Technische Universiteit
Eindhoven
University of Technology

4EM30

Scientific computing for mechanical engineering

Final Assignment Replacement

Author:	Student nr.:	E-mail:
F.J. Ramp	0663263	f.j.ramp@student.tue.nl
R.J. Arnold Bik	0677604	r.j.arnold.bik@student.tue.nl
M.J.C. Schapendonk	0735013	m.j.c.schapendonk@student.tue.nl

June 9, 2016

Contents

Chapter 1

Overview

1.1 The program

A short overview of the program's components.

1.1.1 Mesh

The mesh that is used for this FEM analysis is made using Gmsh. Gmsh creates a .msh file. The .msh file consists of a list of nodes with their coordinates, and a list of elements with their nodes. The file IOlib.py is used to convert these lists to our data structures. This file also constrains the left side of the mesh.

1.1.2 Other input

In main.py it is possible to adjust the material properties (Young's modulus and Poisson's ratio). The .msh file has to be specified here, as is the force that will be exerted on the boundary.

1.1.3 Creating the system of equations

First, the force has to be distributed over the edge of the mesh. This is done using the function Distribute_Force. Now the vector F can be made using the function getF. Subsequently stiffness matrix K can be constructed using the function getK.

1.1.4 Solving the system

The system is solved using the function solveSys. This function first removes the parts with a prescribed displacement (constraint) from the left hand side and moves them to the right hand side. This modified stiffness matrix is now made into a sparse matrix to decrease the solving time. After this system is solved, the vector U of the displacements is reassembled and returned.

1.1.5 Stresses

Finally the stresses in each element are calculated using the displacements of the nodes. This is done using the function get_FEM_stresses. This function uses single point Gaussian quadrature.

Chapter 2

Validation

2.1   Analytical solution

Appendices

Appendix A

Initial velocity

A.1 blabla