iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** I **Month of publication:** January 2022

DOI: https://doi.org/10.22214/ijraset.2022.39902

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue I Jan 2022- Available at www.ijraset.com

Finite Element Analysis using MATLAB

Rahul Yadav
Department of Mechanical Engineering, K J Somaiya College of Engineering

Abstract: ***Finite Element Analysis or FEA is the simulation of a physical phenomenon using a numerical mathematic technique referred to as the Finite Element Method or FEM. This process is at the core of mechanical engineering, as well as a variety of other disciplines. It also is one of the key principles used in the development of simulation software. Engineers can use these FEM to reduce the number of physical prototypes and run virtual experiments to optimize their designs. Finite Element Analysis is used in various fields like structural, fluid flow, heat transfer to estimate the behavior of a component in real environment. There now exists growing body of knowledge connected with the development of mathematical models and numerical simulations of physical model. There are various software packages like Ansys, OptiStruct, COMSOL, Solidworks and many more which provide a close estimate in simulation models. Apart from these software MATLAB also has a Partial Differential Equation (PDE) toolbox which enables us to perform these simulations using some built-in functions and codes. However, a computational numerical technique is not an end of design rather it just provides a great estimate of the final component for which the results are only as good as the input provided and the final component in most of the cases require a physical testing in environment which it is meant to perform as a validation.***

Keywords: Brake pedal, MATLAB, Partial differential equation (PDE) toolbox, Tetrahedral mesh.

I. INTRODUCTION

The main goal of the paper is to make use of MATLAB to create a model for static structural analysis of a basic component and to demonstrate how the model can be made user friendly so that it can be easily used by anyone. The model was made first in MATLAB script and then the script was used to create a MATLAB application. As Finite Element Analysis (FEA) discretises a physical domain into smaller elements. The equations in FEA describe physics of these individual elements, which are then assembled into a larger system of equations that models the entire domain.

A basic component - FSAE Brake pedal is selected for the FE analysis on MATLAB model and then the model was validated using other simulation software- Ansys and Solidworks. Different FE analysis aspects such as various boundary conditions, load application and meshing are discussed keeping the MATLAB model in centre and comparing it with that of other two software.

II. OVERVIEW

A. General Finite Element Method (FEM) Process

Most of the FEA software have pre-processing, solver and postprocessing as the basic three steps in simulating a model. Pre-processing in involves adding geometry, material and physical constraints. Solver basically includes meshing, creating and solving the mathematical equations, it is considered the brain of the software. Finally, the results are analysed in postprocessing step.

B. Process followed in MATLAB Model

As most of the software, process followed in development of the model was similar. See Fig.2. The process in the model starts with the selection of the geometry file. Second step is to add material properties and then the physical constraints and loads on the component. Next step in the model is to mesh the geometry this is done by defining the mesh size. Finally, we solve and analyse the results.

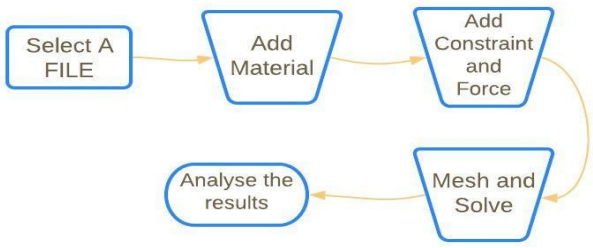


Fig. 1. MATLAB model process

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue I Jan 2022- Available at www.ijraset.com

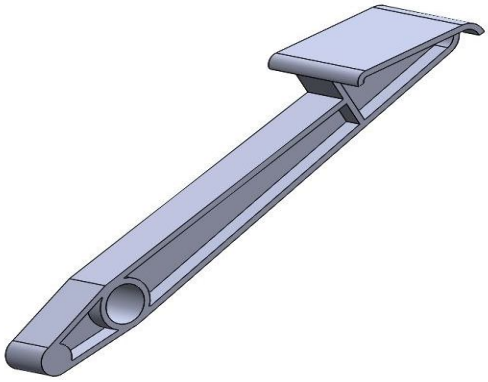


Fig. 2. CAD Model of Brake pedal

III.MATLAB MODEL

After discussing flow of the model in the preceding note, each step in the flow is discussed in detail.

A. Opening the File

MATLAB supports the import of ".stl" (stereolithography) for 3D objects. The selection of file was done by calling "uigetfile" function in live editor. As the file is selected it opens and the faces are numbered, this is to make sure faces which are to be constraint and where force is applied are known. A Partial Differential Equation (PDE) model is defined and the selected geometry is passed for further processes.

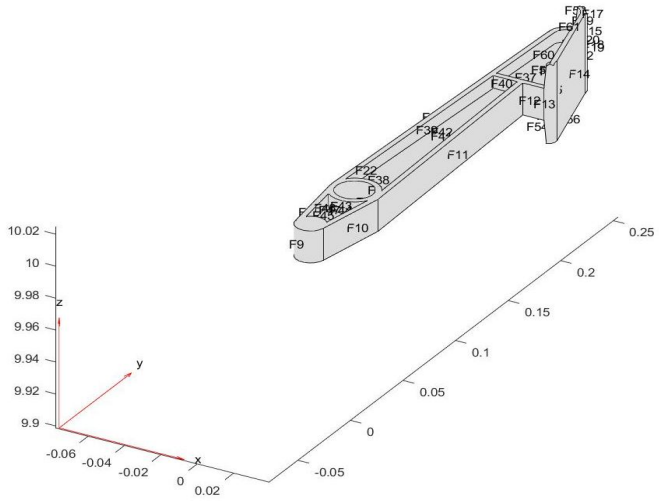


Fig. 3. View of file in MATLAB model

B. Defining Materials

Next step is to define the material, the PDE model created in MATLAB has various predefined functions for performing FE analysis, one such function for defining material properties is "StructuralProperties" the input arguments should be written in "name-number combination", here we are using Aluminium-7075 and its Young's modulus and Poisson ratio are given as input arguments.

C. Boundary Conditions and Loads

The third step is to define the boundary conditions in the model, the function used here is "SturcturalBC" the input arguments are written is same format. Here the input arguments are the face number that are to be constraint and the constraint type (fixed, displaced).

Similarly, forces are defined by using "SturcturalBoundaryLoad" function and name-number pair. Here the forces can be vectorized so that the magnitudes in different direction (X, Y and Z) can be given.

©IJRASET: All Rights are Reserved

D. Mesh

Meshing is an important aspect in the FEA model. The size and quality of the mesh element highly govern the results of the simulation. Typically, smaller size element gives better results but at the cost of computing time, so it is basically a trade-off between these two. Also, is not advisable to have very fine mesh if a coarser mesh gives almost the same result.

In MATLAB, PDE model generates triangular mesh for 2D objects and tetrahedral mesh for 3D objects. The mesh sizing can be changed by using "H_{max}" and "H_{min}" arguments. The growth of the mesh can also be varied by using "H_{grad}". A very fine mesh is possible but as we know decreasing the size of the element will require a higher computing power. However, in MATLAB we cannot vary the mesh size locally for more critical areas.

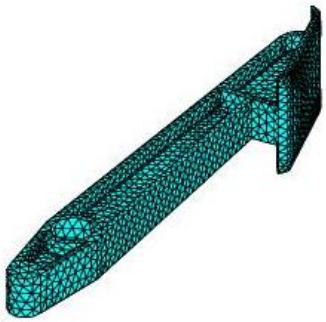


Fig. 4. Tetrahedral mesh of max 0.005 m and min 0.003 m size with a growth gradient of 1.6 in MATLAB

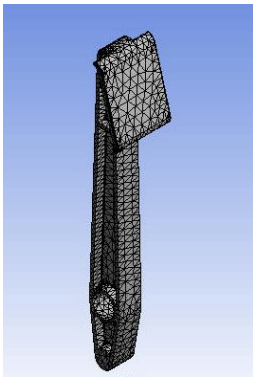


Fig. 5. Tetrahedral mesh of size 0.0035m and fast transition in Ansys



Fig. 6. Tetrahedral mesh in Solidworks

E. Solve

After providing the model with the pre-processing data next step is to solve the model. The PDE toolbox solves the mathematical model, the function used is "solve" and the model is passed as the argument to the function

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue I Jan 2022- Available at www.ijraset.com

IV. RESULTS

The results of displacement and stress were plotted on the 3D object itself to understand better the critical area in the model. The stress and displacement plots from MATLAB are show in Fig. 7. And Fig. 8 respectively.

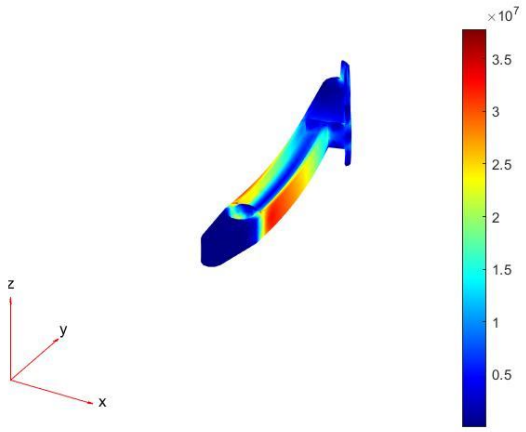


Fig. 7. Stress plot in MATLAB

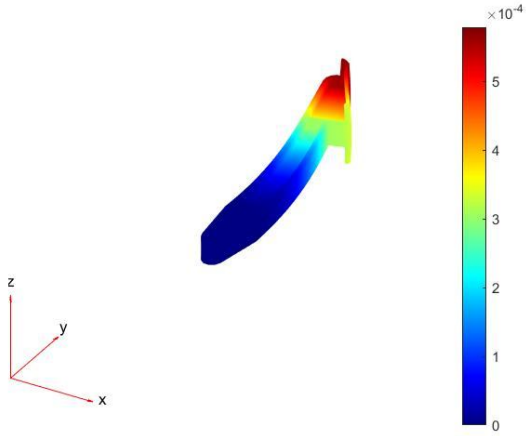


Fig. 8. Displacement plot in MATLAB

V. APPLICATION DEVELOPMENT

After completing with the basic code, the focus shifted to make the model more user friendly and easy to use. For this, MATLAB provides an option for development of an application. Using similar concepts of PDE toolbox of MATLAB, a backend code was written for an app-based model. The basic layout for the application is shown in Fig. 9.

Doing this makes steps like changing the shape of mesh, adding constraints and loads very easy. For the same steps the code has to be rewritten by changing the variables in the syntaxes whereas here we could just input number in textboxes and press button for implementing the command. Steps for using the model are more or less the same and can be interchanged.

Following steps can be followed in using the application: -

1) Select the file (.stl format)
2) Upload material properties
3) Input the mesh size and rate of its growth
4) Add boundary condition and loads
5) Solve

©IJRASET: All Rights are Reserved

Pressing the solve button stress plot and displacement plot window pops up. After the analysis is completed press “Close”, this closes all the window and the application.

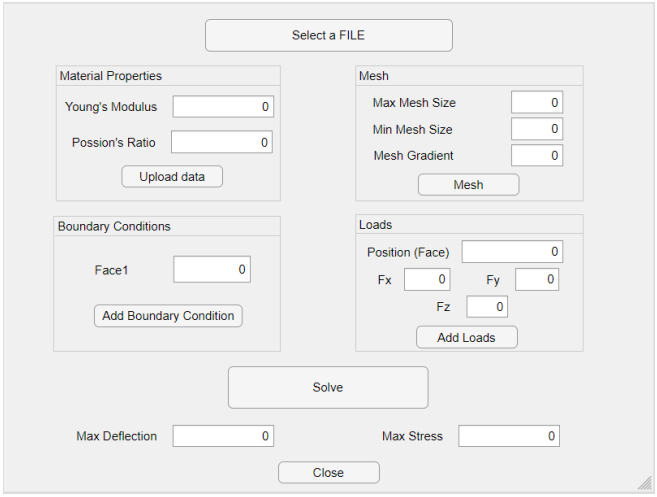


Fig. 9. App layout

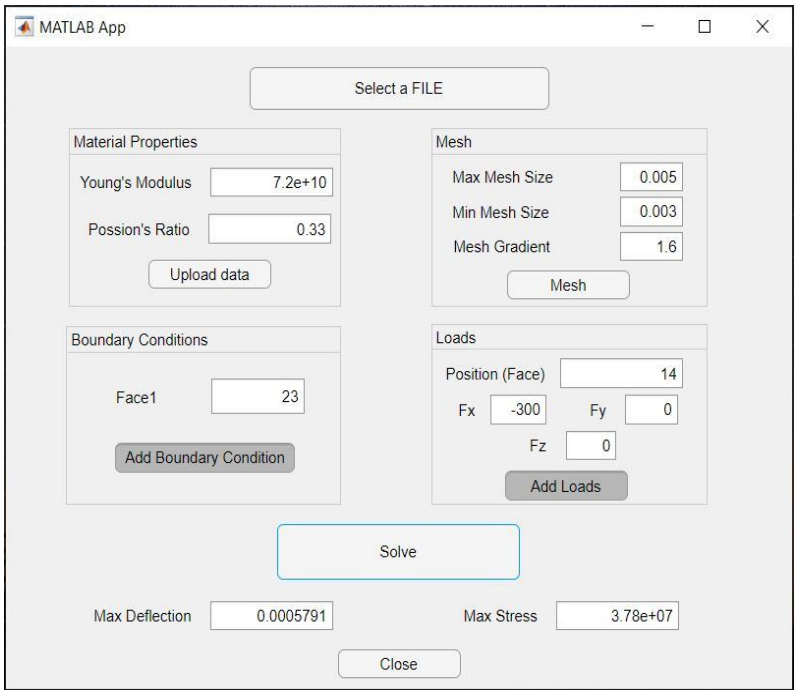


Fig. 10. App results

VI. CONCLUSIONS

A simple yet effective method for FE analysis was implemented using MATLAB and the results were compared with Ansys and Solidworks. The following table shows the results obtained from these softwares.

TABLE I
COMPARISON OF RESULTS

Parameter	Software		
	MATLAB	Ansys	Solidworks
Displacement	0.0005791	0.00055149	0.000493
Stress	3.780e7	5.739e7	3.267e7

The results from different software were closely matching.

MATLAB SCRIPT

```
model = createpde('structural',"static-solid")
%Creating PDE model
[filename, pathname] = uigetfile('*.stl','Select the File')
importGeometry(model,[pathname filename])
%Importing geometry
pdegplot(model,"FaceLabels",'on')
title([filename])
%Opening the geometry
structuralProperties(model,'YoungsModulus',71.7e9,'PoissonsRatio',0.33)
%Uploading Scructural Properties
structuralBC(model,'Vectorized',"on",'Face',23,'Constraint','fixed')
%Adding Boundary Conditions
structuralBoundaryLoad(model,'Face',14,'SurfaceTraction',[0;0;-300])
%Loading Condition
generateMesh(model,'GeometricOrder','quadratic','Hmax',0.005,'Hmin',0.003,'Hgrad',1.6)
pdeplot3D(model)
title('Mesh Model')
%Applying Mesh
result = solve(model)
%Sloving the Model
pdeplot3D(model,'ColorMapData',result.Displacement.Magnitude,'Deformation',result.Displacement)
title('Deformation Plot')
maxdispalcement = max(result.Displacement.Magnitude)
%Plotting the result of Displacement
pdeplot3D(model,'ColorMapData',result.VonMisesStress,'Deformation',result.Displacement)
title('Stress Plot')
maxstress = max(result.VonMisesStress)
%Plotting the result of Stress
```

VII. FUTURE SCOPE

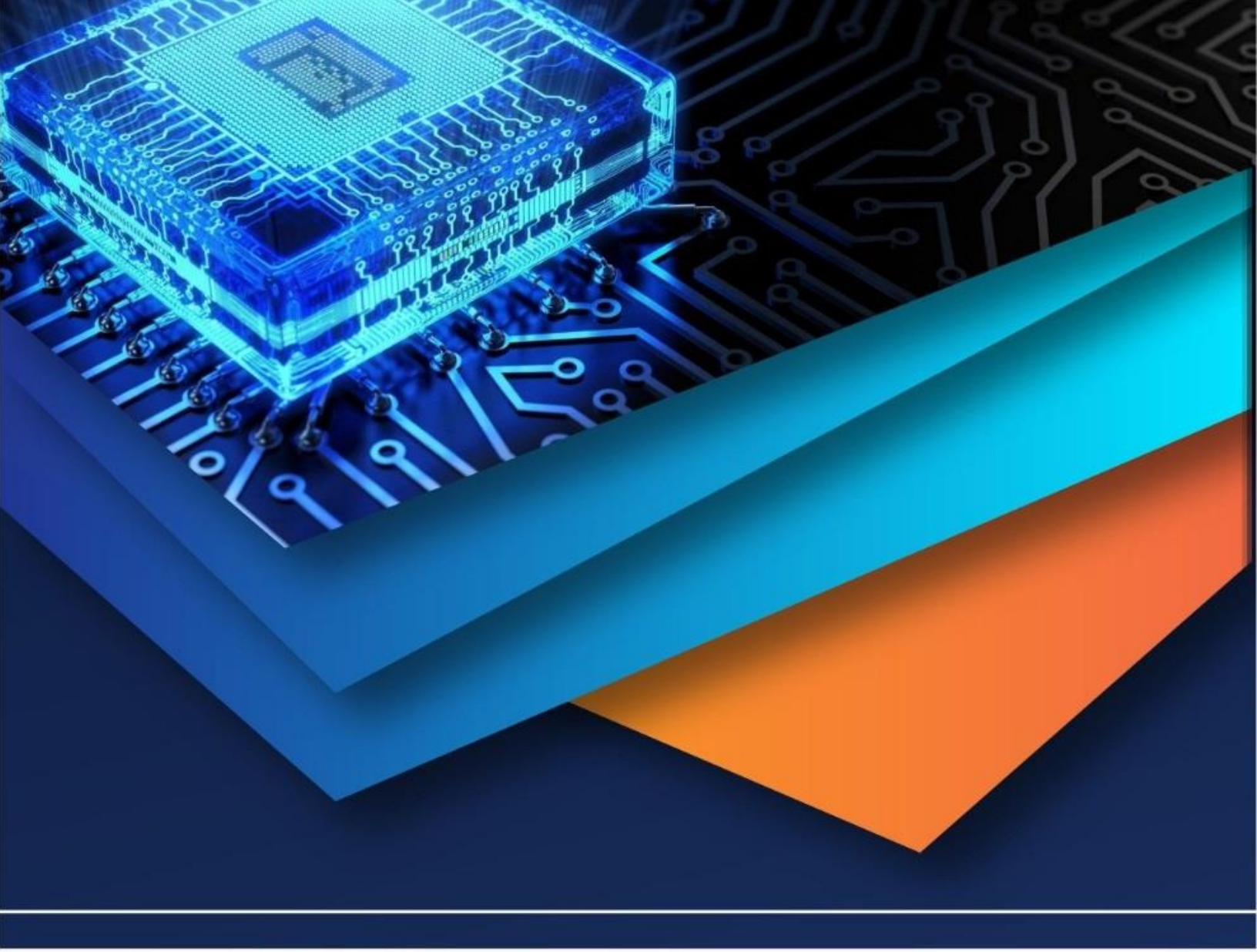
MATLAB is definitely a powerful tool for doing FE analysis which is demonstrated in the preceding note; hence it can be used for doing more complex simulation. We can add various different types of constraints and loads on faces or edges thus making the model more sophisticated. Further, a graph can be plotted in the application interface itself which can be changed between displacement or stress or strain plots using toggle switch.

VIII. ACKNOWLEDGMENT

Rahul Yadav thanks Prof. S. Rajanarasimha of K J Somaiya College of Engineering, Mumbai who motivated to carry the project and gave his valuable suggestions time and again which was very essential to make the project a success.

REFERENCES

[1] J. N. Reddy, “Finite Element Method” Tata McGraw Hill, 2003.
[2] Chandrupatla and Belegundu, “Introduction to Finite Elements in Engineering” PHI / Pearson Education, 2003.
[3] MATLAB Codes for Finite Element Analysis by A.J.M. Ferreira
[4] MATLAB documentation on Finite Element Analysis, (https://in.mathworks.com/discovery/finite-element-analysis.html)



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)