**Masters of Science in Applied Mechanics**
**Speciality in Computational Mechanics**

**Year 2017/2018**

**Master Thesis**

*Diploma co-authorized by Centrale Nantes and University of Nantes*


Presented by

Emayavaramban ELANGO

Supervised by

Vam Thang Pham, PhD


on September 13, 2025

TITLE

# Finite Element Simulation of 2D Metal Strip Vibration in Hot-Dip Galvanization Process

**ENTERPRISE**


**Global R&D Maizières**
ArcelorMittal Maizières Research
Voie Romaine, BP 30320
57283 Maizières-lès-Metz
France

# Acknowledgement

I wish to express my sincere gratitude to my thesis supervisor **Mr Vam Thang Pham**, for believing in me and providing me with the necessary facilities for the project. It is a great honour to work under his supervision. I would also like to express my gratitude to all the team members of Control and Measurement Department for supporting me.

I would like to express my deepest thanks and sincere appreciation to **Prof.Nicolas Chevauegeon** for his suggestions. I am very thankful to **Prof.Panagiotis KOTRONIS** and **Prof.Grégory LEGRAIN** for their timely replies for all my doubts. I am very glad to spend my Master degree with all the professors of Centrale Nantes and for their wisdom.

I would like to express my love and gratitude to my parents, brother and my friends for their kind cooperation and understanding that the thesis is not done in a day. I would also like to thank my fellow interns **Maira Mariani Souza** and **Kenji Fabiano Okada** for making my stay more pleasant and I wish them a bright future.

# Abstract

The Main objective of the internship is to create a Finite element program to predict the vibration of a 2D metal strip in the Hot-Dip Galvanization Process. Hot-Dip galvanization is a process to coat molten zinc metal in a steel plate in order to increase the corrosion resistance. For the process of applying the zinc coating, long steel plate is dipped in a molten zinc bath and drawn out of the bath and passes between air-knife. Air knife is intended to remove excess zinc by blowing air at very high speed. Excess vibration in the steel plate causes an uneven air blow, which in turn affects the quality of the coating. The vibration can be countered to an extent by placing the electromagnets. The electromagnets operate based on the control algorithms. To create an effective control law, an efficient and accurate numerical method is required to test and validate the control law. Finite element is used as a numerical method since it is very accurate and easy to use. There are quite a few challenges in numerical modelling. The plate is very thin and it is highly stretched. The plate is also axially moving in the upward direction all the time. The Equation of motion is derived using the Hamilton Principle with the help of Euler - Lagrange formulation. Two plate theories are used to develop two distinct plate element. An element called 'PAT' triangle is created using Kirchhoff Plate theory, which is a thin plate theory. This element is $C^1$ continuous and requires complex shape function. Reissner Mindlin plate theory is formulated for both thin and thick plates. 'QUAD4' quadrangle element with 4 nodes is created using this plate theory. QUAD4 is $C^0$ continuous which makes it one of the easiest to implement. For directly solving the dynamic system, a type of Newmark time integration algorithm is implemented. Prefactorization technique is used to make it ten times faster than the regular algorithm. The FEM program is also capable of providing the FE matrices in space-state form so, that it can be solved directly in Simulink. To make the solution process even faster, The modal superposition technique is used. This technique created a reduced model from the full model. Object-oriented programming style is adapted to create a user-friendly interface. Many studies are conducted to evaluate the performance of the Elements. Elaborate mesh dependency study is also conducted. Using the knowledge from these tests, an optimised Mesh is created which performs well in most cases. FEM program is compared with the existing Galerkin method for final evaluation.

**KEYWORDS :** Hot-Dip Galvanization, Kirchhoff plate, Ressigner Mindlin Plate, Axially Moving Material, Finite Element Method

# Contents

# List of Figures

# Chapter 1

# Introduction

## 1.1 Hot-Dip Galvanization

The hot - dip galvanization process is form of galvanization process where zinc metal is coated to steel or iron metal. Steel or iron metal is immersed in a molten zinc bath and taken out. Then the metal is let to cool to create a thin layer of zinc coating. This Zinc coating prevents the metal from atmospheric corrosion and galvanic corrosion.

To galvanize the sheet metals, the sheet metals are continuously moved through several process. The over all schematics of the process is given in the figure.1.1. First the sheet metal is prepared by cleaning with chemical solutions then heated in continuous annealing furnace to get desired mechanical properties. The treated sheet metal is immersed in a molten zinc bath and passed through a air knife which removes excess zinc attached to the metal strip and also helps in regulating the thickness of the zinc layer. The metal sheet is continuously moves upward and passes the cooling fans and later to be rolled and shipped. The rollers immersed in the zinc bath, are operated under high temperature at around 500 $°C$. So, they tend to degrade in a matter of days. This results in vibration of the metal strip. The vibration of the metal strip will cause the uneven blow at both side of strip which will result in a uneven coating and degraded surface quality. One best way to control the vibration is by increasing the natural frequency. To increase the dominant natural frequency, the steel strip it is highly stressed in axial direction. Other way is to continuously monitor the vibration and the rollers are changed when needed. Frequent roller change might incur heavy production loss. To overcome this problem electromagnets are employed on both the sides of the plate. These electromagnets attract the plate to the opposite direction of deformation (figure.1.2). Proximity sensors are placed near the electromagnets and its measures the position of the plate. This signal is used to predict the control signal that is sent to the electromagnets. Correcting roller is used to control the tension on the plate by moving horizontally. Stabilizing roller is usually stationary, this guides the plate to be travelled in a precise location between the air knives.

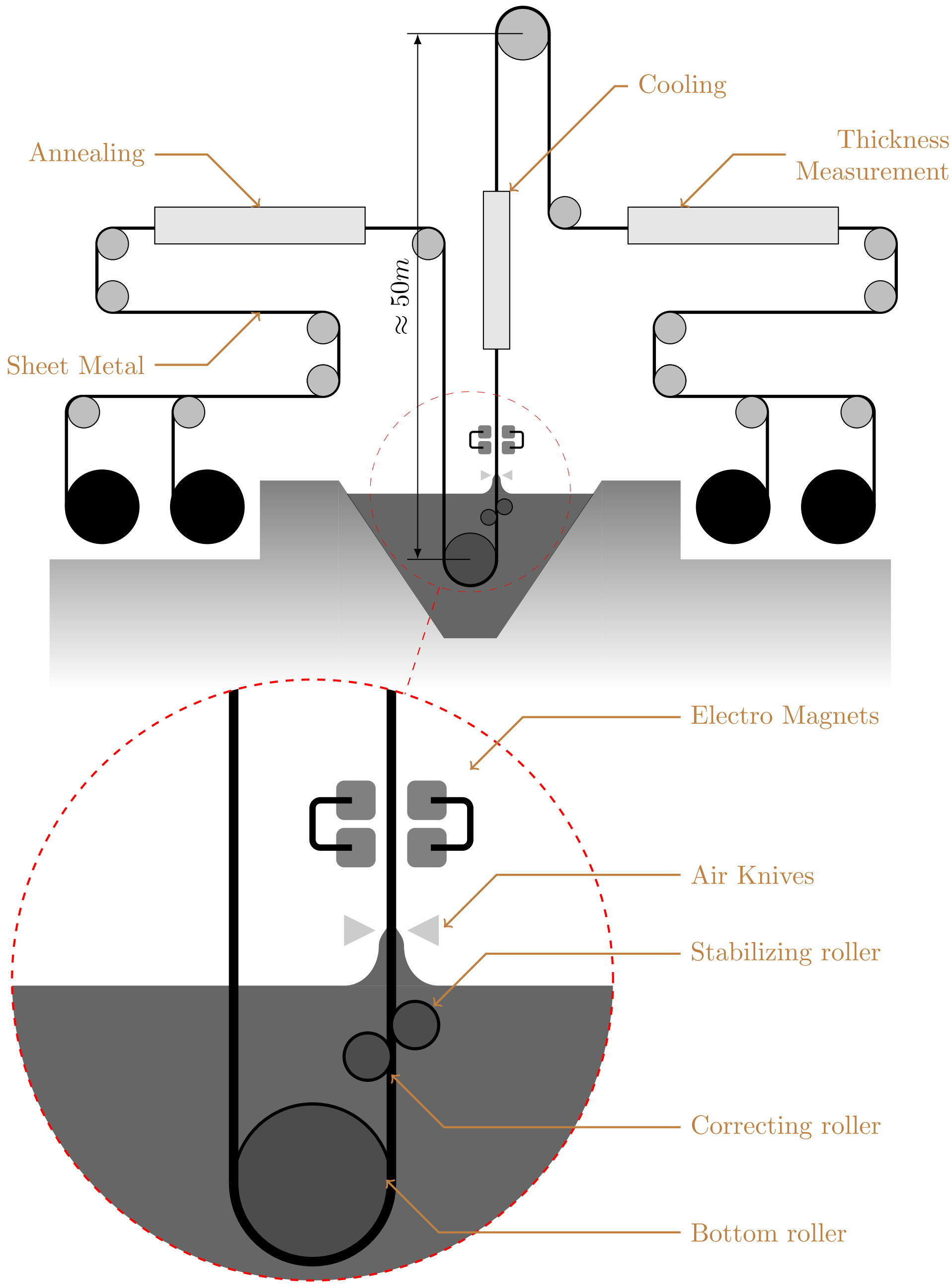



Figure 1.1: Schematics of the hot dip galvanization process

Many control algorithms techniques exist. To design and evaluate the right Control algorithm, a accurate model is required. Engineers mostly use a numerical methods to model system of which the control will be utilized . This required a reliable and accurate numerical model. Finite element method is selected as for this purpose as it is widely used and proven its efficiency for over many decades.

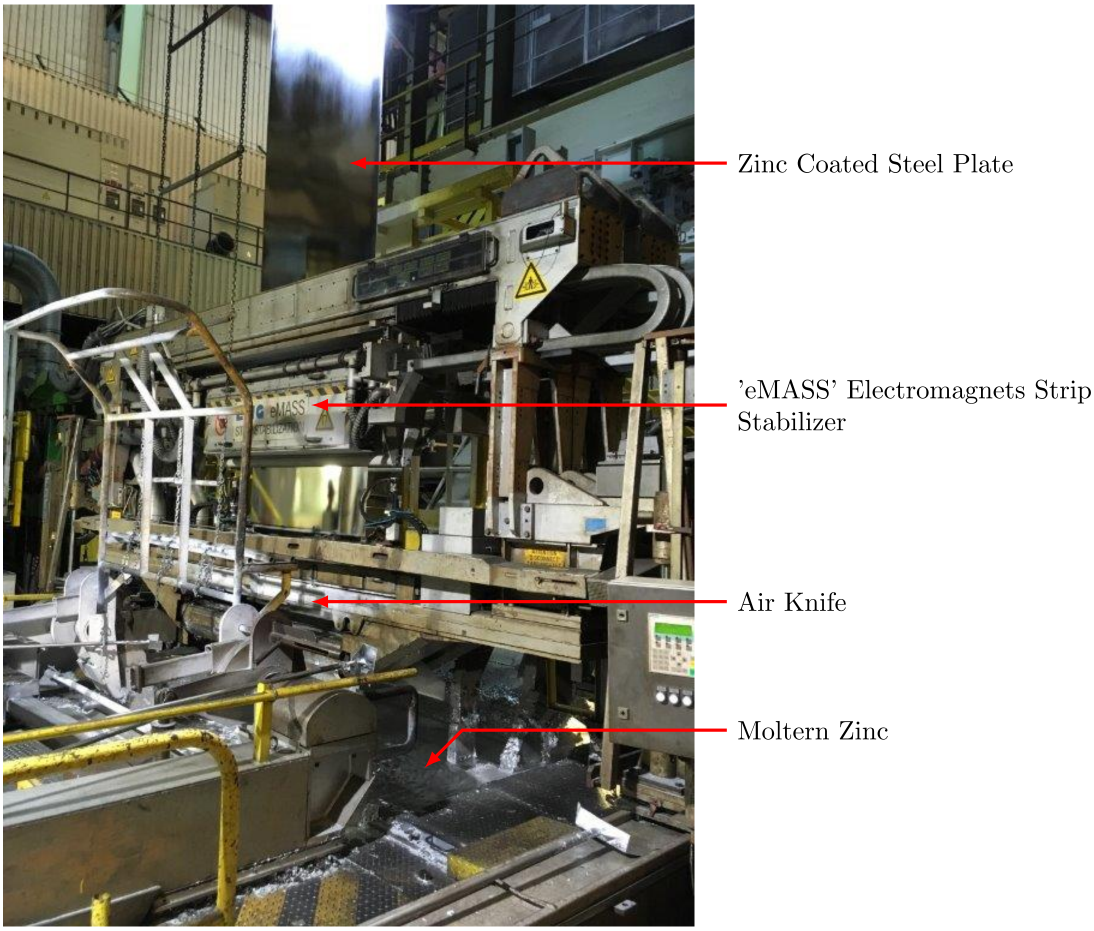



Figure 1.2: Hot-Dip galvanization process

## 1.2 About ArcelorMittal

ArcelorMittal is the global leader in steel production and mining activity. ArcelorMittal is formed in 2006 by merging MittalSteel and Arcelor. The headquarters of ArcelorMittal is in Luxembourg. ArcelorMittal have many research centers around the world and they spend hundreds of millions of dollars in research and development. ArcelorMittal Maizières research SA is the research center where this thesis is undertaken under the department of measurement and control. The main task of this department was to explore and fine turn the new measurement techniques in profit of increasing the quality of the steel production. The control team of the department is specialized in developing advanced control strategies (Model Predictive control, Model - based control etc,.) to continuously improve the comfort of operators and the product quality.

## 1.3 Overview

The metal strip in the hot-dip galvanization process exhibit complex behaviors. The metal strip is constantly traveling between rollers and it is also axially loaded to reduce vibration. It experiences lateral load caused by electromagnets and air knife. It is also imposed with displacement load in both the ends. The problem of moving materials exists since the age of industrial revolution in industries like textile webs production, Paper manufacturing and printing press. Many other examples can be found such as conveyor belts, Magnetic tapes, band saw, power transmission belts etc (Ji-Yun, Keum-Shik, and Chang-Do 2002,Katica 2007).

Later work on axially travelling string is provided by Steinboeck et al. 2015 and Koivurova and Salonen 1999. L. Chen 2005 did a detailed bibliographic study of vibration and control of moving string. Travelling string models are perfect for materials like threads which do not have significant bending stiffness. But, for applications like band saw the bending stiffness is significant. So, traveliing beams had to be developed.L.-Q. Chen and Yang 2006, Chang et al. 2010, Li-Qun 2010, Ghayesh and Amabili 2013 provided the models of travelling beams. Marynowski 2008 in the book detailed information moving materials (strings, beams and strips) are discussed.

Hannu Koivurova 1998 discussed moving membrane and provided a comprehensive derivation of mixed formulation. The paper also address the initial curvature and curved roller contact at the boundaries and how to address them. Fluid interaction is also studied and added mass method is provided to model the fluid around the web. Saxinger et al. 2016 studied the axially moving strip using Kirchhoff plate theory and a detailed study on geometric non-linearity is provided . The plates are kept at high temperatures, so the changes of plastic deformation are high. The history of plastic deformation by the rollers affects the zinc coating because of the cross bow effect (Baumgart et al. 2017). The visco elastic band is studies by Saksa et al. 2003. In this case the axial tension is considered as a constant and known. The time varying axial tension cause parametric resonance which is studied by Kim, Perkins, and Lee 2003.

Finite Element method is the most common numerical method used to simulate solids and structures. Even thought the FEM techniques were developed decades back they have been research extensively, will continued to be developed because of their usefulness. A traveling plate model is used for the for simulation. Finite element method to model moving membrane is discussed in H. Koivurova and Pramila 1997. A commercial software is modified to achieve this. Wang 1999 discussed the Finite element modelling of moving Reissner Mindlin plate using MITC4 element. It is also re-emphasized that the natural frequency tends to be zero as the axial velocity approaches the critical speed. The critical speed increases if the axial tension is also increased.

In the current work, both the plate are implepmented using its respective plate theories. Reissner Mindlin plate is approximated by a QUAD4 element [Ferreira 2009] which is the most easiest to apply. PAT element O. Zienkiewicz and Taylor 2005 is used to approximate the Kirchhoff plate. PAT element is $C^1$ continuous which makes it complicated. Newmark algorithm is selected as the default time integration algorithm. The objective of the thesis is to integrate the FEM with control laws. Easiest way is to represnt the equation in state-space form Young and Hyochoong 1997, so that it can be easily used by MATLAB simulink tool. Another technique is by created operator between output and input so that a relation can be applied. this is achieved by operator overloading function in object oriented programming

## 1.4 Thesis Outline

Theories of plates were discussed in chapter.2. Kirchhoff plate and Reissner Mindlin plate were discussed. Hamilton principle is used to derive the weak from of the problem. A form of mixed Euler - Lagrange formulation is used to derive the velocity of the plate and finally the weak form is discussed. In chapter.3, the finite element method is discussed. The elements types QUAD4 and PAT elements are introduced and the conversion of weak from to FE format is provided. Methods to integrate FE with existing control algorithms were given. In results and discussion chapter, the FEM modal is analyses with different loading and boundary conditions. The effects of overall mesh density and directional mesh density were analyzed. An optimized mesh is provided, that is very effective for this application. and finally a comparison is made between the FEM model and existing Galerkin model. Thesis ended with a conclusion. Future needed developments are also discussed in the conclusion. Appendix contains solution plot related to this report.

# Chapter 2

# Axially Moving Plate formulation

A plate is a structure, traditionally flat with the thickness much smaller than other dimensions. Trusses and Beams are the first to be developed as structures. For trusses and beams, the cross-sectional area is much smaller compared to the length, which makes them one-dimensional structure. For the complex geometries like domes and roofs, beam are not sufficient. So, the plates are developed. Plate and shells theories were around for many decades. Even still, they are currently researched and new elements are developed. Because the plates are computational less expensive than solids and more complex than beams, they are very attractive in many applications. In here, main goal is to develop the weak form of an axially moving plate. For the development of axially moving plates, Extended Hamilton principle is used to derive the equation of motion.

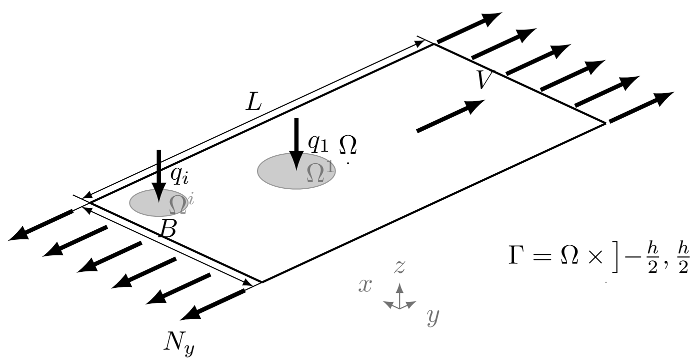



Figure 2.1: Description of domain

A rectangular flat domain of plate with axial tension $N_y$ is shown in the figure.2.1. $\Omega$ is the two-dimensional domain strictly in the x-y plane. The plate has a thickness of $h$, Length $L$ and width $B$. The plate is travelling in the y-direction at velocity $V$. $\Omega^1 \cdots \Omega^i$ are the regions in $\Omega$ where transversal distributed loads $q_1 \cdots q_i$ are applied. This is a thin plate which means $h << L, B$.

## 2.1 Hamilton principle

The equation of motion of the plate is derived using the Hamilton principle. In this case, the material is moving. So a modified form of the Hamilton principle is taken from McIver 1973.

$$\delta H = \int_{t_0}^{t_1} \left(\delta U - \delta K + \delta W + \delta M\right) dt = 0 \tag{2.1}$$

$$\delta \mathbf{u}\Big|_{t_0}^{t_1} = 0 \tag{2.2}$$

$\delta$ is the variation, $t_0$ and $t_1$ are any arbitrary temporal points, $U$ is the total potential energy , $K$ is the kinematic energy and $W$ is the work performed by external forces on the system. $\mathbf{u}$ is the total displacement of the plate. $M$ is the momentum transports at boundaries y=0 and y = L (Saxinger et al. 2016).

$$\delta M = \int_0^W \int_{-h/2}^{h/2} L\rho \mathbf{v} \delta \mathbf{u}\Big|_{y=0}^{y=L} dz dx = 0 \tag{2.3}$$

Here, $\mathbf{v}$ is the total velocity vector of the plate. $M$ becomes zero because the line speed is equal at the boundaries. So, there is no overall change in the mass of the plate.

## 2.2 Plate Theory

Plate theories are formulated by considering some clever assumptions. These assumptions help in simplifying the plates from solids. One of the main assumptions is that the plate thickness does not change after deformation. From this assumption, it is clear that the axial strain $\epsilon_{zz}$ in $z$ direction is not considered. $\sigma_{zz}$ is also neglected because it is very small. But loads in $z$ direction are not neglected. These loads won't contribute to $\sigma_{zz}$ instead they cause bending of the plate. Some plate theories neglect any strain corresponding to $z$ direction. These differences in assumptions cause variations in their properties, which will be discussed in the coming sections. Another assumption is that during the absence of axial deformation, any point in the mid-plane only moves either in an upward or downward direction (figure.2.2). The middle plane is initially flat and its equal distance from the upper and lower faces. Another important assumption is that the flat plane normal to mid-plane will always be a flat plane, they won't distort.

Using these assumptions, the displacement vector of the plate is given as 2.4.

$$\begin{aligned}
u_1 (x, y, z) &= u (x, y) - z\theta_x (x, y) \\
u_2 (x, y, z) &= v (x, y) - z\theta_y (x, y) \\
u_3 (x, y, z) &= w (x, y)
\end{aligned} \tag{2.4}$$

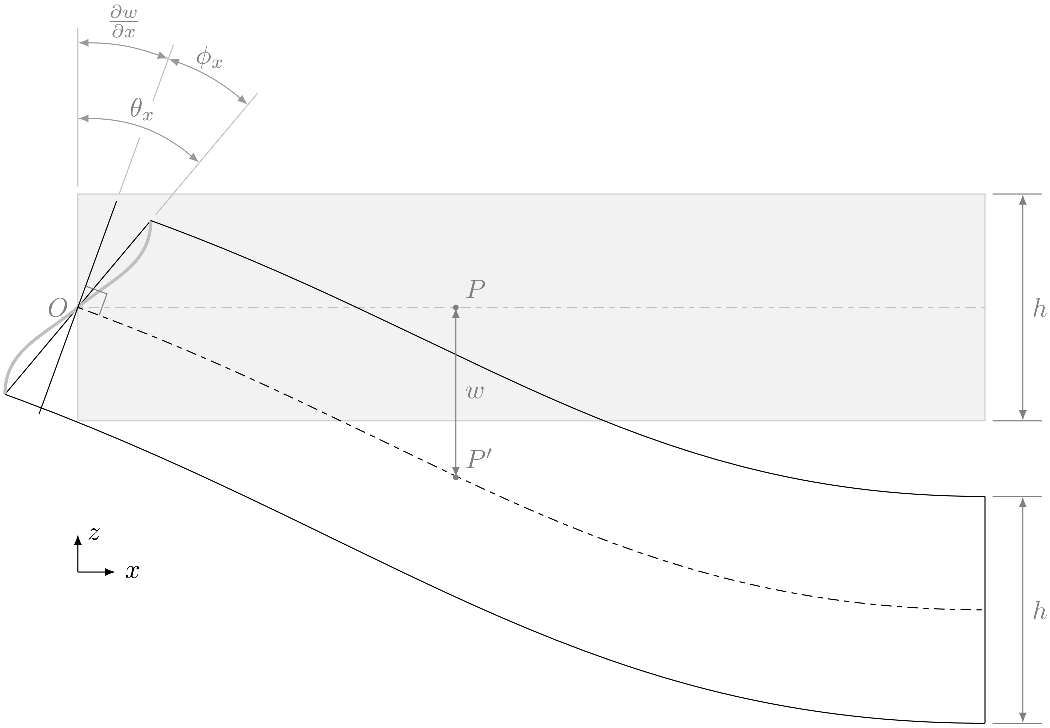



Figure 2.2: A plate under bending deformation

u(x) and v(y) axial displacements are neglected. This is because of the reason that the rollers in the top and bottom positions stays in one position. The axial stress is caused by the differences in material feed in the top and bottom rollers. This axial stress only affects the transverse deformation, not the axial deformation. It is should be noted that the axial stress is applied to increase the bending natural frequency of the plate. From the equation.2.5, It is can be noted that all the components of the displacement vectors are the functions of $w$ direction displacement only.

$$u_1 = -z\theta_x\,(x,y) \qquad u_2 = -z\theta_y\,(x,y) \qquad u_3 = w\,(x,y) \tag{2.5}$$

### 2.2.1 Kirchhoff plate theory

Kirchhoff plate theory is well suitable for thin plates. A straight line normal to mid-plane stays normal and straight after deformation($\phi_x = 0$)see figure:2.2. Because of this assumption, the shear strains ($\epsilon_{23}$ and $\epsilon_{13}$) are neglected.

$$\theta_x = \frac{\partial w}{\partial x} \qquad \theta_y = \frac{\partial w}{\partial y} \tag{2.6}$$

### 2.2.2 Reissner - Mindlin plate

The Reissner Mindlin plate theory is developed for the thick plates but can be used for thin plates with caution. As the thickness tends to zero this plate theory diverges from the thin plate(see. Blaauwendraad 2010). For Reissner Mindlin plate theory, the line normal to the middle plate will not necessarily be normal after deformation, but will be straight $\phi \neq 0$ (see figure: 2.2). In equation.2.7, $\phi_x$ and $\phi_y$ are the angles between plane normal to middle plane and plane of actual deformation.

$$\theta_x = \frac{\partial w}{\partial x} + \phi_x \qquad \theta_y = \frac{\partial w}{\partial y} + \phi_y \tag{2.7}$$

## 2.3 Potential Energy

The total potential strain energy $U$ is given as.

$$U = \frac{1}{2} \int\int\int_{\Gamma} (\boldsymbol{\epsilon})^T \boldsymbol{\sigma} d\Gamma \tag{2.8}$$

$\boldsymbol{\epsilon}$ is the strain tensor, $\boldsymbol{\sigma}$ is the stress tensor and $(\cdot)^T$ denotes transpose of a matrix. The total strain energy is divided into three different terms.

$$U = \frac{1}{2} \int\int\int_{\Gamma} \left(\boldsymbol{\epsilon}^B\right)^T \boldsymbol{\sigma}^B + \left(\boldsymbol{\epsilon}^S\right)^T \boldsymbol{\sigma}^S + \left(\boldsymbol{\epsilon}^A\right)^T \boldsymbol{\sigma}^A d\Gamma \tag{2.9}$$

B,S,A on the superscript indicates bending, shear and axial components of the strain and stress. Of-course the shear strain is zero for Kirchhoff plate. To find the terms in the relation strain displacement and stress-strain relations need to be established. The gravitational potential energy is not considered.

### 2.3.1 Strain - Displacement Relation

The strain tensor is given as

$$\boldsymbol{\epsilon} = \frac{1}{2}(\boldsymbol{\nabla u} + (\boldsymbol{\nabla u})^T) \tag{2.10}$$

expanding equation.2.10 gives equation.2.11.

$$\boldsymbol{\epsilon} = \begin{bmatrix} -z\dfrac{\partial w^2}{\partial x^2} & -z\dfrac{\partial w^2}{\partial x \partial y} & \dfrac{1}{2}\left(\dfrac{\partial w}{\partial x} - \theta_x\right) \\ & -z\dfrac{\partial w^2}{\partial y^2} & \dfrac{1}{2}\left(\dfrac{\partial w}{\partial y} - \theta_y\right) \\ symm. & & 0 \end{bmatrix} \tag{2.11}$$

The strain tensor is separated into to respective terms. The bending strain is given in equation.2.12.

$$\boldsymbol{\epsilon}^B = -z \begin{bmatrix} \dfrac{\partial w^2}{\partial x^2} \\ \dfrac{\partial w^2}{\partial y^2} \\ \dfrac{\partial w^2}{\partial x \partial y} \end{bmatrix} = z\boldsymbol{\kappa} \tag{2.12}$$

$\boldsymbol{\kappa}$ is the curvature of a plate. The shear strain is separated as.

$$\boldsymbol{\epsilon}^S = \frac{1}{2} \begin{bmatrix} \dfrac{\partial w}{\partial x} - \theta_x \\ \dfrac{\partial w}{\partial y} - \theta_y \end{bmatrix} \tag{2.13}$$

For the Kirchhoff plate this term vanishes, which makes sense as the shear strain is not indented to be included. But for the Reissner - Mindlin plate this term does not vanish and gives equation.2.14,

$$\boldsymbol{\epsilon}^S = \frac{1}{2} \begin{bmatrix} -\phi_x \\ -\phi_y \end{bmatrix} \tag{2.14}$$

Only the axial strain in the y-direction is considered equation.2.15(Li et al. 2012). This term is a non-linear strain term. equation.2.15 helps us provide the direct relation between axial stress and lateral deformation $w$. Because of a special assumption, this term will not be non-linear in the final weak form. The assumption is discussed in the following section.

$$\boldsymbol{\epsilon}^A = \left(\frac{\partial w}{\partial y}\right)^2 = (w_{,2})^2 \tag{2.15}$$

### 2.3.2 Constitute law

The Hooke's law for the linear isotropic material is considered. The material is considered to be homogeneously distributed. The stress-strain relation for shear and bending is given separately (L.Gould 1988).

$$\boldsymbol{\sigma}^B = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} = \frac{1}{1-\nu^2} \begin{bmatrix} E & \nu E & 0 \\ \nu E & E & 0 \\ 0 & 0 & (1-\nu^2)G \end{bmatrix} \begin{bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \epsilon_{12} \end{bmatrix} = \boldsymbol{D}\boldsymbol{\epsilon}^B \tag{2.16}$$

$E$ is Young's modulus, $\nu$ is the Poisson's ratio and $G$ is the shear modulus which is given by $G = E/2(1+\nu)$. The shear stress and strain relation is given as

$$\boldsymbol{\sigma}^S = \begin{bmatrix} \sigma_{31} \\ \sigma_{32} \end{bmatrix} = KG \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \epsilon_{31} \\ \epsilon_{32} \end{bmatrix} = \boldsymbol{D_c}\boldsymbol{\epsilon}^S \tag{2.17}$$

$K$ is the shear correction factor (Huang 1989). Shear correction factor value of 5/6 is used for this application (Wang 1999).

From the previous section, it is known that that axial strain is nonlinear. This will the make the weak form nonlinear. Thus requiring a requires special treatment. To overcome this issue, the axial stress($\boldsymbol{\sigma}^A$) is let us a known term. The axial stress is constant and homogeneous in the plate. this make the weak form linear.

$$\boldsymbol{\sigma}^A = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{bmatrix} = \begin{bmatrix} N_x \\ N_y \\ N_{xy} \end{bmatrix} \tag{2.18}$$

For this problem, only $N_y$ is non zero.

### 2.3.3 Variation of the strain energy

First, the integration over the thickness is taken

$$U = \frac{1}{2} \int \int_{\Omega} \int_{-h/2}^{+h/2} \left(\boldsymbol{\epsilon}^B\right)^T \boldsymbol{\sigma}^B + \left(\boldsymbol{\epsilon}^S\right)^T \boldsymbol{\sigma}^S + \left(\boldsymbol{\epsilon}^A\right)^T \boldsymbol{\sigma}^A dz d\Omega \tag{2.19}$$

Thickness is constant all over the plate and is continuous.

$$U = \frac{1}{2} \int \int_{\Omega} \left[ \int_{-h/2}^{+h/2} z^2 dz \right] \boldsymbol{\kappa}^T \boldsymbol{D} \boldsymbol{\kappa} + \left[ \int_{-h/2}^{+h/2} dz \right] \left(\boldsymbol{\epsilon}^S\right)^T \boldsymbol{D_c} \boldsymbol{\epsilon}^S + \left[ \int_{-h/2}^{+h/2} dz \right] \left(\boldsymbol{\epsilon}^A\right)^T \boldsymbol{\sigma}^A d\Omega \tag{2.20}$$

$$U = \frac{1}{2} \int \int_{\Omega} \boldsymbol{\kappa}^T \tilde{D} \boldsymbol{\kappa} + \left(\boldsymbol{\epsilon}^S\right)^T \tilde{\boldsymbol{D}}_{\boldsymbol{c}} \boldsymbol{\epsilon} S + \left(\boldsymbol{\epsilon}^A\right)^T \tilde{\boldsymbol{\sigma}}^{\boldsymbol{A}} d\Omega \tag{2.21}$$

where, $\tilde{\boldsymbol{D}} = h^3\boldsymbol{D}$ similarly $\tilde{\boldsymbol{D}}_{\boldsymbol{c}} = h\boldsymbol{D}_{\boldsymbol{c}}$ and $\tilde{\boldsymbol{\sigma}^A} = h\boldsymbol{\sigma}^A$. The variation of the potential term gives.

$$\delta U = \int\int_\Omega \boldsymbol{\kappa}^T \tilde{\boldsymbol{D}} \delta\boldsymbol{\kappa} + \left(\boldsymbol{\epsilon}^S\right)^T \tilde{\boldsymbol{D}}_{\boldsymbol{c}} \delta\boldsymbol{\epsilon}^S + w_{,2}\tilde{\boldsymbol{\sigma}^A}\delta w_{,2} d\Omega \tag{2.22}$$

## 2.4 Kinetic energy

General Kinetic energy formula is given as

$$K = \frac{1}{2}\int\int\int_\Gamma \mathbf{v}^T\rho\mathbf{v}d\Gamma \tag{2.23}$$

### 2.4.1 Euler - Lagrange formulation

Traditional for solids, Lagrangian description of motion is used, which tracks the material point under deformation and moves along with the point. For fluids, Eulerian Description is used since the fluid is constantly moving. In Eulerian Description, Property changes at a spatial point is recorded. But for the axially moving plate, in addition to deformation, the plate is moving at a constant axial velocity. To describe the velocity a form of mixed Euler-Legrange formulation is selected (Li-Qun 2010).

$$\frac{d(\circ)}{dt} = \frac{\partial(\circ)}{\partial t} + V_i \cdot (\circ)_{,i} \tag{2.24}$$

The plate only move in y direction, which gives.

$$\mathbf{v} = \{\dot{u_1} + V_2 u_{1,2} \quad \dot{u_2} + V_2 u_{2,2} \quad \dot{u_3} + V_2 u_{3,2}\}^T \tag{2.25}$$

### 2.4.2 Variation of the Kinetic energy

Substituting eq: 2.25 in eq: 2.23 gives

$$K = \frac{1}{2}\int\int_\Omega\int_{-\frac{h}{2}}^{\frac{h}{2}} \rho\dot{\boldsymbol{u}}^T\dot{\boldsymbol{u}} + 2\rho V_2\dot{\boldsymbol{u}}^T\boldsymbol{u}_{,2} + \rho V_2^2(\boldsymbol{u}_{,2})^T\boldsymbol{u}_{,2} \quad dzd\Omega \tag{2.26}$$

In here, First term id the acceleration component, second term is the Coriolis and third is the centripetal acceleration components (Li et al. 2012). This makes this equation gyroscopic. Integrating along the thickness gives

$$K = \frac{1}{2} \int \int_{\Omega} \rho \dot{\tilde{\boldsymbol{u}}}^T \boldsymbol{Z} \dot{\tilde{\boldsymbol{u}}} + 2\rho V_2 \dot{\tilde{\boldsymbol{u}}}^T \boldsymbol{Z} \tilde{\boldsymbol{u}}_{,2} + \rho V_2^2 (\tilde{\boldsymbol{u}}_{,2})^T \boldsymbol{Z} \tilde{\boldsymbol{u}}_{,2} \quad d\Omega \tag{2.27}$$

$$\boldsymbol{Z} = \begin{bmatrix} h & 0 & 0 \\ 0 & \frac{h^3}{12} & 0 \\ 0 & 0 & \frac{h^3}{12} \end{bmatrix} \qquad (and) \qquad \tilde{\boldsymbol{u}} = \begin{bmatrix} w \\ \theta_x \\ \theta_y \end{bmatrix} \tag{2.28}$$

Finally the variation of the kinetic energy

$$\delta K = \int \int_{\Omega} \rho \dot{\tilde{\boldsymbol{u}}}^T \boldsymbol{Z} \delta \dot{\tilde{\boldsymbol{u}}} + \rho V_2 \delta \dot{\tilde{\boldsymbol{u}}}^T \boldsymbol{Z} \tilde{\boldsymbol{u}}_{,2} + \rho V_2 \dot{\tilde{\boldsymbol{u}}}^T \boldsymbol{Z} \delta \tilde{\boldsymbol{u}}_{,2} + \rho V_2^2 (\tilde{\boldsymbol{u}}_{,2})^T \boldsymbol{Z} \delta \tilde{\boldsymbol{u}}_{,2} \quad d\Omega \tag{2.29}$$

## 2.5 External Work

The Transverse distributed forces $q_j$ is applied in the regions in $\Omega^j$.$nb$ is the total number of distributed load. The variation of the work by external force is given as

$$\delta W = \sum_{j}^{nb} \int_{\Omega^j} q_j \delta \tilde{\boldsymbol{u}} \quad d\Omega^j \tag{2.30}$$

## 2.6 Final Weak Form

Substituting equations 2.29, 2.3, 2.22 and 2.30 in 2.1 and integration by parts gives a equation,

and by using the relation 2.2, Final weak form is derived as.

$$\begin{aligned} &\int \int_{\Omega} \rho \ddot{\tilde{\boldsymbol{u}}}^T \boldsymbol{Z} \delta \tilde{\boldsymbol{u}} + \rho V_1 \delta \tilde{\boldsymbol{u}} \boldsymbol{Z} \dot{\tilde{\boldsymbol{u}}}_{,2} + \rho V_1 \tilde{\boldsymbol{u}} \boldsymbol{Z} \delta \dot{\tilde{\boldsymbol{u}}}_{,2} - \rho V_1^2 \tilde{\boldsymbol{u}}_{,2} \boldsymbol{Z} \delta \tilde{\boldsymbol{u}}_{,2} \\ &+ \boldsymbol{\kappa}^T \tilde{\boldsymbol{D}} \delta \boldsymbol{\kappa} + \left(\boldsymbol{\epsilon}^S\right)^T \tilde{\boldsymbol{D}}_{\boldsymbol{c}} \delta \boldsymbol{\epsilon}^S + w_{,2} \tilde{\boldsymbol{\sigma}}^A \delta w_{,2} d\Omega = \sum_{j}^{nb} \int_{\Omega^j} q_j \delta \tilde{\boldsymbol{u}} d\Omega^j \end{aligned}$$

# Chapter 3

# Finite Element formulation

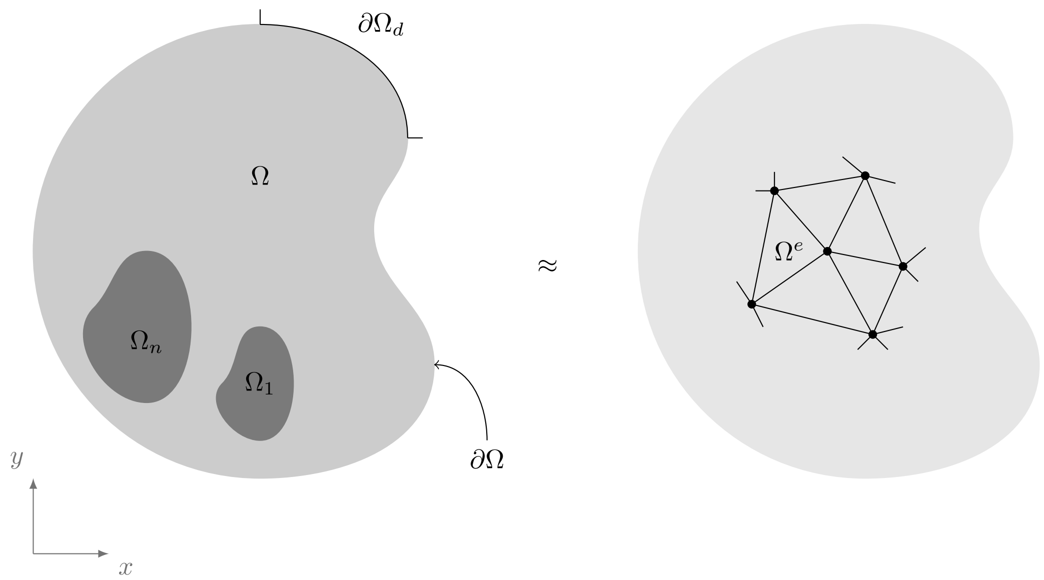



Figure 3.1: FEM domain

In finite element the approximate function is the shape function also called as interpolation function. In here, the continuous domain is discretized into small elements. These elements are either shaped as triangles or quadrangles depending upon the need and choice.

The description of the domain is given in the the figure.3.1. $\Omega$ is the total two dimensional domain in the x , y plane. The domain is discretized into small elements $\Omega^e$. $\Omega^1 \cdots \Omega^n$ are the regions where transverse distributed loads $(q_1 \cdots q_i)$ are described. $\partial\Omega$ is the boundary of the domain. $\partial\Omega_d$ is part of the boundary where Dirichlet boundary condition is applied.

## 3.1 Shape-Functions

The Displacement field of the each element is the function of displacement of degree of freedom of each node, which lets us have a finite number of unknowns to denote the over all displacement field of the domain.

Since it is a plate element, three independent degrees of freedom are described for each node.

$$\tilde{\mathbf{u}} = [w, \theta_x, \theta_y]^T$$

$w$ represents the transverse displacement. $\theta_x$ and $\theta_y$ represents the rotations.

$$\theta_x = \frac{\partial w}{\partial x} \qquad \theta_y = \frac{\partial w}{\partial y} \tag{3.1}$$

The approximate displacement of the element is the given as sum of product of nodal degree of freedom and its corresponding shape functions ($N$,$\overline{N}$,$\overline{\overline{N}}$).

$$\tilde{\mathbf{u}} \approx \sum_{i=1}^{n} \left( N_i w_i + \overline{N}_i \theta_{x_i} + \overline{\overline{N}}_i \theta_{y_i} \right) \tag{3.2}$$

### 3.1.1 Reissner Mindlin Plate Element (Thick/thin plate)

From the previous chapter, it is understood that the Ressigner Mindlin plate is specifically developed for thick plate but the plate also performs reasonably well it thin plate situation. In this element all shear strain is considered, leaving only the out of plane normal stress $\sigma_{zz}$ to be neglected. This element only requires $C^0$ continuity of the approximation, which drastically simplifies the implementation of the shape function (Alexander 1994). The rotations $\theta_x$ and $\theta_y$ are treated as independent variable which in-turn lets us chose independent $C^0$ continuous shape functions for each independent degrees of Freedom.

$$w = \sum_{i=1}^{n} N_i w_i \qquad \theta_x = \sum_{i=1}^{n} \overline{N}_i \theta_{x_i} \qquad \theta_y = \sum_{i=1}^{n} \overline{\overline{N}}_i \theta_{y_i} \tag{3.3}$$

$$\overline{\overline{N}}_i = \overline{N}_i = N_i$$

To begin with a simpler case, a simplest of all shape function, a QUAD4 shape function is selected (Ferreira 2009). Despite it simplicity, the element performs relatively well but it suffers from a 'element lock' when it is too thin (Bathe and Dvorkin 1985).

$$\begin{aligned} N_1 &= \frac{1}{4}(1-\xi)(1-\eta) \quad N_2 = \frac{1}{4}(1+\xi)(1-\eta) \\ N_3 &= \frac{1}{4}(1+\xi)(1+\eta) \quad N_4 = \frac{1}{4}(1-\xi)(1+\eta) \end{aligned} \tag{3.4}$$

The parent element shape function is given in equation.3.4, which is represents the iso-parametric rectangle element. To describe any arbitrary physical quadrangle element, Jacobian transformation is adapted. Using chain rule.

$$\frac{\partial N}{\partial \xi} = \frac{\partial N}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial N}{\partial y}\frac{\partial y}{\partial \xi} \tag{3.5}$$

This relation is written in matrix form to get the Jacobin matrix,

$$\left\{ \begin{array}{c} \dfrac{\partial N}{\partial \xi} \\ \dfrac{\partial N}{\partial \eta} \end{array} \right\} = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial y}{\partial \xi} \\ \dfrac{\partial x}{\partial \eta} & \dfrac{\partial y}{\partial \eta} \end{bmatrix} \left\{ \begin{array}{c} \dfrac{\partial N}{\partial x} \\ \dfrac{\partial N}{\partial y} \end{array} \right\} \quad J = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial y}{\partial \xi} \\ \dfrac{\partial x}{\partial \eta} & \dfrac{\partial y}{\partial \eta} \end{bmatrix} \tag{3.6}$$

The inverse relation is provided by inverse of the Jacobin matrix. the graphical representation of the Jacobin representation is given is given in the figure.3.2 .

$$\left\{ \begin{array}{c} \dfrac{\partial N}{\partial x} \\ \dfrac{\partial N}{\partial y} \end{array} \right\} = J^{-1} \left\{ \begin{array}{c} \dfrac{\partial N}{\partial \xi} \\ \dfrac{\partial N}{\partial \eta} \end{array} \right\} \tag{3.7}$$

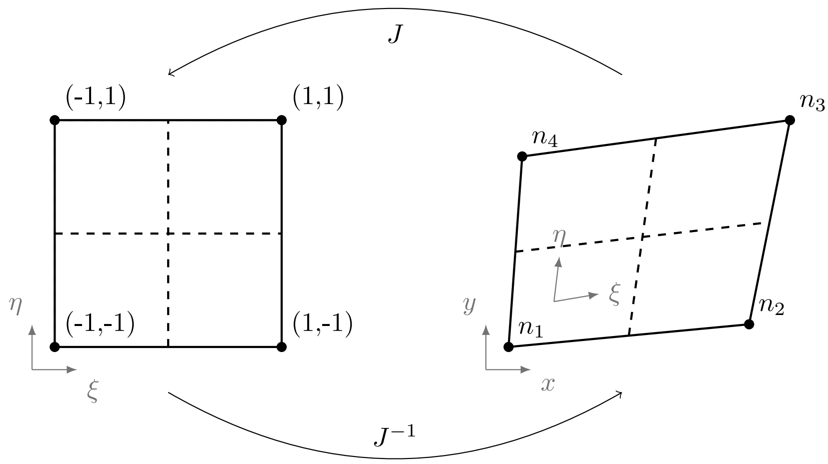



Figure 3.2: Jacobian transformation

The equation.3.2 is represented in matrix form as

$$\tilde{\mathbf{u}} = \left\{ \begin{array}{c} w \\ \theta_x \\ \theta_y \end{array} \right\} \approx \sum_{i=1}^{nN} \left\{ \begin{array}{c} N_i w_i \\ \overline{N}_i \theta_{x_i} \\ \overline{\overline{N}}_i \theta_{y_i} \end{array} \right\} = \begin{bmatrix} N_1 & 0 & 0 & \cdots & N_4 & 0 & 0 \\ 0 & \overline{N}_1 & 0 & \cdots & 0 & \overline{N}_4 & 0 \\ 0 & 0 & \overline{\overline{N}}_1 & \cdots & 0 & 0 & \overline{\overline{N}}_4 \end{bmatrix} \left\{ \begin{array}{c} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ \vdots \\ w_4 \\ \theta_{x_4} \\ \theta_{y_4} \end{array} \right\} = \mathbf{N}\tilde{\mathbf{u}}^{\mathbf{e}}$$

Now, this FEM matrix from is easier to work with. This will be substituted into the weak form of the equation to get the over an ODE in terms of FE matrices. The shape function is independent of time so derivative of $u$ with respect to time will not affect shape function.

$$\dot{\tilde{\mathbf{u}}} = \mathbf{N}\dot{\tilde{\mathbf{u}}}^{\mathbf{e}} \qquad \ddot{\tilde{\mathbf{u}}} = \mathbf{N}\ddot{\tilde{\mathbf{u}}}^{\mathbf{e}} \tag{3.8}$$

The curvature term in the weak form is given in the equation below in equation.3.9. it can be noticed that the double derivative of the displacement shape function this matrix alone will produce singular stiffness matrix but along with shear strain term (equation.3.10)this will not have zero row or column.

$$\boldsymbol{\kappa} \approx \begin{bmatrix} 0 & \overline{N}_{1,1} & 0 & \cdots & 0 \\ 0 & 0 & \overline{\overline{N}}_{1,2} & \cdots & \overline{\overline{N}}_{4,2} \\ 0 & \overline{N}_{1,2} & \overline{\overline{N}}_{1,1} & \cdots & \overline{\overline{N}}_{4,1} \end{bmatrix} \begin{Bmatrix} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ \vdots \\ \theta_{y_4} \end{Bmatrix} = \mathbf{B}\tilde{\mathbf{u}}^{\mathbf{e}} \tag{3.9}$$

$$\tilde{\boldsymbol{\epsilon}}^{\boldsymbol{S}} \approx \begin{bmatrix} N_{1,1} & \overline{N}_1 & 0 & \cdots & 0 \\ N_{1,2} & 0 & \overline{\overline{N}}_1 & \cdots & \overline{\overline{N}}_4 \end{bmatrix} \begin{Bmatrix} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ \vdots \\ \theta_{y_4} \end{Bmatrix} = \mathbf{B_S}\tilde{\mathbf{u}}^{\mathbf{e}} \tag{3.10}$$

Terms to corresponding the axial strain is given in eq : 3.11

$$\tilde{w}_{,2} \approx \begin{bmatrix} N_{1,2} & 0 & 0 & N_{2,2} & \cdots & 0 \end{bmatrix} \begin{Bmatrix} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ w_2 \\ \vdots \\ \theta_{y_{nN}} \end{Bmatrix} = \mathbf{H_A}\tilde{\mathbf{u}}^{e} \tag{3.11}$$

Similarly,

$$\tilde{\boldsymbol{u}}_{,2} \approx \begin{bmatrix} N_{1,2} & 0 & 0 & \cdots & 0 \\ 0 & \overline{N}_{1,2} & 0 & \cdots & 0 \\ 0 & 0 & \overline{\overline{N}}_{1,2} & \cdots & \overline{\overline{N}}_{4,2} \end{bmatrix} \begin{Bmatrix} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ \vdots \\ \theta_{y_{nN}} \end{Bmatrix} = \mathbf{H_v}\tilde{\mathbf{u}}^{e} \tag{3.12}$$

The FE Matrix for the body force is given as

$$\tilde{w} \approx \begin{bmatrix} N_1 & 0 & 0 & N_2 & \cdots & 0 \end{bmatrix} \begin{Bmatrix} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ w_2 \\ \vdots \\ \theta_{y_{nN}} \end{Bmatrix} = \mathbf{N_f}\tilde{\mathbf{u}}^{e} \tag{3.13}$$

### 3.1.2 Kirchhoff plate element ( Thin Plate )

This plate element is more complicated to implement as the shape function required for it need to be $C^1$ continuous.

For Kirchhoff plates, quite a few elemets which passes patch test exists. Smith and Duncan 1970 and O. C. Zienkiewicz and Cheung 1964 proposed easy method to generate stiffness matrix for rectangular and parallelogram elements. For generating any arbitrary quadrangle element the existing rectangle elements can be used by transformation, but they perform badly (O. Zienkiewicz and Taylor 2005). To overcome this issue, two or more triangle elements are joined to create a quadrangle elements. Triangle element uses different coordinate system called area coordinate system (3.3) which has three axis.

#### Area Coordinates

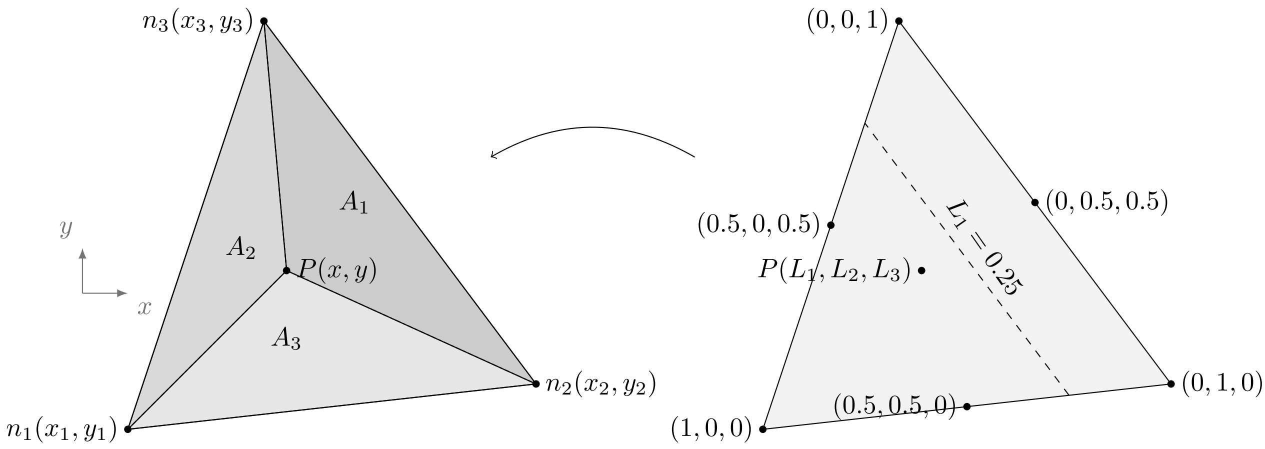



Figure 3.3: Area coordinate

Area coordinate is a parametric coordinates three coordinates($L_1$,$L_2$,$L_3$), which defined as

$$L_1 = \frac{A_1}{A} \quad L_2 = \frac{A_2}{A} \quad L_3 = \frac{A_3}{A} \tag{3.14}$$

for which , it must satisfy

$$L_1 + L_2 + L_3 = 1 \quad (or) \quad A_1 + A_2 + A_3 = A \tag{3.15}$$

There is a linear relation between this coordinate and Cartesian coordinate which is given as

$$\begin{Bmatrix} 1 \\ x \\ y \end{Bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} \begin{Bmatrix} L_1 \\ L_2 \\ L_3 \end{Bmatrix} \tag{3.16}$$

$$2A = \begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix} \tag{3.17}$$

The linear transform between the derivative of coordinates area coordinate and Cartesian coordinate.

$$\left\{ \begin{matrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{matrix} \right\} = \frac{1}{4A} \begin{bmatrix} y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_2 - x_3 & x_3 - x_1 & x_1 - x_2 \end{bmatrix} \left\{ \begin{matrix} \frac{\partial}{\partial L_1} \\ \frac{\partial}{\partial L_2} \\ \frac{\partial}{\partial L_3} \end{matrix} \right\} \tag{3.18}$$

Similarly, the relation for second derivative is given as.

$$\begin{bmatrix} \frac{\partial^2}{\partial x^2} & \frac{\partial^2}{\partial x \partial y} \\ \frac{\partial^2}{\partial x \partial y} & \frac{\partial^2}{\partial y^2} \end{bmatrix} = \frac{1}{16A^2} \begin{bmatrix} y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_2 - x_3 & x_3 - x_1 & x_1 - x_2 \end{bmatrix} \begin{bmatrix} \frac{\partial^2}{\partial L_1^2} & \frac{\partial^2}{\partial L_1 \partial L_2} & \frac{\partial^2}{\partial L_1 \partial L_3} \\ \frac{\partial^2}{\partial L_1 \partial L_2} & \frac{\partial^2}{\partial L_2^2} & \frac{\partial^2}{\partial L_2 \partial L_3} \\ \frac{\partial^2}{\partial L_1 \partial L_3} & \frac{\partial^2}{\partial L_3 \partial L_2} & \frac{\partial^2}{\partial L_3^2} \end{bmatrix} \begin{bmatrix} y_2 - y_3 & x_2 - x_3 \\ y_3 - y_1 & x_3 - x_1 \\ y_1 - y_2 & x_1 - x_2 \end{bmatrix} \tag{3.19}$$

### PAT element shape function

The triangle element with three nodes, is given here. This element passes patch test and it is referred from O. Zienkiewicz and Taylor 2005. In literature this element is referred as PAT. The element was developed by Specht 1988. The element is based on a polynomial expression of nine terms.

$$\begin{aligned} \mathbf{P} = [&L_1 \quad L_2 \quad L_3 \quad L_1L_2 \quad L_2L_3 \quad L_3L_1 \\ &L_1^2L_2 + \tfrac{1}{2}L_1L_2L_3(3(1-\mu_3)L_1 - (1+3\mu_3)L_2 + (1+3\mu_3)L_3)) \\ &L_2^2L_3 + \tfrac{1}{2}L_1L_2L_3(3(1-\mu_1)L_2 - (1+3\mu_1)L_3 + (1+3\mu_1)L_1)) \\ &L_3^2L_1 + \tfrac{1}{2}L_1L_2L_3(3(1-\mu_2)L_3 - (1+3\mu_2)L_1 + (1+3\mu_2)L_2))] \end{aligned} \tag{3.20}$$

$$\mu_1 = \frac{l_3^2 - l_2^2}{l_1^2} \qquad \mu_2 = \frac{l_1^2 - l_3^2}{l_2^2} \qquad \mu_3 = \frac{l_2^2 - l_1^2}{l_3^2} \tag{3.21}$$

$l_a$ is the length of the side which is opposite to the edge a. The nine shape functions for this element using the polynomial expression (equation. 3.20).

$$N = \begin{bmatrix} P(1) - P(4) + P(6) + 2*(P(7) - P(9)) \\ -b(2)*(P(9) - P(6)) - b(3)*P(7) \\ -c(2)*(P(9) - P(6)) - c(3)*P(7) \\ P(2) - P(5) + P(4) + 2*(P(8) - P(7)) \\ -b(3)*(P(7) - P(4)) - b(1)*P(8) \\ -c(3)*(P(7) - P(4)) - c(1)*P(8) \\ P(3) - P(6) + P(5) + 2*(P(9) - P(8)) \\ -b(1)*(P(8) - P(5)) - b(2)*P(9) \\ -c(1)*(P(8) - P(5)) - c(2)*P(9) \end{bmatrix} \tag{3.22}$$

The terms in the weak form is expressed in terms of Shape function and given below

$$\tilde{\mathbf{u}} \approx \left\{ \begin{array}{c} w \\ \theta_x \\ \theta_y \end{array} \right\} = \begin{bmatrix} N_1 & N_2 & N_3 & N_4 & \cdots & N_9 \\ N_{1,1} & N_{2,1} & N_{3,1} & N_{4,1} & \cdots & N_{9,1} \\ N_{1,2} & N_{2,2} & N_{3,2} & N_{4,2} & \cdots & N_{9,2} \end{bmatrix} \left\{ \begin{array}{c} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ w_2 \\ \vdots \\ \theta_{y_3} \end{array} \right\} = \mathbf{N}\tilde{\mathbf{u}}^{\mathbf{e}}$$

The other terms is the weak form are also given as

$$\boldsymbol{\kappa} \approx \begin{bmatrix} N_{1,11} & N_{2,11} & N_{3,11} & \cdots & N_{9,11} \\ N_{1,22} & N_{2,22} & N_{3,22} & \cdots & N_{9,22} \\ N_{1,12} & N_{2,12} & N_{3,12} & \cdots & N_{9,12} \end{bmatrix} \left\{ \begin{array}{c} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ \vdots \\ \theta_{y_3} \end{array} \right\} = \mathbf{B}\tilde{\mathbf{u}}^{\mathbf{e}} \tag{3.23}$$

$$\tilde{w}_{,2} \approx \begin{bmatrix} N_{1,2} & N_{2,2} & N_{3,2} & N_{4,2} & \cdots & N_{9,2} \end{bmatrix} \left\{ \begin{array}{c} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ w_2 \\ \vdots \\ \theta_{y_3} \end{array} \right\} = \mathbf{H_A}\tilde{\mathbf{u}}^{e} \tag{3.24}$$

$$\tilde{\boldsymbol{u}}_{,2} \approx \begin{bmatrix} N_{1,2} & N_{2,2} & N_{3,2} & \cdots & N_{9,2} \\ N_{1,22} & N_{2,22} & N_{3,22} & \cdots & N_{9,22} \\ N_{1,12} & N_{2,12} & N_{3,12} & \cdots & N_{9,12} \end{bmatrix} \left\{ \begin{array}{c} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ \vdots \\ \theta_{y_3} \end{array} \right\} = \mathbf{H_v}\tilde{\mathbf{u}}^{e} \tag{3.25}$$

The FE Matrix for the body force is given as

$$\tilde{w} \approx \begin{bmatrix} N_1 & N_2 & N_3 & N_4 & \cdots & N_9 \end{bmatrix} \left\{ \begin{array}{c} w_1 \\ \theta_{x_1} \\ \theta_{y_1} \\ w_2 \\ \vdots \\ \theta_{y_3} \end{array} \right\} = \mathbf{N_f}\tilde{\mathbf{u}}^{e} \tag{3.26}$$

### 3.1.3 Final FE form

Weak Form to FE format The Finite Element Matrix equation is given as

$$
\begin{aligned}
\int\int_{\Omega} \left(\rho\,[\mathbf{N}]\,[\mathbf{Z}]\,[\mathbf{N}]\,\{\ddot{\tilde{\mathbf{u}}}^e\}\right)\delta\tilde{\mathbf{u}}^e + \left(2\rho V_1\,[\mathbf{N}]\,[\mathbf{Z}]\,[\mathbf{H_v}]\,\{\dot{\tilde{\mathbf{u}}}^e\}\right)\delta\tilde{\mathbf{u}}^e \\
-\left(\rho V_1^2\,[\mathbf{H_v}]\,[\mathbf{Z}]\,[\mathbf{H_v}]\,\{\tilde{\mathbf{u}}^e\}\right)\delta\tilde{\mathbf{u}}^e + \left([\mathbf{B}]\left[\tilde{\mathbf{D}}\right][\mathbf{B}]\,\{\tilde{\mathbf{u}}^e\}\right)\delta\tilde{\mathbf{u}}^e \\
+\left([\mathbf{B_S}]\left[\tilde{\mathbf{D}}_\mathbf{S}\right][\mathbf{B_S}]\,\{\tilde{\mathbf{u}}^e\}\right)\delta\tilde{\mathbf{u}}^e + \left([\mathbf{H_A}]\left[\tilde{\mathbf{N}}_\mathbf{A}\right][\mathbf{H_A}]\,\{\tilde{\mathbf{u}}^e\}\right)\delta\tilde{\mathbf{u}}^e d\Omega \\
= \sum_{i}^{nb}\int\int_{\Omega_i}\left(q_i\left[\tilde{\mathbf{N}}_\mathbf{f}\right]\right)\delta\tilde{\mathbf{u}}^e d\Omega_i
\end{aligned}
$$

After rearranging them to their respective groups.

$$[\mathbf{M^e}]\,\{\ddot{\mathbf{u}}\} + [\mathbf{C^e}]\,\{\dot{\mathbf{u}}\} + [\mathbf{K^e}]\,\{\mathbf{u}\} = \{\mathbf{F^e}\}$$

where

$$
\begin{aligned}
&[\mathbf{M^e}] = \rho\int\int_{\Omega}\left([\mathbf{N}]\,[\mathbf{Z}]\,[\mathbf{N}]\right)d\Omega \\
&[\mathbf{C^e}] = 2\rho V_1\int\int_{\Omega}\left([\mathbf{N}]\,[\mathbf{Z}]\,[\mathbf{H_v}]\right)d\Omega \\
&[\mathbf{K^e}] = -\rho V_1^2\int\int_{\Omega}\left([\mathbf{H_v}]\,[\mathbf{Z}]\,[\mathbf{H_v}]\right)d\Omega + \int\int_{\Omega}[\mathbf{B}]\left[\tilde{\mathbf{D}}\right][\mathbf{B}]\,d\Omega \\
&\quad + \int\int_{\Omega}[\mathbf{B_S}]\left[\tilde{\mathbf{D}}_\mathbf{S}\right][\mathbf{B_S}]\,d\Omega + \int\int_{\Omega}[\mathbf{H_A}]\left[\tilde{\mathbf{N}}_\mathbf{A}\right][\mathbf{H_A}]\,d\Omega \\
&\{\mathbf{F^e}\} = \sum_{i}^{nb}\int\int_{\Omega_i} q_i\left[\tilde{\mathbf{N}}_\mathbf{f}\right]d\Omega_i
\end{aligned}
$$

### 3.1.4 Numerical Integration

For the numerical integration of the Finite element matrix, Gauss quadrature is used here. The integration for a element is provided by sum of product of Gauss weight and value of the term at the Gauss point (equation.3.27).

$$\int\int f(x,y)dxdy = \sum_{i=1}^{n_g} w^i \cdot f(x_g^i, y_g^i) \tag{3.27}$$

$n_g$ is the number of Gauss points. $w^i$ are Gauss weights and $(x_g^i, y_g^i)$ is the Gauss coordinate. Gauss weights and points for each element is given in the table . 3.1. Using Gauss quadrature information, new relation for element mass matrix is given in equation . 3.28

$$[\mathbf{M^e}] = \sum_{i=1}^{ng}\rho\left(w^i\,[\mathbf{N(i)}]^T\,[\mathbf{Z}]\,[\mathbf{N(i)}]\,det(J)\right)d\Omega \tag{3.28}$$

| Element | $n_g$ | Gauss Point $(x_g^i, y_g^i)$ | Gauss weight $(w^i)$ |
|---|---|---|---|
| PAT | 3 | {(1/2,1/2,0) (1/2,0,1/2) (0,1/2,1/2)} | {1/3,1/3,1/3} |
| QUAD4 | 4 | {(0.5,0.5) (0.5,-0.5)(-0.5,0.5)(-0.5,-0.5)} | {1,1,1,1} |
| | 1 | {(0,0)} | {4} |

Table 3.1: Gauss points and weights

All the Element mass Matrices $[\mathbf{M^e}]$ are assembled in the final Mass Matrix $[\mathbf{M}]$, similarly for other element matrices gives us the final FE ODE.

$$[\mathbf{M}]\{\ddot{\mathbf{u}}\} + [\mathbf{C}]\{\dot{\mathbf{u}}\} + [\mathbf{K}]\{\mathbf{u}\} = \{\mathbf{F}\} \tag{3.29}$$

## 3.2 Solution Methods

Since Matlab is used to program, all the necessary functions are already available within MATLAB. To solve the static problems u=A/b function is used as it factorizes the matrix and solves them, which is much efficient than direct inversion of the A matrix. The find the natural frequency **'eigs'** function is used, which find the subspace of the eigen function. This function is very useful since we only ever need few eigen frequencies that the small. To solve the dynamic problem, New mark Time integration is used. The reason is that, this technique is unconditionally stable and ability to provide numerical damping.

## 3.3 Integration with Control Algorithm

The main objective of the thesis is to code FEM that will be used as simulator for control algorithms. To do two methods are used.

### 3.3.1 State - Space Format

State - Space form is the widely used format for control study of a dynamic system. State - state form is represented as first order ordinary differential equation (equation.3.30). The dynamic system that is present here is a second order differential equation.3.29. This Second order equation is converted to first order equation for which it will consistent to the state space form. The another reason is that it is easier to represent input and output of the system in a MATLAB Simulink block.

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}(t) \tag{3.30}$$

$x(t)$ is the state variable.

$$\boldsymbol{x} = \left\{ \begin{array}{c} \boldsymbol{u}(t) \\ \dot{\boldsymbol{u}}(t) \end{array} \right\} \tag{3.31}$$

using this the second order ODE is reprented in state - space form as

$$\dot{\boldsymbol{x}} = \frac{d}{dt} \left\{ \begin{array}{c} \boldsymbol{u}(t) \\ \dot{\boldsymbol{u}}(t) \end{array} \right\} = \begin{bmatrix} 0 & \boldsymbol{I} \\ -\left[\boldsymbol{M}\right]^{-1}\left[\boldsymbol{K}\right] & -\left[\boldsymbol{M}\right]^{-1}\left[\boldsymbol{C}\right] \end{bmatrix} \left\{ \begin{array}{c} \boldsymbol{u}(t) \\ \dot{\boldsymbol{u}}(t) \end{array} \right\} + \begin{bmatrix} 0 \\ \left[\boldsymbol{M}\right]^{-1}\left[\boldsymbol{F}\right] \end{bmatrix} \tag{3.32}$$

Unfortunately, the FEM discretization of the domain have huge number of nodes which means the system size will also be huge.Which creates its own problems. When an attempt to made to solve the full FEM model of the system it the Simulink block would not converge. To overcome this problem a the size of the model is reduced by using modal - superposition method. After using Modal - Superposition technique, drastic improvements in the solution time is observed but the loss of accuracy is not studied during the thesis.

### 3.3.2 Object oriented Programming

In the Industry, to effectively and rapidly control the metal Strip, a sensor is placed which measures the vibration of the plate. Then the data is used to calculate the signal to be given to the electromagnets. The formula can be simple to complex, but the underlying concept is that force vector $F(t+1)$ which is to be applied is the function of displacement vectors $U(t) \cdots U(1)$ of previous time steps.

Effectively testing such control laws is also very essential before implementing in the plant. To implement this, altering the code source file each time is not feasible may prone to error. So to provide a easy interface, Object oriented Programming is adapted. Forces, Displacements, Time series, Solver Time, Probes are the Matlab class. Number of Operator overloading functions are defined to operate between the objects of these classes.

# Chapter 4

# Results and Discussion

To effectively validate both the elements given in chapter.3, multiple convergence tests were studied. To study the effect of boundary conditions and loads, a circular plat with point load and distributed load is analysed for both elements. Here, the element is simply supported on one side and fixed on other side. The same simulation is done for increasing mesh density (figure: 4.1b). The simulation results are plotted in figure : 4.1a. Difference between analytical solution (equation :4.1,equation : 4.2)(see.Timoshenko and Krieger 1987) and numerical solution(figure.A.1 and figure.A.2) at the center of the circular plate where the displacement is maximum is considered as error.

Analytically solution of simply supported circular plate with point load.

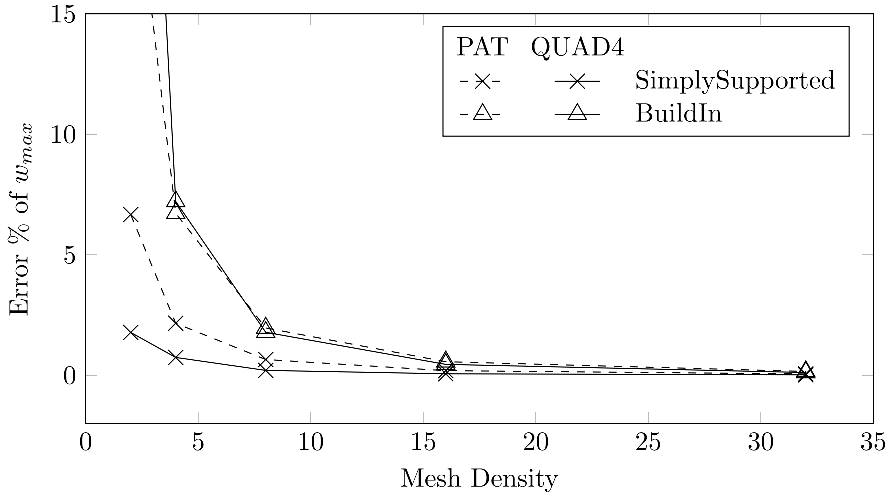
$$w = \frac{P}{16\pi D}\left[\frac{3+\nu}{1+\nu}\left(a^2 - r^2\right) + 2r^2 log\frac{r}{a}\right] \tag{4.1}$$

Analytically solution of build-in circular plate with point load.

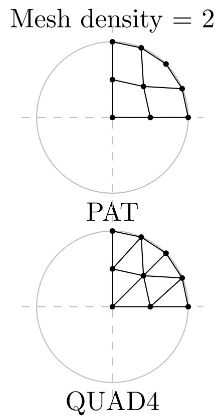
$$w = \frac{Pr^2}{8\pi D} log\frac{r}{a} + \frac{P}{16\pi D}\left(a^2 - r^2\right) \tag{4.2}$$

Where, r is the distance from the center of the circle. From the figure.4.1a, It can be noted that both the elements converges faster for simply supporter boundary condition.For the application of hot dip galvanization, only simply supported boundary condition is necessary. So lack of fast convergence in build-in Boundary condition is not a concern. It can also be noted that the QUAD4 element preforms better than PAT element. Same analysis is also done where load is equally distributed on the surface. the analytically solution of the problem is given by equation.4.3 and equation.4.4 (see. Timoshenko and Krieger 1987).

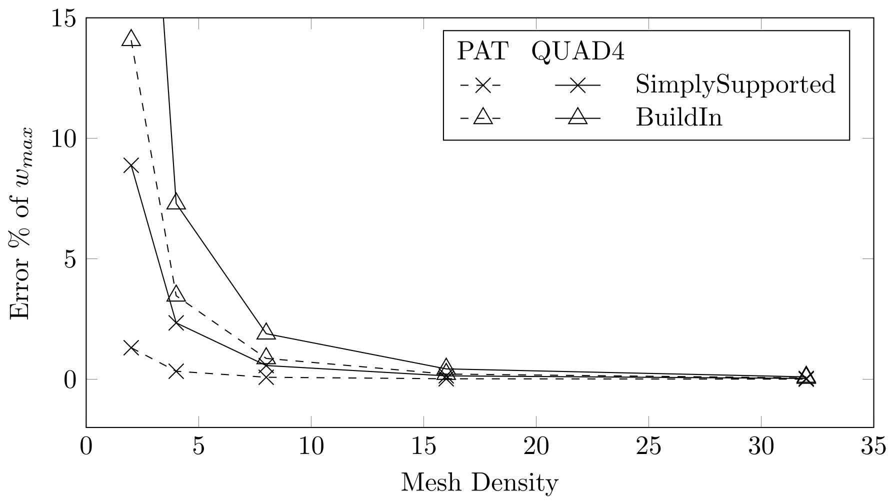
Analytical displacement at the center of simply supported circular plate with Distributed Load.



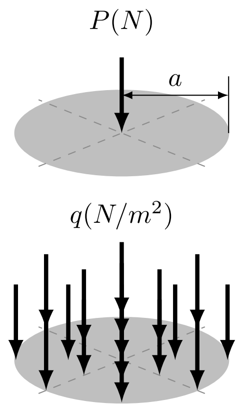

(a) Point load




(b) Mesh




(c) Distributed load




(d) Loads

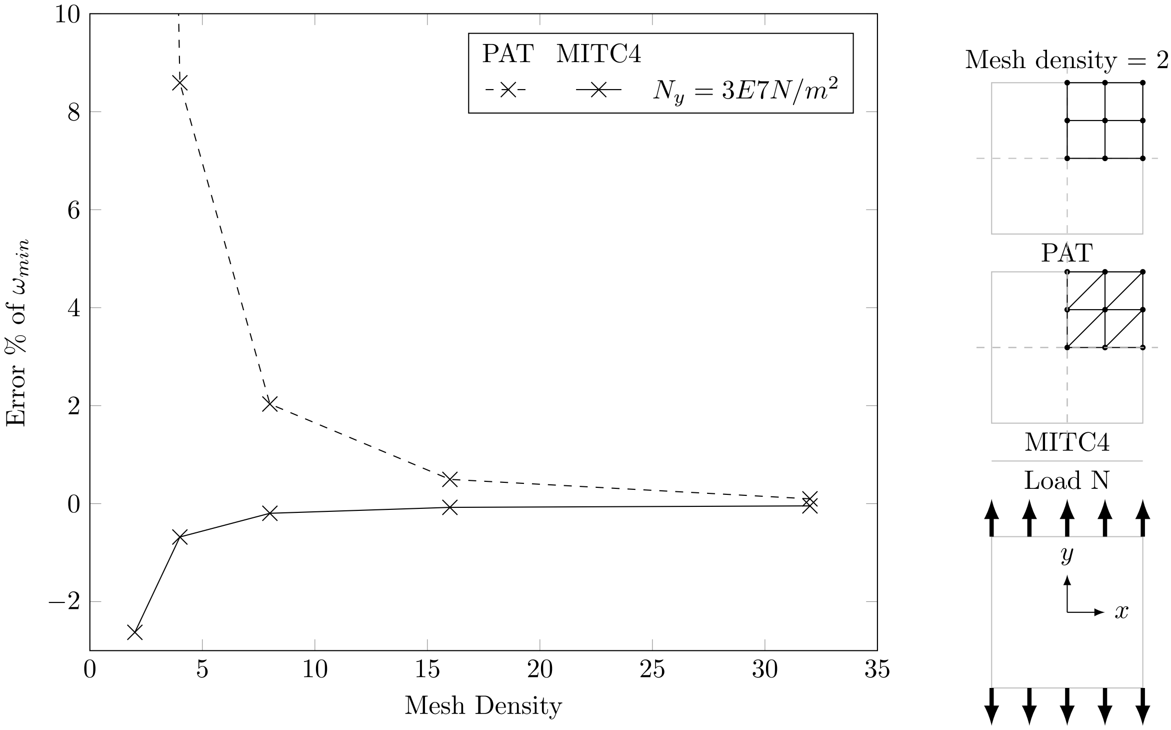
Figure 4.1: Convergence of elements on different loading and boundary conditions.

$$w_{max} = \frac{(5+\nu)\,qa^4}{64D\,(1+\nu)} \tag{4.3}$$

Analytical displacement at the center of Build-in circular plate with Distributed Load.

$$w_{max} = \frac{qa^4}{64D} \tag{4.4}$$

From the figure : 4.1c, the Point to notice is that the PAT element performs better for distributed load. Similar to the previous case. The FEM program doesn't converge fast




(a) Plot of error against mesh density

(b) Mesh

Figure 4.2: Convergence of natural frequency of square plate with axial load

for the build-in load. From this analysis, it can be stated that PAT element with simply supported boundary condition is a better option if distributed load is involved.

The steel strip in hot dip galvaniwation is axially stressed, to understand the convergence of both elements during axial load, a modal analysis is performed where an axial stress of N is applied in the direction of y axis. The dominent natural frequency is computed and compared with the analytical natural frequency (equation . 4.5) (W.Leissa 1969). Corresponding mesh density and load description is given in figure . 4.2b.

The natural frequency of a plate with membrane stress. 'a' is the length of the side.

$$\rho\omega_{mn}^2 = D\left[\left(\frac{\pi}{a}\right)^2\right]^2 + N\left(\frac{\pi}{a}\right)^2 \tag{4.5}$$

The convergence of Natural frequency for a square plate with membrane load is given in the figure:4.2a. It is clearly noted that the QUAD4 element converges much faster then than PAT element. One point to notice is that, the loads are applied such that the membrane-bending interaction stiffness contribution is very high such that it overshadows pure bending stiffness. Apparently this is the case for actual loading in hot dip galvanization application(figure.A.3).

To study the behavior of the plate with imposed displacement and transverse forces,an analysis is done on a long metal strip of length 10m, width 1m figure. 4.3. The plate is

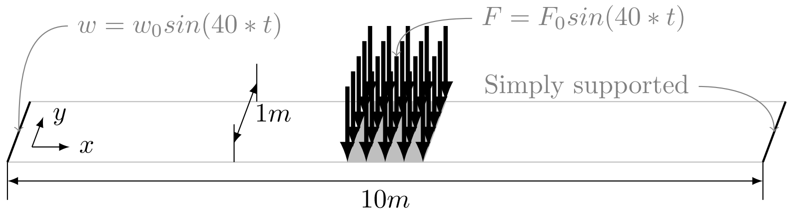



Figure 4.3: Strip with imposed displacement and transverse load

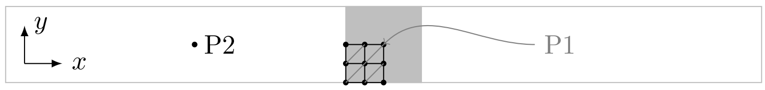



Figure 4.4: Mesh density of strip

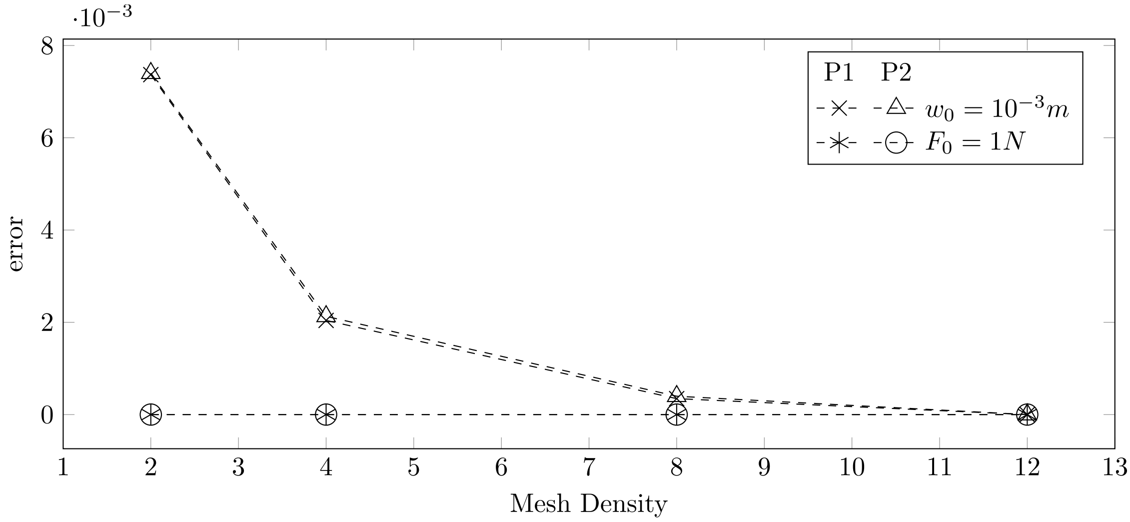



Figure 4.5: Comparision of solution between $w$ and $F$ for PAT element

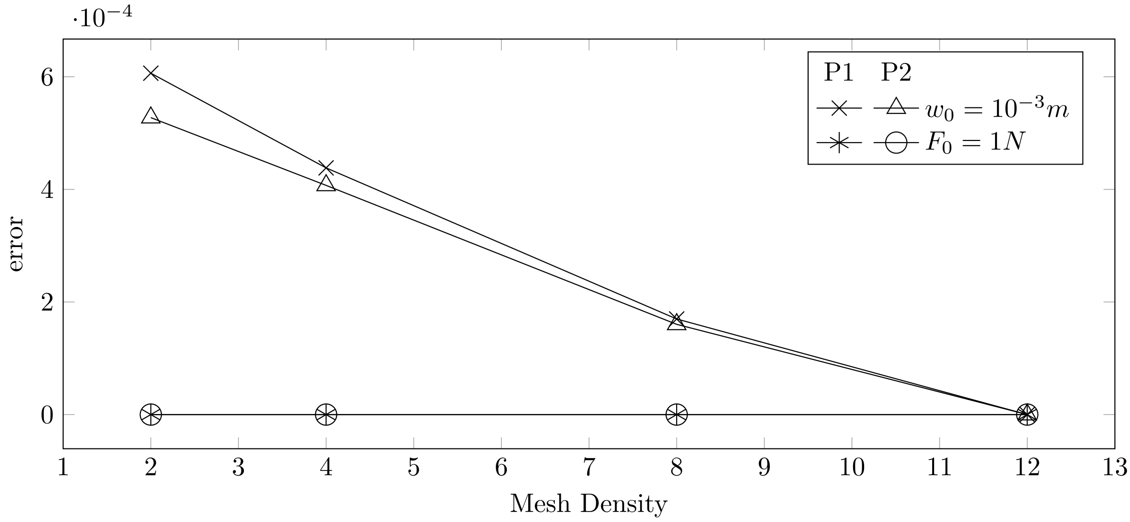



Figure 4.6: Comparision of solution between $w$ and $F$ for QUAD4 element

simply supported on one end and a sinusoidal displacement on other end and sinusoidal distributed load is applied on the middle of the plate. The figure. 4.4 shows the mesh of the strip with density of 2. Two Point P1 and P2 are used for measuring the results and error is calculated.

PAT element is analyzed first. Imposed sinusoidal displacement of strip with increasing mesh density. The data is measured at two points P1 and P2. P1 is at the center of the strip and P2 is 2.5 m away from where the displacement is imposed(fig:A.4). Figure . 4.5 shows the result of the study. It evident that the result converges, but error is higher for small mesh density. It is noted that position of error also does not have huge impact. In same figure, the convergence of strip with transverse load is also plotted. It is clear that the transverse load has little to no effect on mesh density.

When the QUAD4 element is provided with the sinusoidal boundary condition the solution takes longer to converge fig : 4.6. But again, when the transverse load is applied it also has no effect on mesh density. From this analysis it clear that when there is imposed displacement at the boundaries the mesh density has to very fine but at the same time the mesh density has little effect of transverse load.

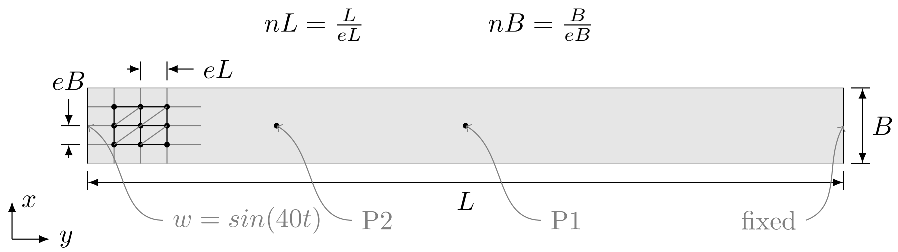



Figure 4.7: Directional mesh density

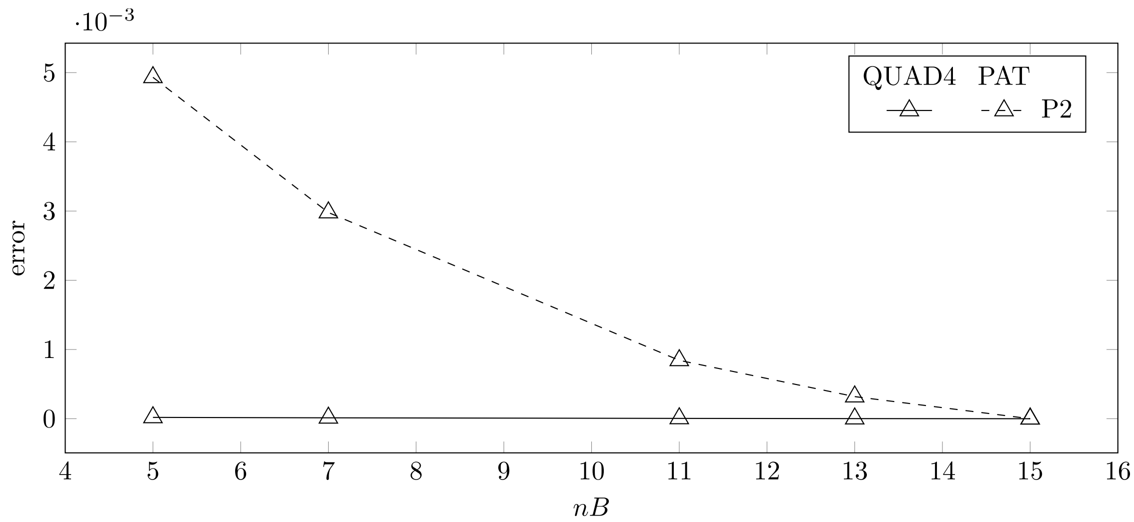



Figure 4.8: Comparison of elements with different directional mesh density in x direction

To understand the effect of directional mesh density, a study is done where the number of elements in one direction is increased consistently and in other direction is kept. This analysis also helped in understand the effect of mesh density to the direction in which

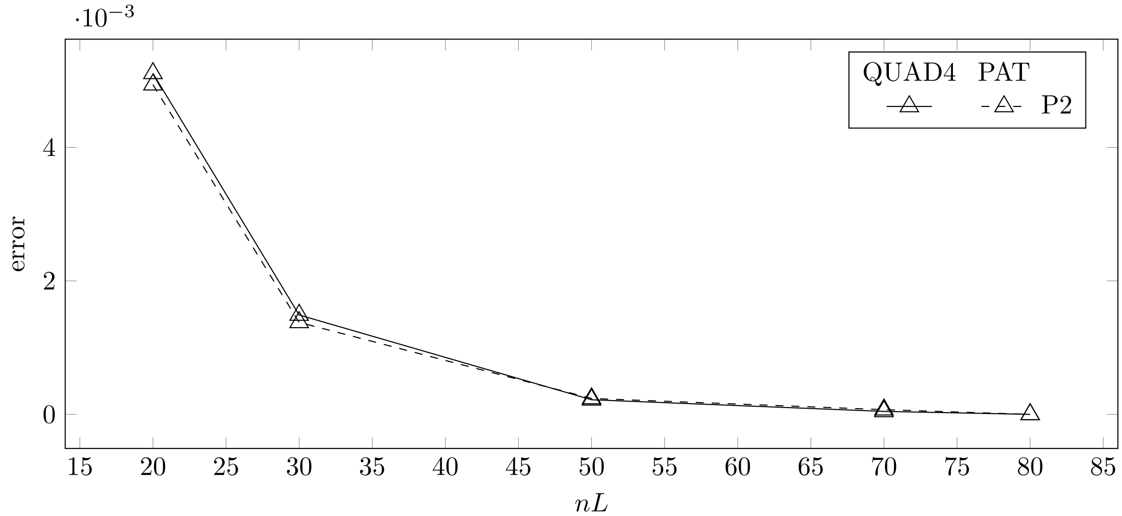



Figure 4.9: Comparison of elements with different directional mesh density in y direction

material is moving. For the study, same strip is taken but the meshes are not prefect squares instead they are rectangles whose aspect ratio changes according to the given $nL$ and $nB$ values. $nL$ is the number of elements in on y direction similarly $nB$ is the number of elements in x direction(figure.4.7). Figure. 4.8 shows the effect of direction density, if the mesh density is increased in direction perpendicular to material transport velocity. It is clear that the QUAD4 element is not affected. But the PAT element is affected by it, It may be because of the shape of the element. Figure . 4.9 shows the effect of results when the mesh density is increased in the parallel direction of line speed. It is evident from this analysis that the element size has to be very fine in the direction of material moving.

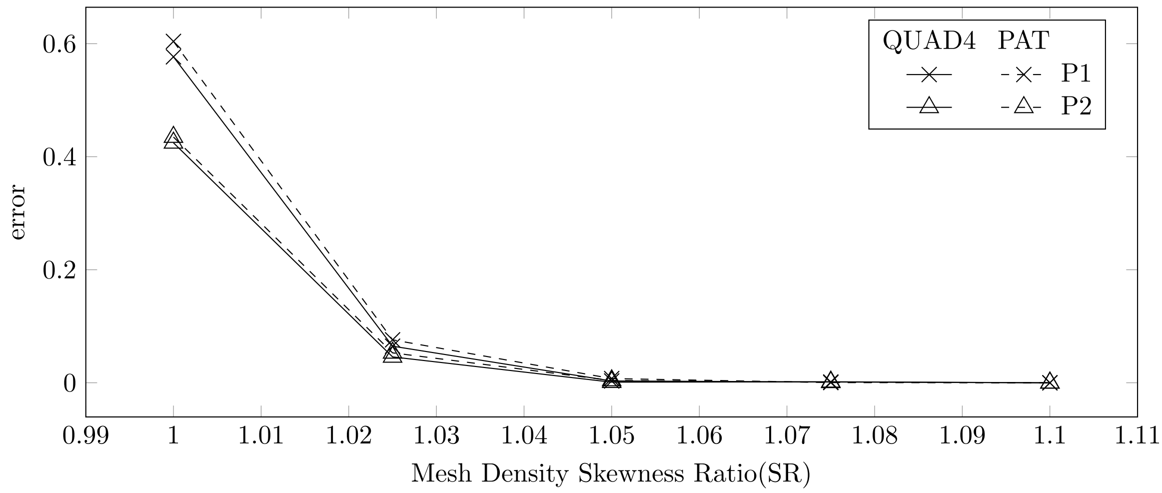



Figure 4.10: Comarison of PAT and MITC4 elements for different mesh density skewness

From the previous study, it is clear that the mesh density in the direction of the axial velocity is very important. It is not clear whether the over all element size affect the result or the element size at the boundary at which the time dependent Dirichlet load in applied. Skewed Mesh density is applied on a strip to understand its effect (figure .4.11).

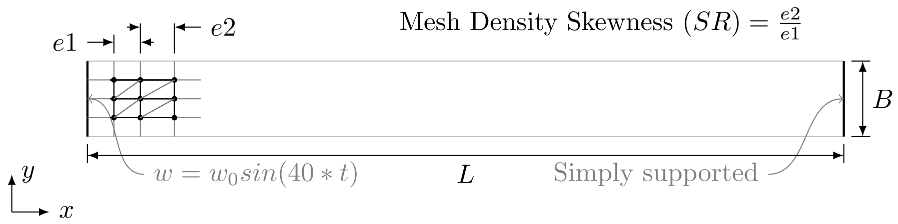



Figure 4.11: Mesh density skewness

The solution converges as the mesh density reached 1.05 and it stay converged as the skewness ratio increases even though the element size at points P1 and P2 increases with large mesh density skewness 4.10. Which tell us that the element size has to be very small only on the boundary where the Dirichlet load is applied and also at the direction of the line speed. This idea is also enforced by the fact that the convergence rate between the points P1 and P2 is not vastly different.

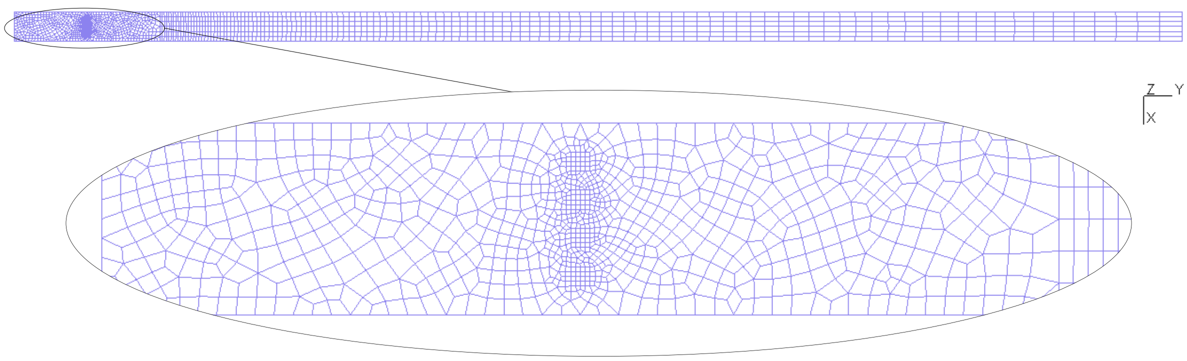


Figure 4.12: Optimized FE mesh

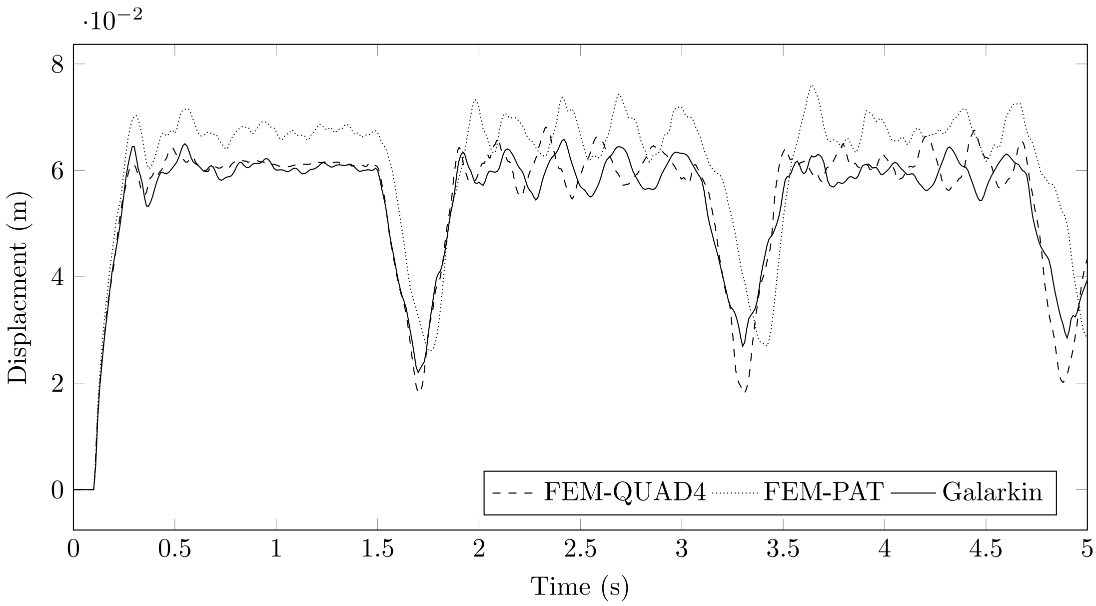



Figure 4.13: Comparison of FEM with existing Galarkin method

The Finite element program that is created during this thesis is compared with the existing Galerkin based numerical simulation tool. Result with QUAD4 element agrees with the solution of the Galerkin method figure.4.13. But the FEM solution with PAT element has noticeable difference in the result. The solution is in the figure. A.5.

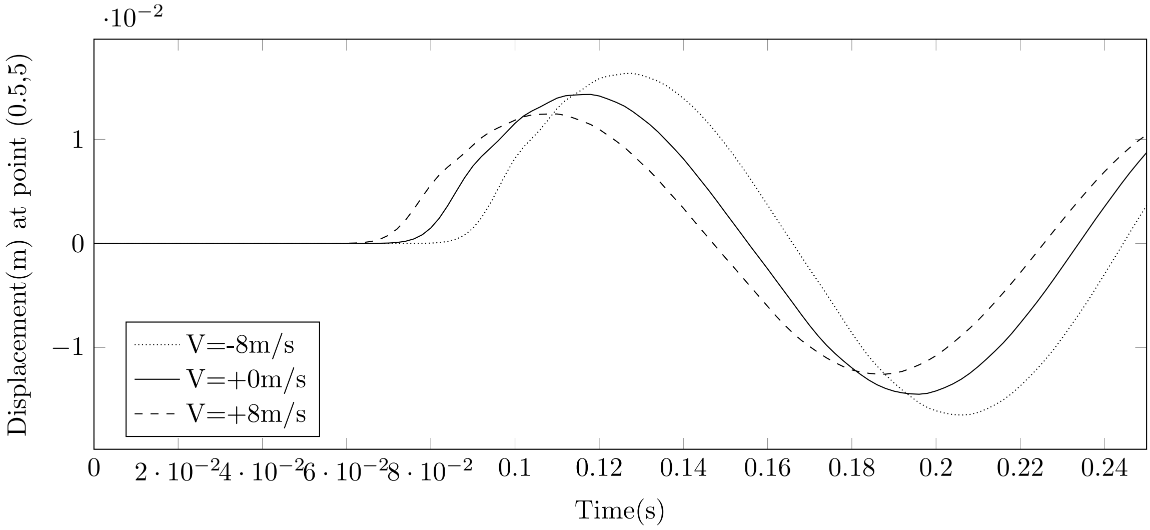



Figure 4.14: A plate with different axial velocity

Strip with different line velocity is studied and the result are plotted in figure.4.14. The solution agrees with the analytical solution calculated.

# Chapter 5

# Conclusion

The vibration of Metal Plate in the Hot-Strip Galvanization process is numerically modeled in this thesis. Hamilton principle is used to derive the equation of motion in weak form. A form of Euler Lagrange formulation is used to define the velocity of the plate. Inclusion of axial velocity makes Coriolis and centripetal acceleration components appear in the equation which makes it gyroscopic. This makes it a little different from the regular plate vibration. Both Kirchhoff thin plate and Reissner - Mindlin Plate theories are used to derive the weak form. A QUAD4 plate element is used to model the plate using Reissner - Mindlin Plate theory and PAT element is used for Kirchhoff plate.

In results and conclusion, both the elements are compared using various studies. In most of the cases the QUAD4 element performs better. Particular when there is point load and simply supported boundary condition. PAT element shows good convergence property for distributed load. Contrary to statement that PAT element is the one of the best plate element in literature, the PAT element shows significantly Poorer convergence than QUAD4 element. Many mesh dependency studies were conducted. After analysis a data, a optimized mesh is created that is promising for the most cases. The solutions of this FEM program is also converges to the solution given by existing Galerkin method that is developed in ArcelorMittal.

The finite element code using MATLAB programming language is developed. Object oriented programming style is used to code the FEM program because for it's intuitive programming approach. Most of the Inputs and outputs like loads , boundary condition, material properties, probes etc., are programmed as user defines classes. Which makes it possible to create multiple instance of classes for same simulation and also enables us to have operator overloading between the objects of these classes. Much importance is given to making the program user friendly as possible. In mean time, attention is paid to make it efficient. Techniques like pre-factorization and Modal Superposition techniques drastically decreased the solution time and makes it possible to solve in real time frame.Efficiency and accuracy finite model makes it a good competitor in numerical simulation world. Instead of directly solving them, The FEM program also output the matrices in state-space form which makes it suitable for integrating into MATLAB Simulink tools. Once solved the FEM program exports the result in POS file format.

Despite close attention paid to the boundary condition, more work needs to be done to improve the accuracy of the program. Multiple physics are involved in Hot-Dip galvanization process like fluid structure interaction between zinc and metal plate and interaction of plate and air near the air knife. There is also the several hundred $^{\circ}C$ temperature difference between top and bottom part of the plate and also the electromagnetic forces in the metal strip. Most prominent one is the interaction between air and the strip. High pressure air is blown into the metal plate which induces more vibration. In literature many people addressed this issue by directly simulating Fluid structure interaction (FSI) problem and others by including added mass effect. In future developments these could be include to included to increase the accuracy.

QUAD4 element proven its accuracy for this case, but there other much more accurate elements like MITC family of elements exists in the literature. MITC4 element would be a suitable candidate as it conforming and passes Patch test. In future work, time could be spend on efficient Plate elements. The contact between the Plates and the Rollers is not addressed here. Including contact behavior increases the usefulness of this code, The code can easily modified to simulate other location in the same process or in different process all together, which involves moving plate. Another important issue that need to be addressed in the bending plastic deformation caused by the rollers. This will cause bending of plate in the air knife. This is called as 'Cross-bow' effect or 'Spring-back effect' effect. This will lead to uneven coating of zinc. There is no immediate solution is in sight for this problem. More research is needed to understand this issue and to come up with a simplified way to address it in numerical simulation.

In overall conclusion, this FEM program is able to efficiently simulate to basic vibration of the metal plate. But in the long run, more complex behaviors need to addressed. There are still more room for increase in solution speed and accuracy. Inclusion of Modal Order Reduction technique can drastically increases the Practicality of the program in the real-time active vibration control.

# Reference


[1] G. Iosilevich Alexander. “An Analysis of finite element for plate bending problems”. MA thesis. Massachusetts Institute of Technology, 1994.

[2] Klaus-Jürgen Bathe and Eduardo N. Dvorkin. “A four-node plate bending element based on Mindlin/Reissner plate theory and a mixed interpolation”. In: *International Journal for Numerical Methods in Engineering* 21.2 (1985), pp. 367–383.

[3] Michael Baumgart et al. “Elasto-plastic bending of steel strip in a hot-dip galvanizing line”. In: *Acta Mechanica* 228.7 (July 2017), pp. 2455–2470.

[4] Johan Blaauwendraad. *Plates and FEM, Surprises and Pitfalls*. Springer New York Inc, 2010.

[5] Jer-Rong Chang et al. “Vibration and stability of an axially moving Rayleigh beam”. In: *Applied Mathematical Modelling* 34.6 (2010), pp. 1482–1497. ISSN: 0307-904X.

[6] L Chen. “Analysis and Control of Transverse Vibration of Axially Moving Strings”. In: *ASME. Appm. Mech. Rev.* 58 (2 2005), pp. 91–116.

[7] Li-Qun Chen and Xiao-Dong Yang. “Vibration and stability of an axially moving viscoelastic beam with hybrid supports”. In: *European Journal of Mechanics - A/Solids* 25.6 (2006), pp. 996–1008. ISSN: 0997-7538.

[8] A.J.M Ferreira. *MATLAB Codes for Finite Element Analysis*. Springer, 2009.

[9] M.H Ghayesh and Marco Amabili. “Three-Dimensional nonlinear planer dynamics of an axially moving Timoshenko beam”. In: *Arch Appl Mech* 83 (4 Apr. 2013), pp. 591–604.

[10] Hou-Cheng Huang. *Static and Dynamic Analysis of Plates and Shells*. Springer-Verlag New York Inc, 1989.

[11] Hendrih Katica. “Transversal vibration of the axially moving sandwich belts”. In: *Arch Appl Mech* 77 (7 July 2007), pp. 523–539.

[12] C.H. Kim, N.C. Perkins, and C.W. Lee. “Parametric resonance of plates in a sheet metal coating process”. In: *Journal of Sound and Vibration* 268.4 (2003), pp. 679–697. ISSN: 0022-460X.

[13] H Koivurova and E.M Salonen. “Comments on non-lineqt formulations for Travelling String And Beam Problems”. In: *Journal of Sound and Vibration* 225.5 (1999), pp. 845–856. ISSN: 0022-460X.

[14] H. Koivurova and A. Pramila. “Nonlinear Vibration of Axially Moving Membrane by Finite Element Method”. In: *Computational Mechanics* 20 (6 Nov. 1997), pp. 573–581.

[15] Hannu Koivurova. “Dynamic Behaviour of an Axially Moving Membrane Interacting With the Surrounding Air and Making Contact with Supporting Structure”. Oulu: University of Oulu, Apr. 1998.

[16] Phillip L.Gould. *Analysis of Shells and Plates.* 1st ed. Springer-Verlag New York Inc, 1988.

[17] Jian Li et al. “Research on Vibration Control Method of Steel Strip for a Continuous Hot-dip Galvanizing Line”. In: *ISIJ International* 52.6 (2012), pp. 1072–1079.

[18] Krzysztof Marynowski. *Dynamics of the Axially Moving Orthotropic Web.* Springer, 2008.

[19] D.B McIver. “Hamilton’s principle for systems of changing mass”. In: *Journal of Engineering Mathematics* 7 (1973), ”249–261”.

[20] Chen Li-Qun. “Nonlinear Vibrations of Axially Moving Beams”. In: *InTech,Nonlinear Dynamics* (2010).

[21] Tytti Saksa et al. “Dynamic Behaviour of a Travelling Viscoelastic Band in Contact With Rollers”. In: *Numerical Methods for Differential Equations, Optimization, and Technological Problems.* 2003. Chap. 22, pp. 393–408.

[22] M. Saxinger et al. “Dynamical Model of Axially Moving Steel Strips”. In: *IFAC-PapersOnLine* 49.20 (2016). 17th IFAC Symposium on Control, Optimization and Automation in Mining, Mineral and Metal Processing MMM 2016, pp. 190–195. ISSN: 2405-8963.

[23] Ian M. Smith and William Duncan. “The effectiveness of excessive nodal continuities in the finite element analysis of thin rectangular and skew plates in bending”. In: *International Journal for Numerical Methods in Engineering* 2.2 (1970), pp. 253–257.

[24] Bernhard Specht. “Modified shape function for the three node plate element passis the patch test”. In: *International Journal for Numerical Methods in Engineering* 26.3 (1988), pp. 705–715.

[25] A. Steinboeck et al. “Dynamical Models of Axially Moving Rods with Tensile and Bending Stiffness”. In: *IFAC-PapersOnLine* 48.1 (2015). 8th Vienna International Conferenceon Mathematical Modelling, pp. 598–603. ISSN: 2405-8963.

[26] S Timoshenko and S.Woinowsky Krieger. *Theory of Plates and Shells.* 2nd ed. McGraw-Hill Book Company, 1987.

[27] Arthur W.Leissa. *Vibration Of Plates.* National Aeronautics and Space Administration, 1969.

[28] Xiaodong Wang. “Numerical analysis of moving orthotropic thin plates”. In: *Computers and Structures* 70.4 (1999), pp. 467–486. ISSN: 0045-7949.

[29] W.Kwon Young and Bang Hyochoong. “Chapter 11 - Control of Flexible Structure”. In: *The Finite Element Method using MATLAB*. CRC press, 1997.

[30] Choi Ji-Yun, Hong Keum-Shik, and Huh Chang-Do. “Vibration Control of an Axially Moving Strip by a Non-linear Boundary Control”. In: *IFAC Proceedings Volumes* 53.1 (2002). ”15th IFAC World Congress”, pp. 1–6.

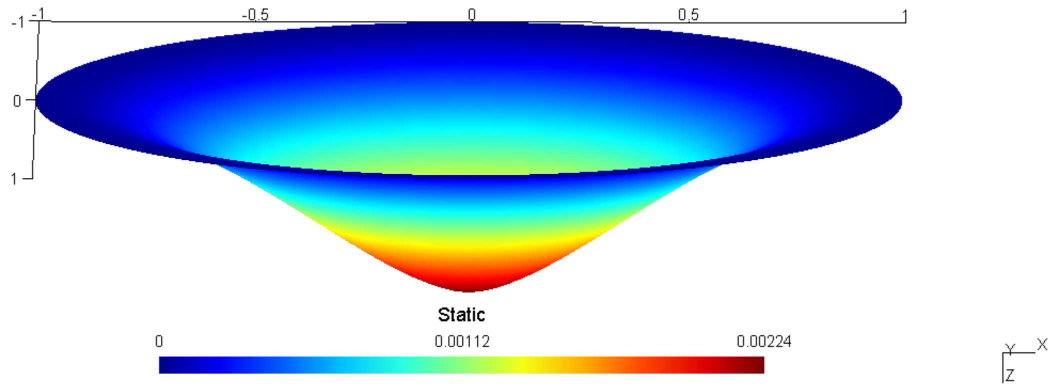
[31] O C Zienkiewicz and Y K Cheung. “The Finite Element Method For Analysis of Elastic Isotropic and Orthotropic Slabs.” In: *Proceedings of the Institution of Civil Engineers* 28.4 (1964), pp. 471–488.

[32] O.C Zienkiewicz and R.L Taylor. “Chapter 11 - Plate bending approximation : thin (Kirchhoff) plates and C1 continuity requirements”. In: *The Finite Element Method for Solids and Structural Mechanics*. 6th ed. Elsevier Science, 2005.

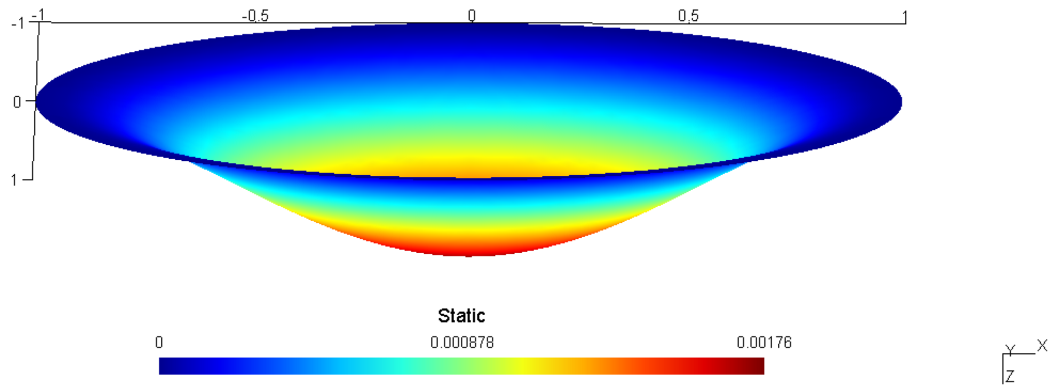
# Appendices

# Appendix A

# Solution Plots

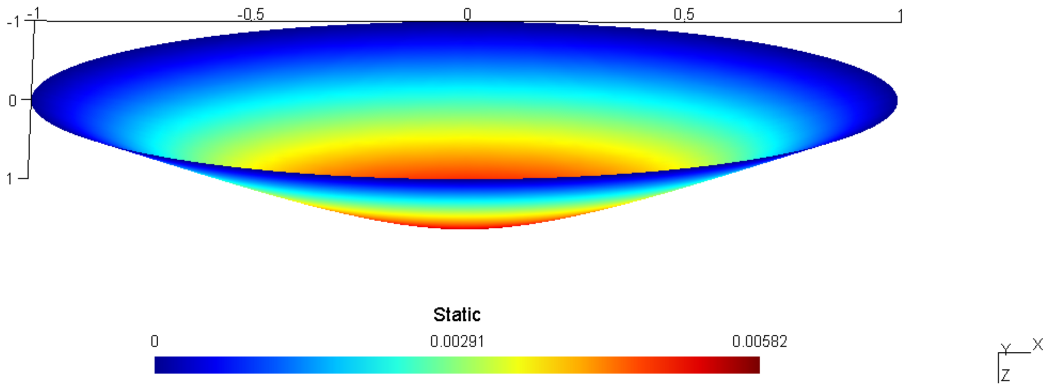



(a) Point Load (P)

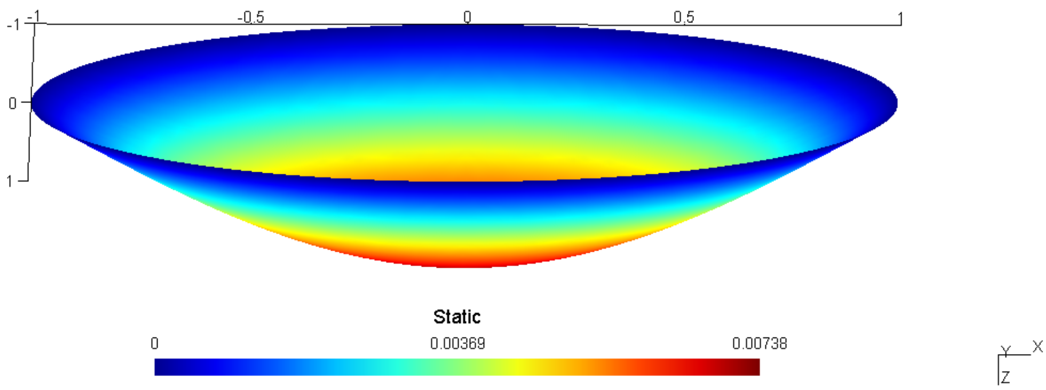



(b) Distributed Load (q)

Figure A.1: Displacement plot of a clamped circular plate

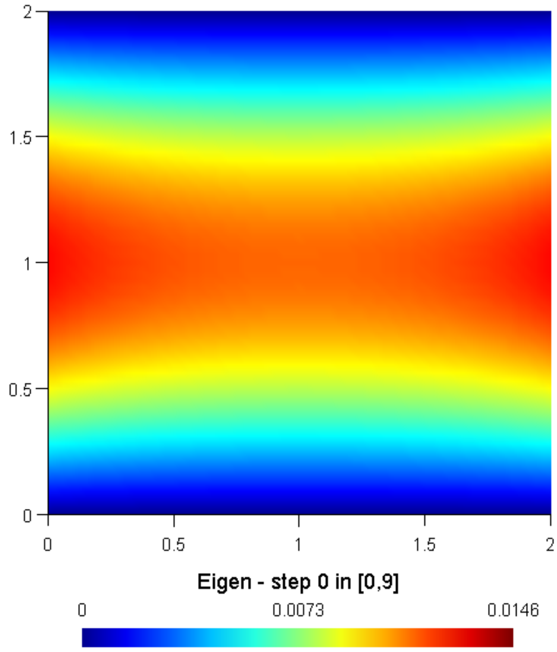



(a) Point Load (P)

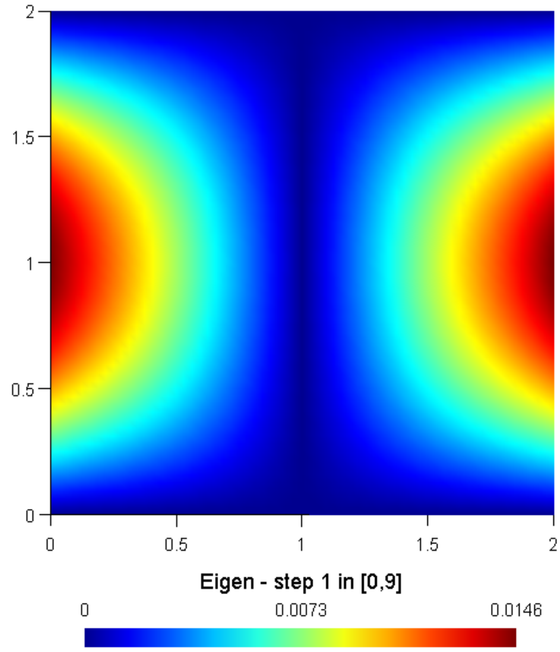



(b) Distributed Load (q)

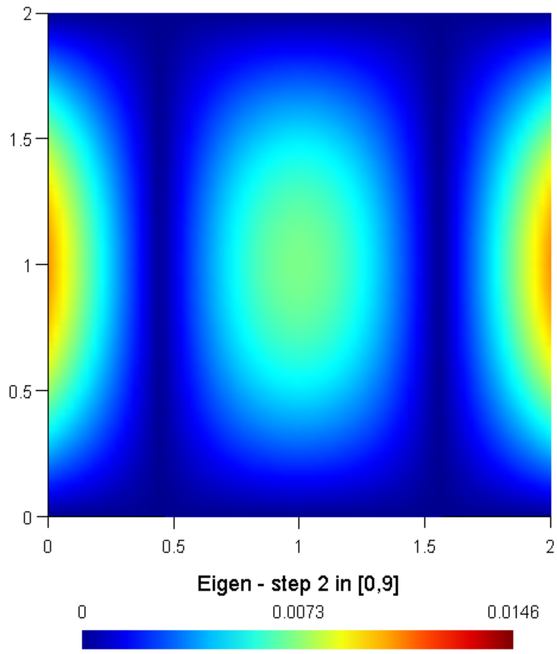
Figure A.2: Displacement plot of a simply supported circular plate



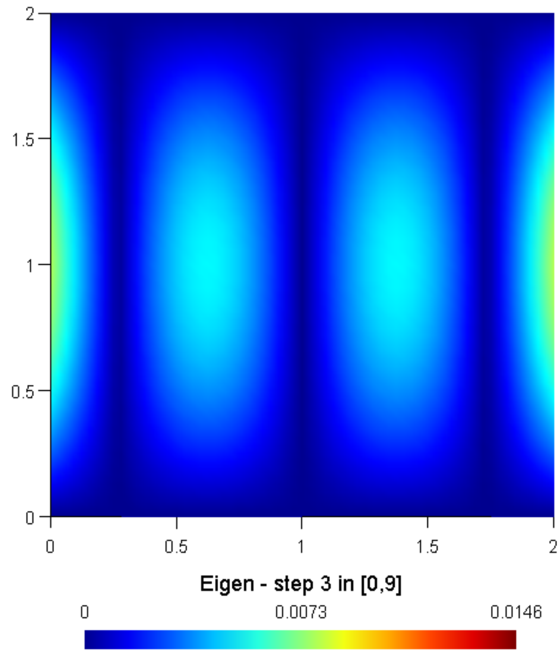

(a) Mode Shape 1




(b) Mode Shape 2

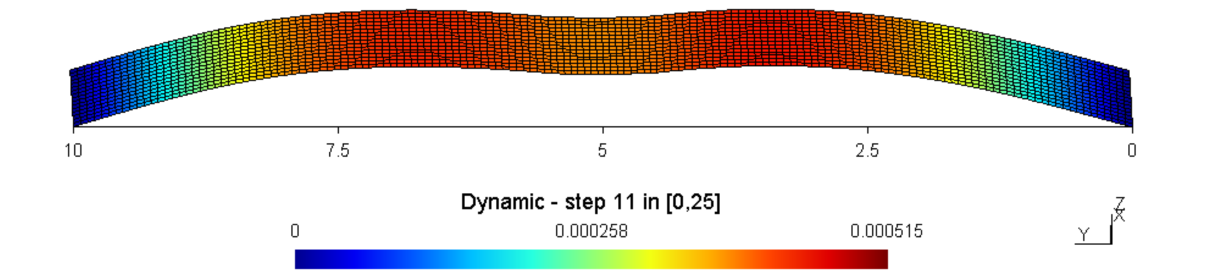



(c) Mode Shape 3

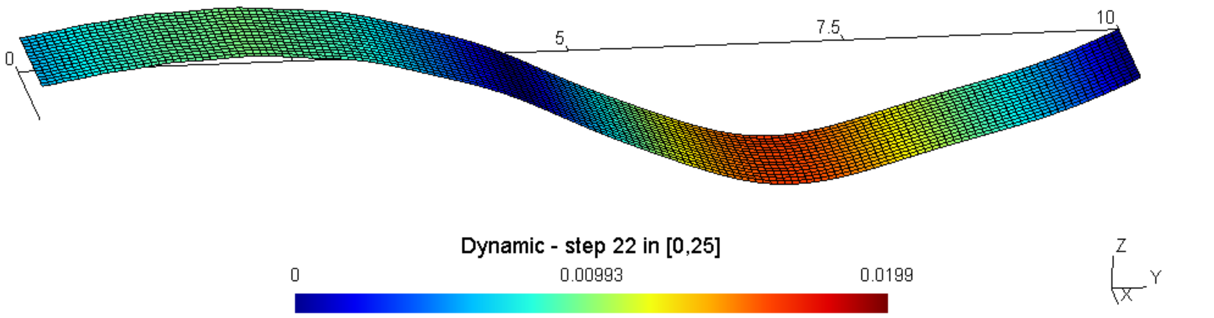



(d) Mode Shape 4

Figure A.3: Natural modes of a square plate under axial load




(a) Transverse distributed load at the middle




(b) Imposed Dirichlet boundary at $x = 0$

Figure A.4: Displacement plot of a plat under different loads

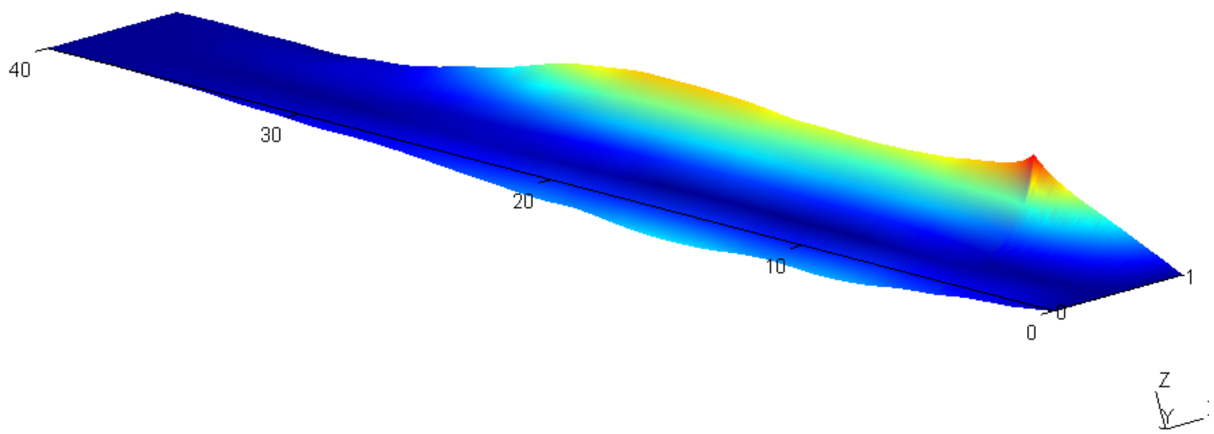



Figure A.5: FE solution of a metal strip with point load

# Appendix B

# User Guide for FEM MATLAB Code

This Guide is intended to provide basic instruction to use to `PLATE_FEM` finite element code. The basic structure of the `PLATE_FEM` is based on the Object Oriented Programming type where it contains many sub-classes inherited from few super-classes. All the class definitions are defined in a single folder named `PLATE_FEM` . To access these classes, Either run the script file in same folder or the folder can be placed in one of the MATLAB paths. In case of creating a user defined MATLAB path, use the option `Home - Set Path - Add folder` to locate the `PLATE_FEM` folder.

### Prerequisite files

Mesh file and Material properties file are a must to run the FEM program. Mesh file is generated by GMSH pre-processing tool with the file format `.MSH` . For the Material property input user type file must be placed in the working directory. It is recommended to use to file name `Mat.dat.txt` . The content of a example MatDat file is given below. It is highly insisted to follow this format. Currently, Linear Isotropic material is only supported.

```
TYPE        = Linear_Isotropic
UNIT        = SI
E           = 1E11
NU          = 0.3
RHO         = 7000
N           = 4E7
V           = 10
THICKNESS   = 0.001
```

### Initiating the program

Since `PLATE_FEM` code generate many output files, It is advised to create a separate a folder for each simulation and place all the prerequisite files in the same folder or any of the sub folders. Run the Matlab script file from this working folder. It is highly advised to clear all pre-exisiting data.

```
clear;
clc;
```

To initiate the study, create an object of the class that corresponds to the indented study. `STATIC` , `MODAL` and `DYNAMIC` are the different existing simulation types. All of them inherits from the super class `PLATE_FEM` .

```
FEM=DYNAMIC("studyname");                % To create a  dynamic study
FEM=MODAL("studyname");                     % To find a natural frequency
FEM=STATIC("studyname");                       % To solve static problem
```

If the script is re-run with same `studyname` , it will overwrite all existing data. To prevent overwriting either change the `studyname` or run it in a different folder. In order to import the material data file and mesh file. Code given below should be used.

```
FEM.ReadMesh("strip_tri.msh");          % Mesh File Input
FEM.Mat(1)=MATDat("Mat.dat.txt");       % Material file name input
```

If the ReadMSH file is exceuted without a file name it will look for a mesh with `studyname` . If nothing is found it will throw an exception. Multiple objects ( `FEM.Mat(1)` $\cdots$ `FEM.Mat(n)` ) of the class ( `MATDat` ) can be created but currently only one material can be successfully applied to the element. Multiple material in a single domain can be applied but it is not fully tested.

### Loads and Boundary condition definition

`Dirichlet` class is used to impose boundary displacement. It has two sub-classes namely `DirichletOnPhyEn` and `DirichletOnNode` . Boundary condition definition must be given using the objects of either of the sub-classes. Basic layout of input is given below.

```
FEM.Up(iu1)   =DirichletOnPhyEn(FEM, PhyEn, NodalDOFS, value   );
FEM.Un(iu2)   =DirichletOnNode  (FEM, Nodes, NodalDOFS, value  );


% FEM = Object of the class PLATE_FEM
% Up  = Object of DirichletOnPhyEn class
% Un  = Object of DirichletOnNode class
% iu1, iu2 = Dirichlet reference number (multiple boundary conditions can b
```

```
% PhyEn = Physical entities values (ex: [2] or [1 2 3] )
% PhyEn entity can be a point or line or area
% NodalDOFS = To set the degree of freedom for which the displacement is in
% value  = displacement value (if value = 0 fixed and value not equal to 0
```

Similar layout to the Dirichlet is used for the Neumann boundary condition. Two subclass are defined `NeumannOnPhyEn` and `NeumannOnNode`. `NeumannOnPhyEn` can only apply transverse point load and transverse distributed load on a `PhyEn`.

```
FEM.Np(in1)  =NeumannOnPhyEn(FEM, PhyEn, NodalDOFS, value  );
FEM.Nn(in2)  =NeumannOnNode (FEM, Nodes, NodalDOFS, value );
```

```
% FEM = Object of the class PLATE_FEM
% Fp  = Object of NeumannOnPhyEn class
% Fn  = Object of NeumannOnNode class
% in1, in2 = Neumann reference number (multiple boundary conditions can be
% PhyEn = Physical entities values (ex: [2] or [1 2 3] )
% PhyEn entity can be a point , line and are
% NodalDOFS = To set the degree of freedom for which the displacement is in
% value  = displacement value (if value = 0 fixed and value not equal to 0
```

### Probe Definition

Probes are used to measure the displacement value at a designated node or PhyEn or a coordinate. For each time step, the average value is stored. This data can be exported to a file or can be plotted.

```
FEM.Pp(ip1)=ProbeOnPhyEn(PhyEn,NodalDOFS);
FEM.Pn(ip2)=ProbeOnNode(node,NodalDOFS);
FEM.Pc(ip3)=ProbeOnCod(coordinate,NodalDOFS);
```

```
% FEM = Object of the class PLATE_FEM
% Pp  = Object of ProbeOnPhyEn class
% Pn  = Object of ProbeOnNode class
% Pc  = Object of ProbeOnCod class
% ip1, ip2, ip3 = Neumann reference number (multiple multiple probes for sa
% PhyEn = Physical entities values (ex: [2] or [1 2 3] multiple Phy definit
%node = Nodes to measure the value (ex: [23 25] multiple node definition w
% NodalDOFS = To set the degree of freedom for which the displacement is in
%coordinate = coordinate at which the solution is measured (ex : (0.5,5) th
```

After solving the `DYNAMIC` or `STATIC` problems, below commands can be used to write the probe data file or to plot the data.

```
FEM.WriteProbe();
FEM.PlotProbe();
```

#### Special commands for `DYNAMIC` class

Since dynamic problems require additional information to solve. Certain commands exist specially for DYNAMIC class. One such command is

```
FEM.T=FEMTime(0.01,1);   %Solution time 1s,Time step size 0.01
```

`FEMTime` class is like a solver clock. In here, we input the time step size and solution time. This class is created such that any modification with in the class will not affect any other functions of the program. So that, in the future automatic time stepping algorithms can be included.

```
t=linspace(0,10,2001);
x=sin(25*t);
FEM.TS(1)=TimeSeries(t,x);
```

`TimeSeries` class is used to store external time varying data. The data can be from a file or from a function (like sin function given in this example). The object `FEM.TS(1)` of the class `TimeSeries` can be operated with the objects of the `Neumann` and `dirichlet` boundary conditions so that time varying loads can be applied.

```
FEM.InitialX('zero');
FEM.InitialV('zero');
```

These commands are used to set the initial displacement and velocity for the `DYNAMIC` problem. These commands are a must for the `DYNAMIC` problem. Currently, only `'zero'` initial displacement and velocity can be applied. For non-zero initial condition code modification is required.

#### Applying Domain, loads and boundary condition

In order to solve the problem, the area solution domain has to be defined. It includes all the 2D domain where loads are defines. Failing to do, might result in a singular matrix. Following commands is used to define the domain.

```
FEM.SetDomain(PhyEn,MatId,'element_type');

% PhyEn  = All the Area Physical entities (ex:[1 21 22 23 24])
%MatID = Material Id
```

Table B.1: Operator overloading reference table

| | | | |
|---|---|---|---|
| $U + U = U$ | $U + C = U$ | $U \times FT = U$ | $U \times C = U$ |
| $U \times TS = U$ | $U \times P = U$ | $F + F = F$ | $F + C = F$ |
| $F \times FT = F$ | $F \times C = F$ | $F \times TS = F$ | $F \times P = F$ |
| $P \times C = P$ | $P + P = P$ | $P + C = P$ | $P - P = P$ |
| $Int(P) = P$ ($\int Pdt$) | $P/FT = P$ | $P.forT(C) = P$ ($P(n + C)$) | $cos(FT) = FT$ |
| $sin(FT) = FT$ | $FT * FT = FT$ | $FT.forT(C) = FT$ ($FT(n + C)$) | $FT * C = FT$ |
| $FT - FT = FT$ | | | |

```
%element_type = type of element (ex:'QUAD4' or 'PAT')
```

Multiple `SetDomain` command can be used to define different material properties. But this functionality is not fully tested. To apply already defined Neumann and Dirichlet loads, following options are used.

```
FEM.ApplyF(FEM.Np(1))
FEM.ImposeU( FEM. Un(1))
```

Same command can also be used to define operators between many of the objects. this functionality is described in next section.

**Operations between `Neumann` ,`Dirichlet` ,`Probe` ,`TimeSeries` ,`FEMTime`**

Operator overloading between objects of different classes is provided which enables us to give complex boundary conditions with ease. Also enables us to provide relations between different variables of the FEM program. Table : B.1 shows all the enabled operators. The right hand side of the operations in the table denotes the return object type.

U = `Dirichlet` ,`DirichletOnNode` and `DirichletOnPhyEn`
F = `Neumann` ,`NeumannOnNode` and `NeumannOnPhyEn`
P = `Probe` ,`ProbeOnNode` ,`ProbeOnCod` and `ProbeOnPhyEn`
TS = `TimeSeries`
FT = `FEMTime`
C = Any double (ex:C=2)

For example, in order to apply sinusoidal Dirichlet boundary condition one of the following snippets can be used.

```
t=linspace(0,10,2001);
x1=sin(25*t);

FEM.TS(1)=TimeSeries(t,x1);
FEM.Up(2)=DirichletOnPhyEn(FEM, [11], [1 0 0], 0.01 );
FEM.ImposeU( FEM. Up(2) * FEM.TS(1) );
```

(or)

```
FEM.T=FEMTime(0.01,3);

FEM.Up(2)=DirichletOnPhyEn(FEM, [11], [1 0 0], 0.01 );
FEM.ImposeU( FEM. Up(2) * sin(40*FEM.T));
```

To input general control formula given in the equation : B.1, following code snippet can be used.

$$F(t_{n+1}) = -(K_1 \cdot U(t_n) + K_2 \cdot \frac{\mathrm{d}U}{\mathrm{d}t} + K_3 \cdot \int_{t_0}^{t_n} U dt) \tag{B.1}$$

```
FEM.Pp(1)=ProbeOnPhyEn([21 22 23 24],[1 0 0]);
FEM.T=FEMTime(0.01,3);

FEM.Fp(1)=NeumannOnPhyEn(FEM, [22 23 24 21], [1 0 0], 1);

K1=10;        K2=2;        K3=10;

P1 = K1 * FEM.Pp(1).forT(0);
P2 = K2 *(FEM.Pp(1).forT(0)-FEM.Pp(1).forT(-1))/(FEM.T.forT(0)-FEM.T.forT(-
P3 = K3 * Int(FEM.Pp(1));
FEM.ApplyF( -1 * FEM.Fp(1) * (P1 + P2 + P3) );
```

### Solution

```
FEM.Solve();
```

This is a common command for all three sub-classes `STATIC` , `MODAL` and `DYNAMIC` . This command will run the basic solution routine of the corresponding sub-class. For `STATIC` there are no additional options available. But for `MODAL` , two solution procedure are available. Just using the command `FEM.Solve();` will solve the Eigen problem using `eig()` function. This function calculates all the Eigen modes and values, which is very expensive. But sending a constant like this `FEM.Solve(20);` , makes use of `eigs()` function. `eigs()` option only explores the subspace of the full eigen space. `FEM.Solve(20);` command will only calculate first 20 natural frequency and modes.

For the `DYNAMIC` class, `FEM.Solve();` will solve the system with regular Newmark algorithm (refereed from ANSYS theory manual) given below but this is an expensive operation.

$$R = F_t + \mathbf{M}\left(a_0 u_t + a_2 v_t + a_3 a_t\right) + \mathbf{C}\left(a_1 u_t + a_4 v_t + a_5 a_t\right)$$
$$u_{t+1} = \left[a_0 \mathbf{M} + a_1 \mathbf{C} + \mathbf{K}\right]^{-1} R$$
$$v_{t+1} = a_1\left(u_{t+1} - u_t\right) - a_4 v_t - a_5 a_t$$
$$a_{t+1} = a_0\left(u_{t+1} - u_t\right) - a_2 v_t - a_3 a_t$$

$$a_0 = \frac{1}{\alpha h^2} \qquad a_1 = \frac{\theta}{\alpha h} \qquad a_2 = \frac{1}{\alpha h}$$
$$a_3 = \frac{1}{2\alpha} - 1 \qquad a_4 = \frac{\theta}{\alpha} \qquad a_5 = \frac{h}{2}\frac{\theta}{\alpha} - 2$$

$$\theta \geq \frac{1}{2}$$
$$\alpha \geq \frac{1}{4}\left(\frac{1}{2} + \theta\right)^2$$

The problem with constant time step size and time independent material property, the matrix that need to be inverted does not change for each time step. we can exploit this feature to increase the solution speed. The matrix needed to be inverted is pre-factorized, which increases the efficiency to many fold. To use this feature, following command is used.

```
FEM.SolveLU();
```

Still this matrix is large and takes lots of time to solve. To further increase the solution speed Modal - superposition technique is used.

```
FEM.SolveLU_RED(n_base);
```

Command will construct `n_base` number of reduced basis and project the basis to the full matrix. This only leave the system with ( `n_base` ) unknowns. To decide the number of ( `n_base` ), it is highly recommended to run the `MODAL` analysis with the command `FEM.Solve(n_base);` and see if the highest natural frequency is well within the safety limit.

### Exporting to state-space form

In order to use the finite element system model with the Simulink module. the FEM ODE is exported into State-Space form using the command.

```
[A_DF,B_DF,C_DF,D_DF]=FEM.GetMat_RED(n_bas)
```

This command will export the reduced model with ( `n_bas` ) number of states. Since the full model is very large so it is very hard to converge. Once solved using Simulink, the results can be extracted and stored in FE matrix format using

```
FEM.ExtractFromSS(Strip1DState)
```

#### Writing Pos file

```
FEM.WritePos()
```

Command will write the solution in a format compatible with the GMSH tool. For the `MODAL` analysis, additional input in this command will let us only store the required natural modes.

```
FEM.WritePos([1 2 3 5 10])
```

#### Additional Options

By default, `PLATE_FEM` command stores all matrix in full matrix format, since full matrix format is faster and easy to display. But the full matrix format will occupy very large memory. This sometimes results in `MEMORY ERROR` . To prevent this error, the major matrices can stored in a sparse matrix format. To enable this option, go to the file `PLATE_FEM` and in the properties section, change `issparse = 0;` into `issparse = 1;` .

In the `PLATE_FEM` , by default only point, line , triangle and quadrangle elements are enabled. To turn on or off any new or existing element, go to the file `FEMMesh` . In the constructor function, find `elemprop1.access` and type '1' to turn on and '0' to off the element in the corresponding location.

`PLATE_FEM` program output a log file, which helps in tracking the progress of the solution. If the log data is not required to be displayed in the command window, it can be turned off by typing `obj.displayon=on;` in the `LG` file constructor.