efficiency even for large value of N, and also when truncating the time of calculation of the unit impulse which reduces the time of analysis, as the equations calculating the unit impulse take time in calculation.

Appendix A Computer program

A.1. The MATLAB code

The code of the main program

```
%Dynamic Analysis of Unbounded Domin
%A program to solve unbounded doamin subjected to dynamic load using
SBFEM
disp('***************************DAUD****************************');
clear;
%Material properties
matno=input('Enter the number of materials: ');
clear Em;clear vm;clear rohm;clear G;clear csm;clear cpm;
for i=1:matno
    disp('For material number:');disp(i);
    Em(i)=input('Enter Young`s modulus: ');%vector containing modulus of
elasticity for each material
    vm(i)=input('Enter Poisson`s ratio: ');%vector containing Poisson
ratio for each material
    rohm(i)=input('Enter medium density: ');%vector containing density
of each material
    G(i)=Em(i)/(2*(1+vm(i)));%vector containing shear modulus for each
material
    csm(i)=sqrt(G(i)/rohm(i));%vector containing the shear velocity for
each material
    cpm(i)=sqrt(Em(i)*(1-vm(i))/(rohm(i)*(1+vm(i))*(1-2*vm(i))));%vector
containing the dailation velocity for each material
end
dim=menu('choose the type problem:', '2D', '3D');
switch dim
    case 1 %2D problems
        s=2;%Spatial dimension
        motiontype=menu('choose the type of motion:', 'In-plane', 'Out-
of-plane');
        switch motiontype
            case 1 %In plane problem
                motiondir=2;
            case 2 %Out of plane problem
                motiondir=1;
        end
    case 2 %3D problems
        s=3;
        motiondir=3;motiontype=0;
end
no_ele_type_unb=input('Enter the the number of element types for
unbounded domain: ');
no_ele_type_b=input('Enter the the number of element types for bounded
domain: ');
disp('Types of elements for unbounded:')
switch dim
    case 1 %2D problems
```

```
        disp('The types are: L2, L3: ')
    case 2 %3D problems
        disp('The types are: Q4, Q8, Q9: ')
end
clear ele_type_unb;
ele_type_unb=zeros(1,2,no_ele_type_unb);%initializing ele_type_unb
for i=1:no_ele_type_unb
    disp('For element type number:');disp(i);
    ele_type_unb(1,:,i)=input('Enter the element type: ','s');%A row
vector contain the type of elements used in unbounded
end
disp('Types of elements for bounded:')
switch dim
    case 1 %2D problems
        disp('The types are: F2, Q4, Q8, Q9 ')
    case 2 %3D problems
        disp('The types are:  ')
end
clear ele_type_b;
[ele_type_b]=zeros(1,2,no_ele_type_b);%initializing ele_type_b
numberonodeframe=0;line=0;
for i=1:no_ele_type_b
    disp('For element type number:');disp(i);
    ele_type_b(1,:,i)=input('Enter the element type: ','s');%A row
vector contain the type of elements used in bounded
    if strcmp(char(ele_type_b(1,:,i)),'F2')==1
        line=strcmp(char(ele_type_b(1,:,i)),'F2');
        numberonodeframe=input('Enter the number of frame nodes: ');
    end
end
max_node_per_elem=input('Enter the maximum node per element: ');
numberelements=input('Enter the number of elements: ');
%input the structure coordinates (anti-clockwise)
%Initializing
clear nodeCoordinates;clear Coordinates
nodeCoordinates=zeros(max_node_per_elem,s,numberelements);%initializing
nodeCoordinates
disp('Loading name of nodes.....')
%nodename(i,j) : name of the ith node in the jth element
%nodename=xlsread('coordinates.xlsx',-1);
nodename=load('nodename.txt');
numberonode=input('Enter the total number of nodes: ');
numberonodeB=input('Enter the number of nodes for bounded domain: ');
numberonodeUB=input('Enter the number of nodes for Unbounded domain: ');
disp('Loading the problem coordinates:')
%Coordinates=xlsread('coordinates.xlsx',-1);%Problem coordinates
Coordinates=load('Coordinates.txt');%Problem coordinates
for e=1:numberelements
    for i=1:max_node_per_elem
        if nodename(i,e)==0
            break;
        else
        nodeCoordinates(i,:,e)=Coordinates(nodename(i,e),:);
        end
```

```
    end
end
if matno==1
    for i=1:numberelements
        material(i)=1;
    end
else
    disp('Loading material for each element....')
    %material=xlsread('coordinates.xlsx',-1);%material for each element
    material=load('material.txt');%material for each element
end
clear type_ele_unb;clear type_ele_b;
disp('Loading type of element for each element....')
type_ele_unb=load('type_ele_unb.txt');%A row vector containing numbers
the corresponds to element type assigned for each element for unbounded
type_ele_b=load('type_ele_b.txt');%A row vector containing numbers the
corresponds to element type assigned for each element for bounded
clear D;clear E;clear v;clear roh;clear cs;clear cp;
switch dim
    case 1
        for e=1:numberelements
            E(e)=Em(material(e));%vector containing modulus of
elasticity for each element
            v(e)=vm(material(e));%vector containing Poisson ratio for
each material
            roh(e)=rohm(material(e));%vector containing density of each
material
            cs(e)=csm(material(e));%vector containing shear velocity per
element
            cp(e)=cpm(material(e));%vector containing dilational
velocity per element
            if e>size(type_ele_unb)
                if strcmp(ele_type_b(:,:,type_ele_b(e-
size(type_ele_unb,2))),'F2')==1
                    continue
                else
                    [D(:,:,e)]=elasticitymat2(motiontype,E(e),v(e));
                end
            else
                [D(:,:,e)]=elasticitymat2(motiontype,E(e),v(e));
            end
        end
    case 2
        for e=1:numberelements
            E(e)=Em(material(e));%vector containing modulus of
elasticity for each element
            v(e)=vm(material(e));%vector containing Poisson ratio for
each material
            roh(e)=rohm(material(e));%vector containing density of each
material
            cs(e)=csm(material(e));%vector containing shear velocity per
element
            cp(e)=cpm(material(e));%vector containing dilational
velocity per element
```

```
            [D(:,:,e)]=elasticitymat3(E(e),v(e));
        end
end
disp('Loading the name of the elements of the bounded substructure....')
%elem_boun=xlsread('coordinates.xlsx',-1);
elem_boun=load('elem_boun.txt'); %A row vector containing the names of
elements in bounded domain
disp('Loading the name of the elements of the unbounded
substructure....')
%elem_unboun=xlsread('coordinates.xlsx',-1);
elem_unboun=load('elem_unboun.txt'); %A row vector containing the names
of elements in unbounded domain
clear Eo;clear E1;clear E2;clear Mo;clear CI;clear eleDof;
%eleDof is a matrix of number of rows equal the number of elements and
number of columns equal the number of the largest Dofs of the system
% It contains the Dof for each element
[Eo,E1,E2,Mo,CI,eleDof]=SBFE_Const(elem_unboun,max_node_per_elem,nodenam
e,dim,motiontype,...

numberonodeUB,motiondir,D,roh,nodeCoordinates,cp,cs,...
                                   type_ele_unb,ele_type_unb);
%A loop to renumber the Dof of the SBFEs according to the whole system
not only to the Unbounded system
for i=1:size(eleDof,1)
    for j=2:2:size(eleDof,2)
        if eleDof(i,j)==0
            break;
        else
            eleDof(i,j)=eleDof(i,j-1)+numberonode;
        end
    end
end
Dof1=size(Eo,1); %Dof of the unbounded domain before eliminating the
restraint Dofs
clear K;clear Mass;clear C;clear Kf;clear Mf;clear framelename;clear
nodenameframe;
[K,Mass,Kf,Mf,eleDof,framelename,T,nodenameframe]=Stiff_Mass(elem_unboun
,elem_boun,max_node_per_elem,nodename,dim,motiontype,...

numberonode,motiondir,D,E,roh,nodeCoordinates,type_ele_b,ele_type_b,...

line,eleDof,numberonodeframe);
Dof3=size(K,1); %Dof of the whole structure before eliminating the
restraint Dofs
C=zeros(Dof3);
%Get a row vector containing the names of rhe Dofs of the system
clear dofname;clear dofname1;
dofname=zeros(1,Dof3);
k=1;
if line==1 %A frame element exist
    a=0;
    for i=1:numberonode
        for l=1:size(nodenameframe,2)
            if i==nodenameframe(l)
```

```
                a=1;
                break;
            else
                a=0;
            end
        end
        for j=1:motiondir+a
            if a==1 && j==motiondir+a
                dofname(k)=motiondir*numberonode+1;
            else
                dofname(k)=i+(j-1)*numberonode;
            end
            k=k+1;
        end
    end
else %No frame element exist
    for i=1:numberonode
        for j=1:motiondir
            dofname(k)=i+(j-1)*numberonode;
            k=k+1;
        end
    end
end
dofname1=dofname;
nt=input('Enter the time range: ');
dt=input('Enter the time step: ');
d_t=input('Enter the time step for calculation of the unit impulse: ');
Nn=round(d_t/dt);%number of time steps between 2 consecutive values of
unit impulse
%Input of the applied forces
%P : total force (concentrated + distributed)
numberoload=input('Enter the number of load values: ');
clear P;
[P]=ForceODisp(numberoload,numberonode,nt,dt,dim,motiontype,motiondir,..
.
               Coordinates,nodeCoordinates,nodename,s,elem_unboun,...
               type_ele_b,ele_type_b,type_ele_unb,ele_type_unb,...
               line,numberonodeframe,nodenameframe);
%Definiing the restriaints on the structure
BC=menu('Is there any restraints: ','Yes','No');
clear restraints1;clear restraints2;clear restraints3;clear restraints;
switch BC
    case 1 %Restraints exists
        disp('Enter the restriants at each node')
        disp('For no restraint enter 0 and for the opposite enter 1')
        %restraints : row matrix contains the restraints in X,Y and
        %Z directions
        disp('Loading the system restraints....')
        %restraints=xlsread('coordinates.xlsx',-1);
        restraints=load('restraints.txt'); %A row vector containing the
restraints of
        %the system as 1 for restraint direction and 0 for unrestraint
direction
        restraints1(1,:)=restraints(1,1:motiondir*numberonodeUB);
```

```
        restraints3(1,:)=restraints(1,:);
        nor=0;%Number of restraints
        norU=0;%Number of restraints for unbounded domain
        for i=1:Dof3
            if restraints(i)==1
                nor=nor+1;
            end
        end
        o=Dof3;
        h=1;
        %Eliminating rows and columns in the matrices according to the
restraints
        for j=1:nor
            for i=h:o
                if restraints(i)==1
                    K(:,i)=[];K(i,:)=[];
                    Mass(:,i)=[];Mass(i,:)=[];
                    C(:,i)=[];C(i,:)=[];
                    P(i,:)=[];
                    restraints(i)=[];
                    o=o-1;
                    break;
                end
            end
            h=i;
        end
        o=motiondir*numberonodeUB;
        for i=1:o
            if restraints1(i)==1
                norU=norU+1;
            end
        end
        clear restraints2;
        h=1;restraints2(1,:)=restraints1(1,:);
        %Eliminating rows and columns in the matrices according to the
restraints
        for j=1:norU
            for i=h:o
                if restraints1(i)==1
                    Eo(:,i)=[];Eo(i,:)=[];
                    E1(:,i)=[];E1(i,:)=[];
                    E2(:,i)=[];E2(i,:)=[];
                    Mo(:,i)=[];Mo(i,:)=[];
                    CI(:,i)=[];CI(i,:)=[];
                    restraints1(i)=[];
                    o=o-1;
                    break;
                end
            end
            h=i;
        end
    case 2 %No restraints exists
end
```

```
Dof2=size(Eo,1); %number Dofs for the unbounded after eliminating the
restraint Dof
Dof=size(K,1); %number Dofs for the whole structure after eliminating
the restraint Dof
th=input('Enter the value of the extrapolation parameter: ');
clear MI;clear MMI;
[MI,MMI,M]=Unit_impulse_L(CI,Eo,E1,E2,Mo,nt,d_t,s,Dof2,th,1); %Unit
impulse response function for acceleration
%A loop modifying the MI matrix for rigid interface
behaviourtyp=menu('The type of behviour of the interface and structure
is','Flexible','Rigid');
switch behaviourtyp
    case 1
    case 2
        clear MR;clear MassR;clear KR;clear CR;clear PR;
        for i=1:round((nt*(1/d_t))+1)
            for j=1:motiondir*numberonode
                for k=1:motiondir*numberonode
                    MR(j,k,i)=0;
                    if k<=motiondir
                        MR1(j,k,i)=0;
                    end
                    if k<=motiondir
                        if j<=motiondir
                            MR2(j,k,i)=0;
                        end
                    end
                end
            end
        end
        for j=1:motiondir*numberonode
            for k=1:motiondir*numberonode
                MassR(j,k)=0;
                KR(j,k)=0;
                CR(j,k)=0;
                if k<=motiondir
                    MassR1(j,k)=0;
                    KR1(j,k)=0;
                    CR1(j,k)=0;
                end
                if k<=motiondir
                    if j<=motiondir
                        MassR1(j,k)=0;
                        KR1(j,k)=0;
                        CR1(j,k)=0;
                    end
                end
            end
        end
        for i=1:round((nt*(1/dt))+1)
            for j=1:motiondir*numberonode
                PR(j,i)=0;
                if j<=motiondir
                    PR1(j,i)=0;
```

```
                end
            end
        end
        for i=1:round((nt*(1/d_t))+1)
            for j=1:Dof
                for k=1:Dof
                    MR(dofname(j),dofname(k),i)=MI(j,k,i);
                end
            end
        end
        for j=1:Dof
            for k=1:Dof
                MassR(dofname(j),dofname(k))=Mass(j,k);
                KR(dofname(j),dofname(k))=K(j,k);
                CR(dofname(j),dofname(k))=C(j,k);
            end
        end
        for i=1:round((nt*(1/dt))+1)
            for j=1:Dof
                PR(dofname(j),i)=P(j,i);
            end
        end
        for i=1:round((nt*(1/d_t))+1)
            for k=2:numberonode
                MR1(:,1,i)=MR(:,k,i)+MR(:,k-1,i)+MR1(:,1,i);
                if motiondir==2
                    MR1(:,2,i)=MR(:,k+numberonode,i)+MR(:,(k-
1)+numberonode,i)+MR1(:,2,i);
                elseif motiondir==3
                    MR1(:,2,i)=MR(:,k+numberonode,i)+MR(:,(k-
1)+numberonode,i)+MR1(:,2,i);
                    MR1(:,3,i)=MR(:,k+(2*numberonode),i)+MR(:,(k-
1)+(2*numberonode),i)+MR1(:,3,i);
                end
            end
            for j=2:numberonode
                MR2(1,:,i)=MR1(j,:,i)+MR1(j-1,:,i)+MR2(1,:,i);
                if motiondir==2
                    MR2(2,:,i)=MR1(j+numberonode,:,i)+MR1((j-
1)+numberonode,:,i)+MR2(2,:,i);
                elseif motiondir==3
                    MR2(2,:,i)=MR1(j+numberonode,:,i)+MR1((j-
1)+numberonode,:,i)+MR2(2,:,i);
                    MR2(3,:,i)=MR1(j+(2*numberonode),:,i)+MR1((j-
1)+(2*numberonode),:,i)+MR2(3,:,i);
                end
            end
        end
        for k=2:numberonode
            MassR1(:,1)=MassR(:,k)+MassR(:,k-1)+MassR1(:,1);
            KR1(:,1)=KR(:,k)+KR(:,k-1)+KR1(:,1);
            CR1(:,1)=CR(:,k)+CR(:,k-1)+CR1(:,1);
            if motiondir==2
```

```
                MassR1(:,2)=MassR(:,k+numberonode)+MassR(:,(k-
1)+numberonode)+MassR1(:,2);
                KR1(:,2)=KR(:,k+numberonode)+KR(:,(k-
1)+numberonode)+KR1(:,2);
                CR1(:,2)=CR(:,k+numberonode)+CR(:,(k-
1)+numberonode)+CR1(:,2);
            elseif motiondir==3
                MassR1(:,2)=MassR(:,k+numberonode)+MassR(:,(k-
1)+numberonode)+MassR1(:,2);
                MassR1(:,3)=MassR(:,k+(2*numberonode))+MassR(:,(k-
1)+(2*numberonode))+MassR1(:,3);
                KR1(:,2)=KR(:,k+numberonode)+KR(:,(k-
1)+numberonode)+KR1(:,2);
                KR1(:,3)=KR(:,k+(2*numberonode))+KR(:,(k-
1)+(2*numberonode))+KR1(:,3);
                CR1(:,2)=CR(:,k+numberonode)+CR(:,(k-
1)+numberonode)+CR1(:,2);
                CR1(:,3)=CR(:,k+(2*numberonode))+CR(:,(k-
1)+(2*numberonode))+CR1(:,3);
            end
        end
        for j=2:numberonode
            MassR2(1,:)=MassR1(j,:)+MassR1(j-1,:)+MassR2(1,:);
            KR2(1,:)=KR1(j,:)+KR1(j-1,:)+KR2(1,:);
            CR2(1,:)=CR1(j,:)+CR1(j-1,:)+CR2(1,:);
            if motiondir==2
                MassR2(2,:)=MassR1(j+numberonode,:)+MassR1((j-
1)+numberonode,:)+MassR2(2,:);
                KR2(2,:)=KR1(j+numberonode,:)+KR1((j-
1)+numberonode,:)+KR2(2,:);
                CR2(2,:)=CR1(j+numberonode,:)+CR1((j-
1)+numberonode,:)+CR2(2,:);
            elseif motiondir==3
                MassR2(2,:)=MassR1(j+numberonode,:)+MassR1((j-
1)+numberonode,:)+MassR2(2,:);
                MassR2(3,:)=MassR1(j+(2*numberonode),:)+MassR1((j-
1)+(2*numberonode),:)+MassR2(3,:);
                KR2(2,:)=KR1(j+numberonode,:)+KR1((j-
1)+numberonode,:)+KR2(2,:);
                KR2(3,:)=KR1(j+(2*numberonode),:)+KR1((j-
1)+(2*numberonode),:)+KR2(3,:);
                CR2(2,:)=CR1(j+numberonode,:)+CR1((j-
1)+numberonode,:)+CR2(2,:);
                CR2(3,:)=CR1(j+(2*numberonode),:)+CR1((j-
1)+(2*numberonode),:)+CR2(3,:);
            end
        end
        for i=1:round((nt*(1/dt))+1)
            for j=2:numberonode
                PR1(1,i)=PR(j,i)+PR(j-1,i)+PR1(1,i);
                if motiondir==2
                    PR1(2,i)=PR(j+numberonode,i)+PR((j-
1)+numberonode,i)+PR1(2,i);
                elseif motiondir==3
```

```
                    PR1(2,i)=PR(j+numberonode,i)+PR((j-
1)+numberonode,i)+PR1(2,i);
                    PR1(3,i)=PR(j+(2*numberonode),i)+PR((j-
1)+(2*numberonode),i)+PR1(3,i);
                end
            end
        end
        clear MI;clear Mass;clear K;clear C;clear P;
        MI=MR2;
        Mass=MassR2;
        K=KR2;
        C=CR2;
        P=PR1;
        Dof=motiondir;
end
%Calculation of acceleration of the domain under force P
%ua,uv,ud : are medium acceleration, velocity and displacement
clear ud;clear uv;clear ua;
[ud,uv,ua]=disp_vel_acc(MI,Mass,K,C,P,nt,dt,Nn,M,Dof,Dof2); %Function to
calculate Displacement, velocity and acceleration
sa_element=menu('Do you want to calculate the straining actions
for:','some Plane element','some Frame element','No element');
switch sa_element
    case {1,2}
        %The loop insert the ua,uv and ud in ua1,uv1 and ud1 such that
the remooved rows are placed by zeros
        switch BC
            case 1 %A restraint exists
                disp('Insert the removed rows in the disp., vel., and
acc. matrices')
                clear ua1;clear uv1;clear ud1;clear P1;
                [ua1]=zeros(Dof3,round((nt*(1/dt))+1));%initializing
acceleration without eliminating the restraint Dofs
                [uv1]=zeros(Dof3,round((nt*(1/dt))+1));%initializing
velocity without eliminating the restraint Dofs
                [ud1]=zeros(Dof3,round((nt*(1/dt))+1));%initializing
displacement without eliminating the restraint Dofs
                [P1]=zeros(Dof3,round((nt*(1/dt))+1));%initializing
applied force without eliminating the restraint Dofs
                switch dim
                    case 1 %2D problems
                        switch motiontype
                            case 1 %In plane problems
                                j=0;
                                for i=1:Dof3
                                    if restraints3(i)==0
                                       j=j+1;
                                       ua1(i,:)=ua(j,:);
                                       uv1(i,:)=uv(j,:);
                                       ud1(i,:)=ud(j,:);
                                       P1(i,:)=P(j,:);
                                    end
                                end
                            case 2 %Out of plane problem
```

```
                    end
                case 2 %3D problem
                    %Rows
                    %Columns
            end
        case 2 %No restraint
            clear ua1;clear uv1;clear ud1;
            ua1=ua;
            uv1=uv;
            ud1=ud;
            P1=P;
    end
    disp('Rearrange the disp., vel. and acc. as all X then Y then
Z')
    clear ua2;clear uv2;clear ud2;clear P2;
    [ua2]=zeros(Dof3,round((nt*(1/dt))+1));%initializing
acceleration without eliminating the restraint Dofs
    [uv2]=zeros(Dof3,round((nt*(1/dt))+1));%initializing velocity
without eliminating the restraint Dofs
    [ud2]=zeros(Dof3,round((nt*(1/dt))+1));%initializing
displacement without eliminating the restraint Dofs
    [P2]=zeros(Dof3,round((nt*(1/dt))+1));%initializing applied
force without eliminating the restraint Dofs
    %The loop inserts ua1, uv1, ud1 and P1 in ua2, uv2, ud2 and P2
in the order all X
    %then all Y then all Z
    for i=1:Dof3
        ua2(dofname1(i),:)=ua1(i,:);
        uv2(dofname1(i),:)=uv1(i,:);
        ud2(dofname1(i),:)=ud1(i,:);
        P2(dofname1(i),:)=P1(i,:);
    end
    %Insert the acceleration, velocity and displacement of specified
elements in a
    %separate matrices.
    if sa_element==1
        clear uae;clear uve;clear ude;clear Pe;
        disp('Loading the plane or SB elements at which the internal
forces are needed')
        ele_sa_plane=load('ele_sa_plane.txt');

[uae]=zeros(size(eleDof,2),round((nt*(1/dt))+1),size(ele_sa_plane,2));%i
nitializing element acceleration

[uve]=zeros(size(eleDof,2),round((nt*(1/dt))+1),size(ele_sa_plane,2));%i
nitializing element velocity

[ude]=zeros(size(eleDof,2),round((nt*(1/dt))+1),size(ele_sa_plane,2));%i
nitializing element displacement

[Pe]=zeros(size(eleDof,2),round((nt*(1/dt))+1),size(ele_sa_plane,2));%in
itializing element applied force
        for e=1:size(ele_sa_plane,2)
            for j=1:size(eleDof,2)
```

```
                    if eleDof(ele_sa_plane(e),j)==0
                        continue;
                    end
                    uae(j,:,e)=ua2(eleDof(ele_sa_plane(e),j),:);
                    uve(j,:,e)=uv2(eleDof(ele_sa_plane(e),j),:);
                    ude(j,:,e)=ud2(eleDof(ele_sa_plane(e),j),:);
                    Pe(j,:,e)=P2(eleDof(ele_sa_plane(e),j),:);
                end
            end
        end
        %Calculate the frame elements straining actions
        if line==1 %A line element exist in the model
            if sa_element==2
                clear uafe;clear uvfe;clear udfe;clear Pfe;
                disp('Loading the frame elements at which the internal
forces are needed')
                ele_sa_frame=load('ele_sa_frame.txt');

[uafe]=zeros(6,round((nt*(1/dt))+1),size(ele_sa_frame,2));%initializing
element acceleration
 [uvfe]=zeros(6,round((nt*(1/dt))+1),size(ele_sa_frame,2));%initializing
element velocity
[udfe]=zeros(6,round((nt*(1/dt))+1),size(ele_sa_frame,2));%initializing
element displacement
[Pfe]=zeros(6,round((nt*(1/dt))+1),size(ele_sa_frame,2));%initializing
element applied force
                for e=1:size(ele_sa_frame,2)
                    for j=1:6
                        if eleDof(ele_sa_frame(e),j)==0
                            continue;
                        end
                        uafe(j,:,e)=ua2(eleDof(ele_sa_frame(e),j),:);
                        uvfe(j,:,e)=uv2(eleDof(ele_sa_frame(e),j),:);
                        udfe(j,:,e)=ud2(eleDof(ele_sa_frame(e),j),:);
                        Pfe(j,:,e)=P2(eleDof(ele_sa_frame(e),j),:);
                    end
                end
                disp('Calculate the frame elements straining actions')
                clear Fe;Fe=0;
[Fe]=zeros(6,round((nt*(1/dt))+1),size(framelename,2));%initializing
element straining action
                %Calculate of the straining actions
                for e=1:size(framelename,2)
                    for i=1:round((nt*(1/dt))+1)
                       Fe(:,i,e)=Mf(:,:,e)*uafe(:,i,e)+Kf(:,:,e)*udfe(:,i,e);
                        Fe(:,i,e)=T(:,:,e)*Fe(:,i,e);
                    end
                end
            end
        end
    case 3
end
```

Appendix B

Prove of the relation: $\left(|J|[b_2]\right)_{,\eta} + \left(|J|[b_3]\right)_{,\zeta} = -2|J[b_1]$

From equations (2.9),(2.17),(2.18),(2.20),(2.21) and (2.24)

$$|J|[b_2] = \begin{bmatrix} y_{,\zeta}z - yz_{,\zeta} & 0 & 0 \\ 0 & xz_{,\zeta} - x_{,\zeta}z & 0 \\ 0 & 0 & x_{,\zeta}y - xy_{,\zeta} \\ 0 & x_{,\zeta}y - xy_{,\zeta} & xz_{,\zeta} - x_{,\zeta}z \\ x_{,\zeta}y - xy_{,\zeta} & 0 & y_{,\zeta}z - yz_{,\zeta} \\ xz_{,\zeta} - x_{,\zeta}z & y_{,\zeta}z - yz_{,\zeta} & 0 \end{bmatrix} \tag{B.1}$$

$$|J|[b_3] = \begin{bmatrix} yz_{,\eta} - y_{,\eta}z & 0 & 0 \\ 0 & x_{,\eta}z - xz_{,\eta} & 0 \\ 0 & 0 & xy_{,\eta} - x_{,\eta}y \\ 0 & xy_{,\eta} - x_{,\eta}y & x_{,\eta}z - xz_{,\eta} \\ xy_{,\eta} - x_{,\eta}y & 0 & yz_{,\eta} - y_{,\eta}z \\ x_{,\eta}z - xz_{,\eta} & yz_{,\eta} - y_{,\eta}z & 0 \end{bmatrix} \tag{B.2}$$

Calculate equation $\left(|J|[b_2]\right)_{,\eta} + \left(|J|[b_3]\right)_{,\zeta}$:

$$\left(y_{,\zeta}z - yz_{,\zeta}\right)_{,\eta} + \left(yz_{,\eta} - y_{,\eta}z\right)_{,\zeta} = -2\left(y_{,\eta}z_{,\zeta} - z_{,\eta}y_{,\zeta}\right) \tag{B.3}$$

$$\left(xz_{,\zeta} - x_{,\zeta}z\right)_{,\eta} + \left(x_{,\eta}z - xz_{,\eta}\right)_{,\zeta} = -2\left(x_{,\zeta}z_{,\eta} - x_{,\eta}z_{,\zeta}\right) \tag{B.4}$$

$$\left(x_{,\zeta}y - xy_{,\zeta}\right)_{,\eta} + \left(xy_{,\eta} - x_{,\eta}y\right)_{,\zeta} = -2\left(x_{,\eta}y_{,\zeta} - x_{,\zeta}y_{,\eta}\right) \tag{B.5}$$

Therefore,

$$\left(|J|[b_2]\right)_{,\eta} + \left(|J|[b_3]\right)_{,\zeta} = -2\begin{bmatrix} y_{,\eta}z_{,\zeta} - z_{,\eta}y_{,\zeta} & 0 & 0 \\ 0 & x_{,\zeta}z_{,\eta} - x_{,\eta}z_{,\zeta} & 0 \\ 0 & 0 & x_{,\eta}y_{,\zeta} - x_{,\zeta}y_{,\eta} \\ 0 & x_{,\eta}y_{,\zeta} - x_{,\zeta}y_{,\eta} & x_{,\zeta}z_{,\eta} - x_{,\eta}z_{,\zeta} \\ x_{,\eta}y_{,\zeta} - x_{,\zeta}y_{,\eta} & 0 & y_{,\eta}z_{,\zeta} - z_{,\eta}y_{,\zeta} \\ x_{,\zeta}z_{,\eta} - x_{,\eta}z_{,\zeta} & y_{,\eta}z_{,\zeta} - z_{,\eta}y_{,\zeta} & 0 \end{bmatrix} = -2|J|[b_1] \tag{B.6}$$

REFERENCES

[1] Lysmer, J. and Kuhlemeyer, R.L., "Finite dynamic model for infinite media", ASCE J. of the Engineering Mechanics division, 95, pp.859-877, 1969.

[2] Wolf, J.P., Dynamic Soil-Structure Interaction, Prentice Hall, Englewood Cliffs, NJ, 1985.

[3] Astley, R.J., Finite Elements in Solids and Structures, Chapman & Hall, London, 1992.

[4] Wolf, J.P. and Song, C., "Unit impulse response matrix of unbounded medium by infinitesimal finite-element cell method", Computer methods in applied mechanics and engineering, 122, pp.251-272, 1995.

[5] Wolf, J.P. and Song, C., "Consistent infinitesimal finite-element cell method: in plane motion", Computer methods in applied mechanics and engineering, 123, pp.355-370, 1995.

[6] Song, C. and Wolf, J.P., "Consistent infinitesimal finite-element cell method: Three-dimensional vector wave equation", International journal of numerical methods in engineering, 39, pp.2189-2208, 1996.

[7] Wolf, J.P. and Song, C., Finite-element Modeling of Unbounded Media, John Wiley and Sons, Chichester, 1996.

[8] Song, C. and Wolf, J.P., "The scaled boundary finite-element method – alias consistent infinitesimal finite-element cell method – for elastodynamics", Computer methods in applied mechanics and engineering, 147, pp.329-355, 1997.

[9] Wolf, J.P. and Song, C., "Unit impulse response of unbounded medium by scaled boundary finite-element method", Computer methods in applied mechanics and engineering, 159, pp.355-367, 1998.

[10] Song, C. and Wolf, J.P., "The scaled boundary finite-element method: analytical solution in frequency domain", Computer methods in applied mechanics and engineering, 164, pp.249-264, 1998.

[11] Wolf, J.P. and Song, C., "The scaled boundary finite-element method – a primer: derivation", Computers and structures, 78, pp.191-210, 2000.

[12] Song, C. and Wolf, J.P., “The scaled boundary finite-element method – a primer: solution procedures”, Computers and structures, 78, pp.211-225, 2000.

[13] James,P.D. and Andrew, J.D., “Scaled boundary finite element analysis of a non-homogeneous axisymmetric domain subjected to general loading”, International journal of numerical and analytical methods in geomechanics, 27, pp.813-835, 2003.

[14] Wolf, J.P., The scaled boundary finite-element method, John Wiley and Sons, England, 2003.

[15] Chopra, A., Dynamics of structures theory and applications to earthquake engineering, Prentice Hall, India, 2003.

[16] Lehmann, L., “An effective finite-element approach for soil-structure analysis in time domain”, Structural engineering and mechanics, 21, pp.1-14, 2005.

[17] Bazyar, M.H. and Song, C., “Time-harmonic response of non-homogeneous elastic unbounded domains using the scaled boundary finite-element method”, Earthquake engineering and structural dynamics, 35, pp.357-383, 2006.

[18] Bazyar, M.H. and Song, C., “Transient analysis of wave propagation in non-homogeneous elastic unbounded domains by using the scaled boundary finite-element method”, Earthquake engineering and structural dynamics, 35, pp.1787-1806, 2006.

[19] Bazyar, M.H. and Song, C., “Dynamic soil structure interaction analysis using scaled boundary finite-element method”, Ph.D.thesis, South Wales university, 2007.

[20] Verruijt, A, Soil dynamics, http://geo.verruijt.net , 2008.

[21] Liu, T., Zhao, C., Zhang, C. and Jin, F., “A semi-analytical artificial boundary for time-dependent elastic wave propagation in two-dimensional homogeneous half space”, Soil dynamics and Earthquake engineering, 30, pp.1352-1360, 2010.

[22] Radmanovic, B. and Katz, C., “A High performance scaled boundary finite element method”, IOP Conf. Series: Materials Science and Engineering, 2010.

[23] Chen, X., Birk, C., and Song, C., “Modeling of wave propagation in Unbounded domains using the Scaled Boundary Finite Element Method”, APCOM & ISCM, 2013.

[24] SAP2000 V16, 2013, Structural Analysis Program Software, Integrated Finite Element Analysis and Design of Structures, Computer and Structures, Inc., University Avenue, Berkeley, California.

[25] Liu, T. personal communication through E-mail, 2014

ARABIC SUMMARY

محتوى الرســالة

تتناول الرسالة موضوع التحليل الديناميكي للمسائل ذات الوسط الغير محدود باستخدام طريقة تحجيم حدود العناصر المحدودة ، وتحتوى الرسالة على ستة أبواب بيانها كما يلى :

الباب الأول : مقدمــة

يتضمن هذا الباب عرض للمسئلة (مسئلة الوسط الغير محدود)، مقدمة عن طريقة الحل (تحجيم حدود العناصر المحدودة) و عرض مقارنة بينها و بين طريقة العناصر المحدودة.

الباب الثانى : طريقة تحجيم حدود العناصر المحدودة

يعرض هذا الباب عرضا مفصلاً الأستنتاج الرياضي للمعادلة المستخدمة في الحل، معادلة الجساءة الديناميكية (بدلالة التردد) و الحل العددي لها بدلالة الزمن و ايضا كيفية حساب استجابة الوسط (الأزاحة و السرعة و العجلة).

الباب الثالث : البرمجة

يقدم هذا الباب شرح البرنامج الذي تم وضعه بواسطة الباحث.

الباب الرابع : أمثلة عددية لها حل رياضي معروف

يقدم هذا الباب عدد من المسائل و يتم حلها بالطريقة العددية المعروضة سابقا و مقارنة النتائج بالحل الرياضي.

الباب الخامس : أمثلة عددية عملية

يقدم هذا الباب عدة مسائل ذات الطابع العملي و يتم حلها بالطريقة العددية المعروضة سابقا و مقارنة النتائج باستخدام طريقة العناصر المحدودة.

الباب السادس: الخلاصة والاستنتاجات

يلخص هذا الباب ما تم انجازه في البحث و الأستنتاجات.

ملخص الرســالة

يقدم هذا البحث شرح لطريقة تحجيم حدود العنصر المحدودة لحل المسائل ذات الوسط الغير محدود و المعرضة لأحمال ديناميكية و تطبيقها علي مسائل التفاعل بين التربة و المنشاء. في هذا البحث تم عرض الأستنتاج الرياضي الكامل لمعادلة الجساءة الديناميكية بدلالة الزمن $M^{\infty}(t)$، و تم حل المعادلة عدديا بفرض ان علاقة الجساءة خطية في كل فترة زمنية أخذا في الأعتبار أن الجساءة الديناميكية بدلالة الزمن $M^{\infty}(t)$ علاقتها تكون خطية مع الزمن بعد فترة زمنية معينة يتم فرضها و يتم الحساب علي فترات زمنية أكبر من المستخدمة في حساب استجابة الوسط [22]، تم حساب استجابة الوسط (الازاحة، السرعة و العجلة) بحل معادلة الرابطة بين القوة و الأزاحة بدلالة الزمن باستخدام التكامل بالتجزيء داخل كل فترة زمنية [22]. أيضا تم عمل ربط بين طريقة العناصر المحدودة و طريقة تحجيم حدود العناصر المحدودة.

تم عمل برنامج باستخدام MATLAB لحل المعادلات، و حل مجموعة من المسائل التي لها حل رياضي معروف للتحقق من صحة البرنامج، و تم حل مسائل ذات الطابع العملي التي ليس لها حل رياضي و للتاكد من صحة الحل تمت المقارنة مع طريقة العناصر المحدودة. و لاحظت أن هناك تقارب في النتائج في حالة المسائل ذات الحل المعروف و أيضا المسائل العملية. و لذلك فانا طريقة تحجيم حدود العناصر المحدودة طلريقة مقبولة لحل مسائل التفاعل بين التربة و المنشاء و أيضا توفر الكثير من الوقت مقارنة بطريقة العناصر المحدودة.

التحليل الديناميكي للمسائل ذات الوسط غير المحدود بتطبيق طريقة تحجيم حدود العناصر المحدودة

إعداد

مهندس / محمد أحمد كمال عبد الخالق أحمد سليمان

رسالة مقدمة إلى كلية الهندسة بشبرا ، جامعة بنها
كجزء من متطلبات الحصول على درجة الماجستير
في الرياضيات الهندسية

يعتمد من لجنة الممتحنين: التوقيع

أ.د. عادل عبد الرحمن مجاهد ------------------
أستاذ الرياضيات التطبيقية (ممتحن خارجي)
قسم الرياضيات و الفيزيقا الهندسية
كلية الهندسة - جامعة القاهرة

أ.د. يوسف فوزى راشد ------------------
أستاذ تحليل وميكانيكا الانشاءات (ممتحن خارجي)
قسم الهندسة الانشائية
كلية الهندسة - جامعة القاهرة

أ.د. عبدالرحمن علي سعد ------------------
أستاذ الرياضيات التطبيقية (مشرف و مقررا)
قسم الرياضيات و الفيزيقا الهندسية
كلية الهندسة بشبرا - جامعة بنها

التاريخ :-----------------

كلية الهندسة بشبرا ، جامعة بنها
جمهورية مصر العربية
2015

التحليل الديناميكي للمسائل ذات الوسط غير المحدود بتطبيق طريقة تحجيم حدود العناصر المحدودة

إعداد

مهندس / محمد أحمد كمال عبد الخالق أحمد سليمان

بكالوريوس الهندسة المدنية (مرتبة الشرف) 2005 – كلية الهندسة بشبرا- جامعة بنها

رسالة مقدمة إلى كلية الهندسة بشبرا ، جامعة بنها
كجزء من متطلبات الحصول على درجة الماجستير
في الرياضيات الهندسية

تحت إشراف

أ.د. عبدالرحمن علي سعد
أستاذ الرياضيات التطبيقية
قسم الرياضيات و الفيزيقا الهندسية
كلية الهندسة بشبرا - جامعة بنها

د. عماد حسن زهران
مدرس الرياضيات التطبيقية
قسم الرياضيات و الفيزيقا الهندسية
كلية الهندسة بشبرا - جامعة بنها

كلية الهندسة بشبرا ، جامعة بنها
جمهورية مصر العربية
2015

التحليل الديناميكي للمسائل ذات الوسط غير المحدود بتطبيق طريقة تحجيم حدود العناصر المحدودة

إعداد
مهندس / محمد أحمد كمال عبد الخالق أحمد سليمان

رسالة مقدمة إلى كلية الهندسة بشبرا ، جامعة بنها
كجزء من متطلبات الحصول على درجة الماجستير
في الرياضيات الهندسية

كلية الهندسة بشبرا ، جامعة بنها
جمهورية مصر العربية
2015