Rensselaer Polytechnic Institute

MANE 4240: Introduction to Finite Elements

Fall 2020

Milestone 3 Report

Solid Mechanics – Deformation of Plates

Finite Element Analysis Graduate Study Progression to Examine Rupture Discs

Ian Navin

Professor Ernesto Gutierrez-Miravete

12.6.20

Table of Contents

List of Symbols

a	Location on Beam/Plate
a_{mn}	Coefficient, Navier, Displacement
b	Location on Beam/Plate
BCA	Boundary Condition at a
BCB	Boundary Condition at b
c	Height on Y axis
C	Integration Constant
D	Flexural Rigidity
δ	Deflection
E	Modulus of Elasticity
Σ	Summation
G	Shear Modulus
h	Height of Beam
I	Moment of Inertia
L	Length of Beam
M	Moment
m	Integer (Navier, x)
n	Integer (Navier, y)
ν	Poissons Ratio
p_{mn}	Coefficient, Navier, Load
Φ	Global Function
π	Pi
q	Beam Load (Distributed)
ρ	Density
σ	Normal Stress
τ	Shear Stress
u	Deflection (x direction)
u’	Derivative of Deflection

v	Test Function Variable/Deflection (y)
v’	Derivative of Test Function
x	Horizontal Direction Variable

List of Figures

Introduction/Assumptions/Background

In the attempt to study, create, develop, test, verify, validate, and certify a finite element model of a realistic engineering system, I chose to focus on the deformation of plates. In my field of work, I will be developing, designing, and analyzing rupture discs. Rupture discs are a pressure relief safety component that is usually used to protect crucial systems from over-pressurization. Some special cases allow fluids to be released for safety critical landing systems. The component consists of a body and a thin metal disc or membrane that is sacrificed at a rated pressure. When the rupture disc is installed in the customers system, one side of the disc is subject to positive pressure while the other side sees vacuum pressure. The rupture disc is rated to burst at a pressure that depends on the pressure ratings of components in the system. Using finite elements to study plate deformation will further my understanding of the behavior of a plate or membrane during the mode of deformation as pressure builds on the disc.

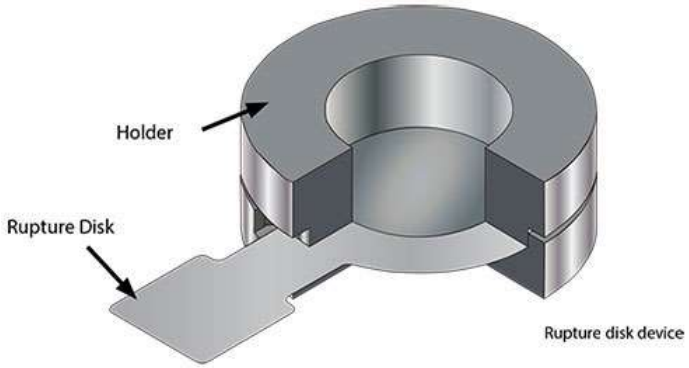


Figure 1: General Rupture Disc Components (Before Burst)

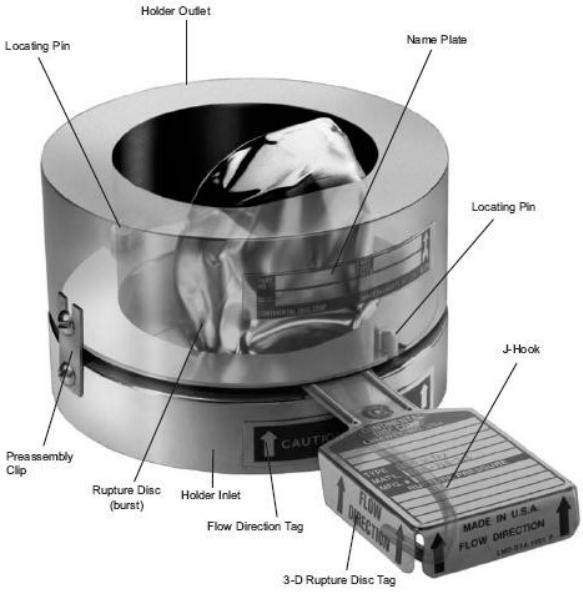


Figure 2: Rupture Disc Components (After Burst)

Rupture discs and plates come in many forms. Some are assembled using two housings that clamp the disc/plate around the edges and allows for the disc to be replace upon rupture. Others are fastened using alloying of the disc with braze and the housing.

The mode of bursting can prove to be quite unpredictable unless controlled tightly. The disc will typically burst where the most stress and strain are present, but the way it bursts can happen in a few ways. Ideally, the disc opens by flowering in the center, or peels open like a soda can top. Many times, depending on the thickness of the plate and the rated pressures, the disc can just crack, restricting the flow of the system and not fully relieving the pressure or the material can shed into the customers system. This foreign object debris is unwelcome and occurs a lot with disruptions in the materials microstructure. One way that shedding has been mitigated is with a scored or stamped disc as shown below. This creates a thinner section in the center of the disc/plate where the material discontinuity is mechanically controlled. This allows the rupture of the disc to be better controlled and less susceptible to rupturing or forming a restrictive crack. This method of scoring/stamping is to be studied and analyzed to be proved out using the finite element method.

Figure 3: Scored Burst Disc

Figure 4: Controlled Rupture "Flower" from Scored/Stamped Disc

When studying plates, we consider the plate to be a flat structural component whose thickness is very small in comparison to its' length or width. The thickness of the plate can grow to a point where plate theory is no longer considered useful (plane stress) and a three-dimensional structure such as a beam would better describe the element. Some assumptions that are made from a classic plate theory include [10]:

1.) Deflection of the midplane is small compared to the thickness
2.) The midplane remains unstrained subsequent to bending
3.) Plane shear strains are zero. Out of plane strain is omitted
4.) Out of plane normal stress is neglected

For purposes of basic understanding and a preliminary approach, we will consider beam/plate deflection since they are very closely related in elementary theory of Strength of Materials. To simplify and build on the study of a beam or plate, we will start with a beam, simply supported on both ends with a transverse load as shown below.

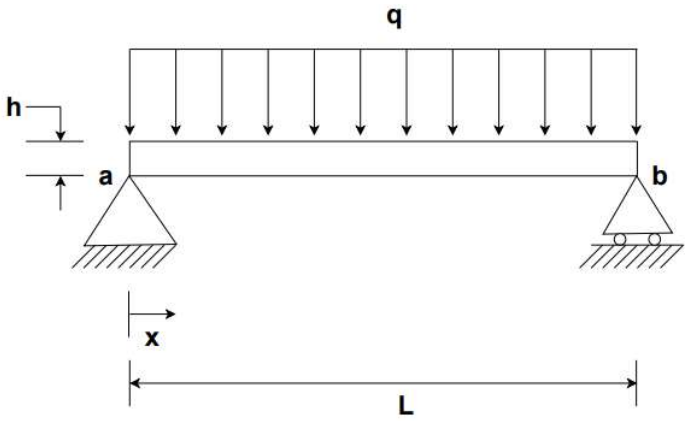


Figure 5: Simply Supported Beam

To explore a similar model with differing boundary conditions and show that we fully understand the development of the variational formulation we will consider a cantilever beam with an equivalent transverse load as shown below.

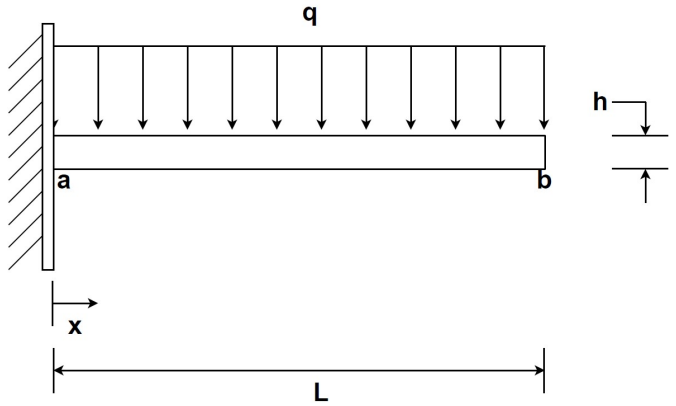


Figure 6: Cantilever Beam

After exploration of the Euler-Bernoulli equivalent simply supported and cantilever beam, we will move to a deeper understanding of beam and plate theory by exploring Timoshenko's approach which more accurately depicts the behavior of beams by considering shear deformation.

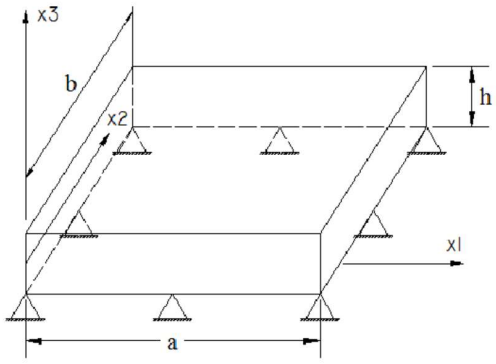


Figure 7: Simply Supported Plate

For clarity of variables involved in the basic beam problems, the length of the beam is L, the height is h, the breadth is b, the load acting on the beam is q, M is the bending moment, u is the deflection, E is the modulus of elasticity, I is the moment of inertial, EI is the flexural rigidity of the beam, and a and b are points at the ends of the beam. When looking at the Timoshenko models, h is the thickness, a is the width of the plate, and b is the depth of the plate.

To fully analyze the beams, we will start with basic equations that describe the beams including the bending moment and the differential equation for the deflection of the beams. From the differential equation, we will derive the variational formulation for the problem. This variational form will allow us to study numerical methods to find an approximate solution to the problem using Galerkin Approximation and compare the approximation to the exact solution.

Through this familiarity of the Galerkin Approximation, we can further our understanding of the use of finite elements by breaking the model into multiple elements, made up of nodes. In a basic sense, the model will be one element with two nodes defined at the ends by the boundary conditions. We can further break the model into more elements and study how this affects the approximation. Utilization of Lagrange Polynomials to create local basis functions at the nodes and across the elements will allow the development of Global basis functions that can be used as linear piecewise test functions in the Galerkin approximation. Using piecewise functions in the test functions will be more realistic approach to the Galerkin approximation since these are easily constructed using the Lagrange polynomials and the approach can translate to all finite element studies using the Galerkin approximation.

Utilization of FEA software packages will allow the comparison of the coding in the Galerkin approximation. These packages use the same type of methods and coding as we are deriving, so the contrast will create much needed insight into how these types of software work. If we can find

that the solutions are nearly the same, then we can trust these more powerful tools to conduct model of higher dimensions.

Expanding from the 1D elements, we can break up the beams into two dimensional elements that essentially form the mesh in which to evaluate the beams. These elements will be made up of nodes that will scale the x and y axis. We can explore mesh refinement to see how the approximation accuracy compared to the exact solution will improve.

After exploration of the Euler-Bernoulli equivalent simply supported and cantilever beam, we will move to a deeper understanding of beam and plate theory by exploring Timoshenko's approach much more accurately depicts the behavior of beams by considering shear deformation. We will also look at meshing and study the effects of considering plane strain vs plane stress in our approximation. These results will be compared to Timoshenko's plate theory exact solutions.

Later, a three-dimensional plate model, where we will compare results to the 2D and 3D simply supported plates will be evaluated and furthermore clamped and scored 3D plate models will be explored using finite elements in COMSOL. These models will be more realistic to real world applications with consideration to materials that the field uses. In our case we will consider Inconel 625. Inconel 625 is used in many of our customers system because it lends itself well to environments that require high strength, high temperature, and corrosion resistance. The material properties of this material will be considered at annealed conditions. This is because to assemble these parts on the production floor, the disc is brazed to the housing. This process anneals the disc. We will also consider the disc properties at room temperature since this is the temperature in which we rate the burst pressure. The Modulus of Elasticity of Annealed Inconel 625 at room temperature is 8.14E10 Pascals, the tensile strength is 8.27E8 – 10.34E8 Pascals, the yield strength (0.2% offset) is 41.4E8 – 65.5E8 Pascals, the density is 8440 lb/ft^3, and poisons ratio is 0.278 [11].

Analysis

Derivation of Preliminary Equations of Beam/Plate Deformation and Exact Solution for Simply Supported Beam Deflection

As discussed in the introduction, we first need to describe the governing equation which is given by

$$EI\frac{d^4u}{dx^4} = q(x) \tag{1.0}$$

We can also describe the boundary conditions at a and b (or at zero and length L) as

$$u(a) = 0\ and\ u(b) = 0\ or\ BCA = u(0) = 0\ and\ BCB = u(L) = 0 \tag{1.1}$$

And we know that the second derivative of the deflection is the moment

$$M = \frac{d^2u}{dx^2}\Big|_{x=0,L} = 0 \tag{1.2}$$

Integrating the governing equation (Eq. 1.0) with respect to x, four times, will yield the general solution of the deflection as derived below

$$\iiint\int EI\frac{d^4u}{dx^4}dx = q(x) \tag{1.3}$$

$$\iiint\frac{d^3u}{dx^3}dx = \frac{qx}{EI} + C_1 \tag{1.4}$$

$$\iint\frac{d^2u}{dx^2}dx = \frac{qx^2}{2EI} + C_1x + C_2 \tag{1.5}$$

$$\int\frac{du}{dx}dx = \frac{qx^3}{6EI} + \frac{1}{2}C_1x^2 + C_2x + C_3 \tag{1.6}$$

$$u(x) = \frac{qx^4}{24EI} + \frac{1}{6}C_1x^3 + \frac{1}{2}C_2x^2 + C_3x + C_4 \tag{1.7}$$

Substitution of the boundary conditions into the above equations allow us to solve for C_1, C_2, C_3, and C_4.

$$C_1 = -\frac{qL}{2} \tag{1.8}$$

$$C_2 = 0 = C_4 \tag{1.9}$$

$$C_3 = \frac{qL^3}{12} \tag{1.10}$$

Introducing the constants into the general solution yields the exact solution for the displacement

$$u(x) = \frac{-q(\frac{1}{6}Lx^3 - \frac{1}{12}x^4)}{2EI} + \frac{qL^3x}{24EI} \tag{1.11}$$

From the exact solution for the displacement equation, we can obtain the bending moment equation of a beam from elementary Strength of Materials [2],[9] by differentiating the displacement equation two times (can be seen in Eq. 1.5).

$$M(x) = \frac{qLx}{2} - \frac{qx^2}{2} \tag{1.12}$$

We also know that our differential equation for the deflection u(x) of the beam is

$$-EI\frac{d^2u}{dx^2} = M \tag{1.13}$$

The exact solution for the displacement of the beam can be found using the differential equation for the deflection of the beam (Eq. 1.13), the bending moment equation (Eq. 1.12), and our boundary conditions (Eq. 1.1 and 1.2). These can be defined in Maple and the exact solution can be solved for. The coding and calculations in Maple can be seen below

```
> restart;
```

Navin_Ian MANE 4240 Milestone 1, Exact Solution

Bending Moment Equation of a beam from Strength of Materials

```
> M := (q·L·x)/2 - (q·x^2)/2;
```

$$M := \frac{1}{2} q L x - \frac{1}{2} q x^2 \tag{1}$$

Differential Equation for the deflection u(x) of the beam

```
> de1 := -E·II·(d^2/dx^2)(u(x)) = M;
```

$$de1 := -E\,II\left(\frac{d^2}{dx^2} u(x)\right) = \frac{1}{2} q L x - \frac{1}{2} q x^2 \tag{2}$$

Boundary Conditions for a Simply Supported Beam

```
> BCA := u(0) = 0; BCB := u(L) = 0;
```

$$BCA := u(0) = 0$$
$$BCB := u(L) = 0 \tag{3}$$

Exact Solution

```
> S1 := dsolve({de1, BCA, BCB}, u(x));
```

$$S1 := u(x) = -\frac{q\left(\frac{1}{6} L x^3 - \frac{1}{12} x^4\right)}{2\,E\,II} + \frac{q L^3 x}{24\,E\,II} \tag{4}$$

Figure 8: Maple Coding of Exact Solution of Simply Supported Beam Deflection

We can see that the exact solution obtained in Maple is the same solution obtained in Equation 1.11. This creates confidence that our derivations, assumptions, and boundary conditions were correct.

If values are assigned to the variables, we can obtain a plot of the deflection of the beam from end to end. We will define the variables as follows: E=2E11, b=0.1, h=0.1, I=(bh^3)/12, L=1, and q= -1E6. The plot of the exact solution deflection from x=0 to x=L, as obtained from Maple, is shown below. We can observe that the plot of the deflection from end to end is symmetric.

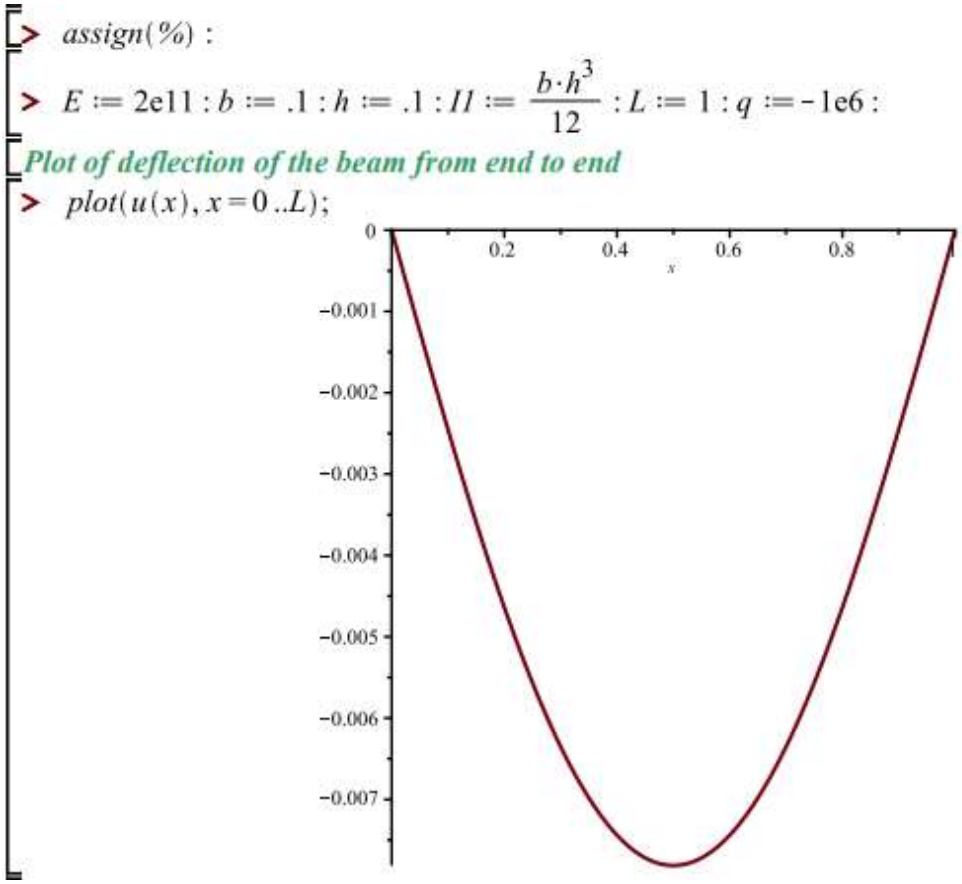


Figure 9: Maple Coding of Exact Solution SSB Plot and Variable Assignment

Derivation of Preliminary Equations of Beam/Plate Deformation and Exact Solution for Cantilever Beam Deflection

Again, we describe the governing equation which is given by Eq. 1.0.

The boundary conditions at a and b will differ from the simply supported beam as shown below

$$u(a) = 0, u'(a) = 0, u''(b) = 0 \; and \; u'''(b) = 0 \qquad (1.14)$$

We can integrate the governing equation again as we did with the simply supported beam and solve for the constants of integration using the boundary conditions. This will yield the exact solution for deflection of a cantilever beam as shown below.

$$u(x) = -\frac{qx^2}{24EI}(6L^2 - 4Lx + x^2) \qquad (1.15)$$

From the exact solution for the displacement equation, we can obtain the bending moment equation of a beam from elementary Strength of Materials [2],[9] by differentiating the displacement equation two times (can be seen in Eq. 1.5).

$$M(x) = -\frac{q(L-x^2)}{2} \tag{1.16}$$

We can see how much the deflection and moment equations differ due to the differing boundary conditions. This allows us to explore the behavior of these plates/beams when constrained differently.

The exact solution for the displacement of the cantilever beam can be found using the differential equation for the deflection of the beam (Eq. 1.15), the bending moment equation (Eq. 1.16), and our boundary conditions (Eq. 1.14). These can be defined in Maple and the exact solution can be solved for. The coding and calculations in Maple can be seen below

```
> restart;
Bending Moment Equation of a beam from Strength of Materials
> M := -q·(L-x)^2/2;
```

$$M := -\frac{q\,(L-x)^2}{2}$$

```
Differential Equation for the deflection uex(x) of the beam
> de1 := -E·I1·(d^2/dx^2)(uex(x)) = M;
```

$$de1 := -E\,I1\left(\frac{d^2}{dx^2}\,uex(x)\right) = -\frac{q\,(L-x)^2}{2}$$

```
Boundary Condition definition
> u0 := 0; s0 := (q·L^3/(6·E·I1));
```

$$u0 := 0$$

$$s0 := \frac{q\,L^3}{6\,E\,I1}$$

```
> ss0 := -(q·x/(6·E·I1))·(3·L^2 - 3·L·x + x^2);
```

$$ss0 := -\frac{q\,x\,(3\,L^2 - 3\,L\,x + x^2)}{6\,E\,I1}$$

```
> BCA := uex(0) = u0; BCB := D(uex)(0) = 0;
```

$$BCA := uex(0) = 0$$

$$BCB := D(uex)(0) = 0$$

```
Exact Solution
> dsolve({de1, BCA, BCB}, uex(x));
```

$$uex(x) = \frac{q\,(L-x)^4}{24\,E\,I1} + \frac{q\,L^3\,x}{6\,E\,I1} - \frac{q\,L^4}{24\,E\,I1}$$

Figure 10: Maple Coding of Exact Solution of Cantilever Beam Deflection

We can see that the exact solution obtained in Maple is the same solution obtained in Equation 1.11. This creates confidence that our derivations, assumptions, and boundary conditions were correct.

If the same values are assigned to the variables, we can obtain a plot of the deflection of the beam from end to end. The plot of the exact solution deflection from x=0 to x=L, as obtained from Maple, is shown below.

Plot of Beam Deflection from x=0 to x=L

```
> assign(%) :
> q := -1e6 : L := 1 : E := 2e11 : h := .1 : b := .1 : II := b·h^3/12 :
> plot(uex(x), x = 0..L);
```

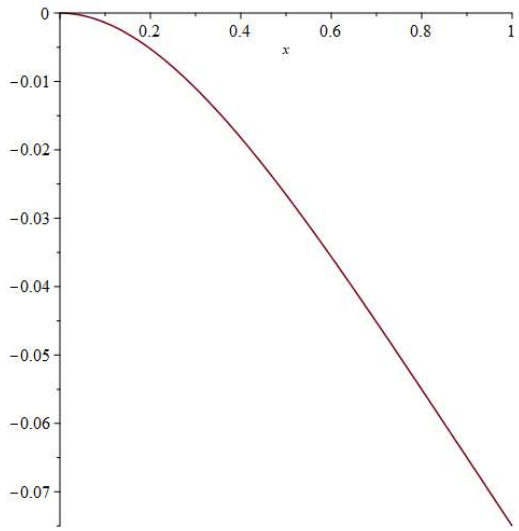


Figure 11: Maple Coding of Exact Cantilever Solution Plot and Variable Assignment

Derivation of the Variational Formulation

Derivation of the Variational Formulation first requires the definition of the differential equation and the boundary conditions. The combination of these two factors create what is known as the differential formulation of a Boundary Value Problem (BVP) meaning that the problem is fully defined by its boundaries and differential equation [4], [5]. The Variational Formulation is what the name implies; a variation of the differential formulation.

To obtain the variational formulation, a test function is introduced. Though this test function is not precisely defined, it has stipulations in order to be used. First, the boundary points of this function must be equivalent to the boundary values for the deflection, $u(x)$. The second condition is that the function must be piecewise differentiable at least once [4]. A piecewise function is a function that is made up of different functions and values at different intervals. This can be seen in a step function where the function looks like stairs. The step function equals a different constant value at the varying intervals (i.e. $f(x) = 2\ at\ 0 < x \leq 1,\ f(x) = 3\ at\ 1 < x \leq 2$). Piecewise differentiable therefore means that each function at the varying intervals is differentiable, though collectively, the function might not be differentiable at intervals not covered by the piecewise function.

The test function that we will introduce will be $v(x)$. We know our differential equation for the beam deflection from Eq. 1.13 and the boundary conditions from Eq. 1.1. We will multiply the test function on both sides of our differential equation and then integrate over our boundaries which will eventually utilize integration by parts. The derivation is as follows

$$v(x) * -EI\frac{d^2u}{dx} = M * v(x) \tag{1.14}$$

$$-EI\frac{d^2u}{dx}v(x) = Mv(x) \tag{1.15}$$

$$\int_a^b -EI\frac{d^2u}{dx}vdx = \int_a^b Mvdx \tag{1.16}$$

As a side note, we will define $u(x)$, $v(x)$, and degrees of derivatives of each for ease of computation

$$u = u(x),\ \frac{du}{dx} = u',\ \frac{du'}{dx} = u'',\ \ du = u'dx,\ \ du' = u''dx \tag{1.17}$$

$$v = v(x),\ \frac{dv}{dx} = v',\ \ dv = v'dx \tag{1.18}$$

Continuing

$$EI\int_a^b -vdu' = \int_a^b Mvdx \tag{1.19}$$

Taking the integral of -vdu' separately (using integration by parts)

$$\int_a^b -vdu' = \int_a^b -u''vdx = \left[-[u'v]_a^b - \int_a^b v'u'dx\right] \tag{1.20}$$

$$\int_a^b -vdu' = \int_a^b -u''vdx = -[u'v]_a^b + \int_a^b v'u'dx \tag{1.21}$$

Defining $-[u'v]_a^b$

$$-[u'v]_a^b = -[u'(b)v(b) - u'(a)v(a)] = -u'(b)v(b) + u'(a)v(a) = u'(a)v(a) - u'(b)v(b) = [u'v]_b^a \tag{1.22}$$

$$-[u'v]_a^b = [u'v]_b^a \tag{1.23}$$

Plugging back into Eq. 1.19

$$\int_a^b EIu'v'dx + [u'v]_b^a = \int_a^b Mvdx \tag{1.24}$$

Inputting Boundary Conditions for the simply supported beam, $u(a) = v(a) = u(b) = v(b) = 0$ yields the final variational formulation

$$\int_a^b EIu'v'dx = \int_a^b Mvdx \tag{1.25}$$

Inputting the boundary conditions for the cantilever beam and knowing that the slope u'(b) is the max deflection of the cantilever

$$u'(b) = \frac{qL^3}{6EI} \tag{1.26}$$

We can plug in our boundary conditions and known values of the slope at the end of the beam into Equation 1.24 and obtain the variational formulation of the cantilever beam

$$\int_a^b EI(u'v'dx - u'(b)v(b)) = \int_a^b Mvdx \tag{1.27}$$

Initial Galerkin Approximation with Known Simple Function (Simply Supported Beam)

In structures, the structure is discretized into elements represented by nodes. The stiffness properties of each local element are represented in a matrix equation that describes the structures behavior and can therefore solve for the displacements and forces in the structure locally and then globally by assembling all the local elements. The Galerkin method is a sub-method of the method of weighted residuals to obtain the stiffness matrix and therefore obtain our approximate/numerical solution, in our case displacements, to the structure at hand. [8]

The method utilizes the variational formulation and its test function of the differential formulation, defined global functions, and undefined parameters in the form of linear combinations [3]. Solving for the undefined parameters will allow a numerical solution to be found. Depending on the number of global parameters utilized, the numerical solution will approach or diverge from the exact solution.

In this case, we introduce five admissible, known functions that are oscillatory as the global functions. Starting with one of the parameters of the global functions will allow the comparison to the exact solution. Thereafter, a five parameter Galerkin approximation will be used to show the differences in the exact solution with varying parameters of the global functions. Knowing that sine functions are oscillatory, we use five sine functions as listed below [1]

$$\Phi_1 = \sin(\pi x) \tag{1.28}$$

$$\Phi_2 = \sin(2\pi x) \tag{1.29}$$

$$\Phi_3 = \sin(3\pi x) \tag{1.30}$$

$$\Phi_4 = \sin(4\pi x) \tag{1.31}$$

$$\Phi_5 = \sin(5\pi x) \tag{1.32}$$

Utilizing these global functions for the linear combinations in the variational formulation and integrating over the domain will yield linear algebraic equations that can be solved for the unknown variables. Once obtained, the numerical solution is found, and the plotted comparison is shown.

The one parameter Galerkin approximation as calculated in Maple is shown below

Generation of Five suitable admissible, known and simple functions (oscillatory), "od" used to close a "do" clause

```
> for i from 1 to 5 do phi[i] := sin(i·Pi·x) od;
```

$$\phi_1 := \sin(\pi x)$$

$$\phi_2 := \sin(2\pi x)$$

$$\phi_3 := \sin(3\pi x)$$

$$\phi_4 := \sin(4\pi x)$$

$$\phi_5 := \sin(5\pi x) \quad (5)$$

One Parameter Galerkin Approximation of uG1

Test Function defined as v1

Unknown constant coefficient, b1, introduced to define approximation

```
> uG1 := b1·phi[1]; v1 := phi[1]; duG1dx := diff(uG1, x); dv1dx := diff(v1, x);
```

$$uG1 := b1 \sin(\pi x)$$

$$v1 := \sin(\pi x)$$

$$duG1dx := b1\,\pi \cos(\pi x)$$

$$dv1dx := \pi \cos(\pi x) \quad (6)$$

Above defined funtions, uG1 and v1, and their derivatives input into the Variational Form of the Differential Equation

```
> eq1 := int(E·I1·duG1dx·dv1dx, x = 0..L) = int(M·v1, x = 0..L) : evalf(eq1);
```

$$8.224670340\ 10^{6}\ b1 = -64503.06887 \quad (7)$$

Solving of Linear Algebraic Equation to determine b1

```
> b1 := solve(eq1, b1);
```

$$b1 := -0.007842632738 \quad (8)$$

```
> evalf(b1);
```

$$-0.007842632738 \quad (9)$$

Plotting of uG1 and u(x) to show the Approximation of Galerkin with the Exact Solution

```
> plot([uG1, u(x)], x = 0..L);
```

Figure 12: Maple Coding of Galerkin Approximation

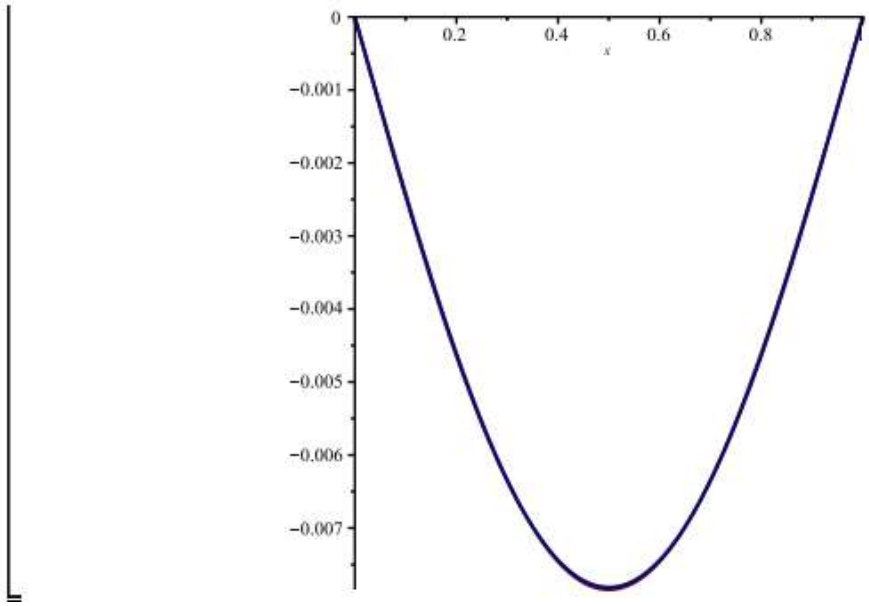


Figure 13: Maple Plot of One-Parameter Galerkin and Exact Solution

A continuation of the numerical solution is introducing the five parameter Galerkin approximation is shown below

Five Parameter Galerkin Approximation (note use of "sum" funtion)

> $uG5 := sum(a[j] \cdot \mathrm{phi}[j], j = 1..5);$

$$uG5 := a_1 \sin(\pi x) + a_2 \sin(2\pi x) + a_3 \sin(3\pi x) + a_4 \sin(4\pi x) + a_5 \sin(5\pi x) \quad \textbf{(10)}$$

> **for** *i* **from** 1 **to** 5 **do** $v[i] := \mathrm{phi}[i]$ **od**;

$$v_1 := \sin(\pi x)$$
$$v_2 := \sin(2\pi x)$$
$$v_3 := \sin(3\pi x)$$
$$v_4 := \sin(4\pi x)$$
$$v_5 := \sin(5\pi x) \quad \textbf{(11)}$$

Figure 14: Maple Coding of Five Parameter Galerkin Approximation

```
> duG5dx := diff(uG5, x);
```

$$duG5dx := a_1 \pi \cos(\pi x) + 2 a_2 \pi \cos(2 \pi x) + 3 a_3 \pi \cos(3 \pi x) + 4 a_4 \pi \cos(4 \pi x) + 5 a_5 \pi \cos(5 \pi x) \quad (12)$$

```
> for i from 1 to 5 do dvdx[i] := diff(v[i], x) od;
```

$$dvdx_1 := \pi \cos(\pi x)$$

$$dvdx_2 := 2 \pi \cos(2 \pi x)$$

$$dvdx_3 := 3 \pi \cos(3 \pi x)$$

$$dvdx_4 := 4 \pi \cos(4 \pi x)$$

$$dvdx_5 := 5 \pi \cos(5 \pi x) \quad (13)$$

Above defined funtions substituted into Variational Form of the Differential Equation to create a System of Equations

```
> for i from 1 to 5 do eq[i] := int(E·I1·duG5dx·dvdx[i], x = 0..L) = int(M·v[i], x = 0..L) od;
```

$$eq_1 := 8.224670337\ 10^6\ a_1 = -64503.06887$$

$$eq_2 := 3.289868136\ 10^7\ a_2 = 0.$$

$$eq_3 := 7.402203302\ 10^7\ a_3 = -2389.002551$$

$$eq_4 := 1.315947254\ 10^8\ a_4 = 0.$$

$$eq_5 := 2.056167584\ 10^8\ a_5 = -516.0245509 \quad (14)$$

```
> for j from 1 to 5 do evalf(eq[j]) od;
```

$$8.224670337\ 10^6\ a_1 = -64503.06887$$

$$3.289868136\ 10^7\ a_2 = 0.$$

$$7.402203302\ 10^7\ a_3 = -2389.002551$$

$$1.315947254\ 10^8\ a_4 = 0.$$

$$2.056167584\ 10^8\ a_5 = -516.0245509 \quad (15)$$

Figure 15: Maple Coding of Five Parameter Galerkin Approximation Cont.

Systems of Equations Solved for the coefficients (a1 - a5)

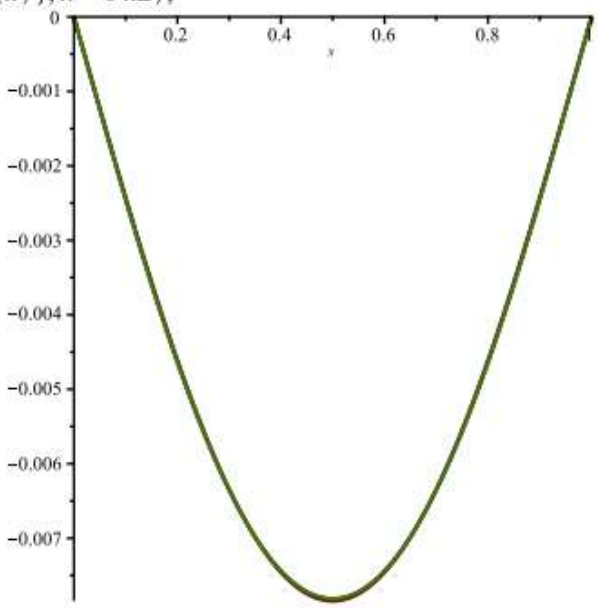
```
> solve( {eq[index]$index = 1..5}, {a[index]$index = 1..5});
```

$$\{a_1 = -0.007842632741,\ a_2 = 0.,\ a_3 = -0.00003227420882,\ a_4 = 0.,\ a_5 = -2.509642477\ 10^{-6}\} \quad (16)$$

```
> assign(%);
```

Plotting of One Parameter and Five Parameter Galerkin Approximation with the Exact Solution

```
> plot( {uG1, uG5, u(x)}, x = 0..L);
```




```
> evalf(sqrt(int((u(x) - uG1)^2, x = 0..L)));
```

$$0.00002289385377 \quad (17)$$

```
> evalf(sqrt(int((u(x) - uG5)^2, x = 0..L)));
```

$$4.158095082\ 10^{-7} \quad (18)$$

Figure 16: Maple Coding of Five Parameter Galerkin Approximation with Plot of One Parameter, Five Parameters, and Exact Solution and Errors Associated with Each

Finite Element Galerkin Approximation using Local and Global Basis Functions (Simply Supported Beam)

Now that we have a basic grasp of how the Galerkin Approximation is utilized, we can now use the Galerkin approximation to understand the finite element approach. We will still be using the variational formula of the differential form but will be using different test functions. As discussed earlier, the test function has a couple of requirements. First, the boundary points of this function must be equivalent to the boundary values for the deflection, $u(x)$. The second condition is that the function must be piecewise differentiable at least once [4]. A piecewise function is a function that is made up of different functions and values at different intervals. This can be seen in a step

function where the function looks like stairs. The step function equals a different constant value at the varying intervals (i.e. $f(x) = 2\ at\ 0 < x \leq 1,\ f(x) = 3\ at\ 1 < x \leq 2$). Piecewise differentiable therefore means that each function at the varying intervals is differentiable, though collectively, the function might not be differentiable at intervals not covered by the piecewise function. Instead of using admissible, simple and know functions for the test function, we will create piecewise functions using Lagrange Polynomials, which is a numerical interpolation method comprised of the sum of polynomials. These interpolation functions use pre-described polynomials and plug in the end values of the model.

The Lagrange polynomials will allow the local basis functions to be constructed within each element of the model. This requires a definition of the number of elements and nodes to make up the model. We will first start with the simply supported beam with a two-element model.

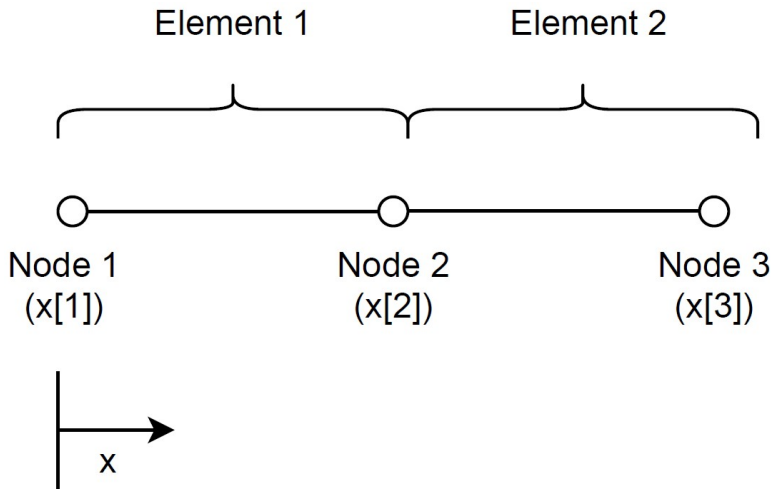


Figure 17: Two-Element Model

As we can see in the model, the node positions are defined to evenly break up the model. We need to first need to construct the polynomials that will act as the weight functions. This will be explained below

$$Polynomial\ P_a = \frac{(x-x_b)}{(x_a-x_b)}, Polynomial\ P_b = \frac{(x-x_a)}{(x_b-x_a)} \qquad (1.33)$$

When the values of the locations are input, we will get a linear polynomial. These are called weight functions because the "weight" of the function will shift depending on the location. This shift creates the convergence of the basis function and the actual function. Local finite element basis functions are obtained by generalizing the linear interpolating polynomials for an arbitrary pair of end points, which will be the end nodes of the element interval. Once the values of the node locations are implemented the basis functions can be evaluated and plotted. We can also recognize that the derivatives of these linear polynomials are constants.

For our example of the simply supported beam with a two-element model structure, we will construct the polynomials and the subsequent local basis functions using Maple programing.

$$> x[1] := 0 : x[2] := \frac{L}{2} : x[3] := L :$$

$$> p11 := \frac{(x - x[2])}{(x[1] - x[2])} : p12 := \frac{(x - x[1])}{(x[2] - x[1])} :$$

$$> p21 := \frac{(x - x[3])}{(x[2] - x[3])} : p22 := \frac{(x - x[2])}{(x[3] - x[2])} :$$

$$> plot(\{p11, p12\}, x = x[1] .. x[2]);$$

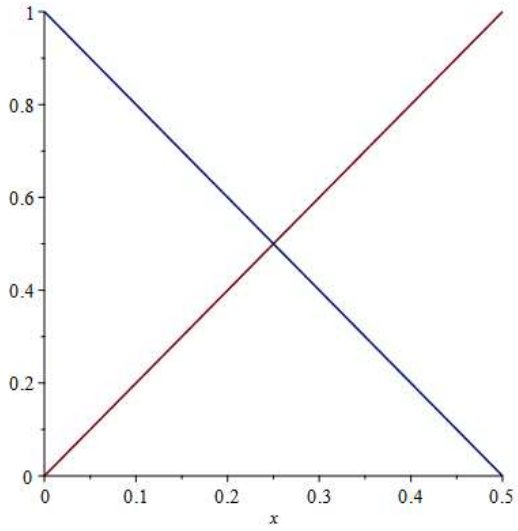


Figure 18: Plot of Local Basis Functions p11 and p12 from x=x[1] to x[2]

We can see the nodes are given represented values along the beam. X[1] represents position zero, x[2] represents the middle of the beam, and x[3] represents position L (in our case 1 on the x axis). The polynomials are constructed from the perspective of the first part of the first element (p11), then the second part of the first element (p12), the first part of the second element (p21), and lastly the second part of the second element (p22). The plot of the first element polynomials (from node 1 to node 2) can be seen above. The plot of the second element (from the second node to the third node) can be seen below. One thing to take notice of is that the plots vertical axis will always go from 0 to 1, but we do see a change in the horizontal axis because we are looking locally at the element of interest, not the whole span of the model (i.e. plot p11, p12 is from the span x[1] to x[2] or 0 to 0.5 and plot of p21, p22 is from the span x[2] to x[3] or 0.5 to 1).

```
plot({p21,p22}, x = x[2] .. x[3]);
```

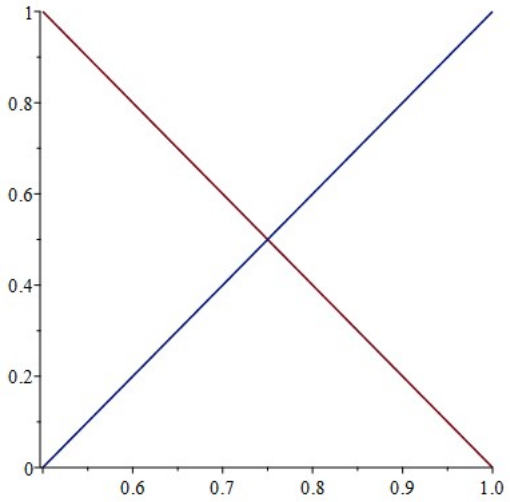


Figure 19: Plot of Local Basis Functions p21 and p22 from x=x[2] to x[3]

Global basis functions can be developed using the local basis functions by using piecewise functions. This will allow each local basis function to be represented by the full span of the model. Different types of nodes will be represented differently. The interior nodes will have basis functions running like a "witch hat" whereas the end nodes will only be one half since there will be no global basis function for the other side of the node.

We can construct the global basis functions in Maple using the "piecewise function" which will allow us to call certain local basis functions for the appropriate span of the model. For this example, we will have three global basis functions for

```
> p[1] := piecewise(x[1] ≤ x ≤ x[2], p11, x[2] ≤ x ≤ x[3], 0) :
> p[2] := piecewise(x[1] ≤ x ≤ x[2], p12, x[2] ≤ x ≤ x[3], p21, 0) :
> p[3] := piecewise(x[1] ≤ x ≤ x[2], 0, x[2] ≤ x ≤ x[3], p22) :
```

Figure 20: Maple Global Basis Function Construction (Two-Element Model)

These equations can be easily broken down into words. For global basis function p[1], it states for the span of node one to node two, use local basis function from first half of element one. For the span of node two to node three, use zero. This graph is shown below

$$p_1 := \begin{cases} -2x+1 & 0 \le x \le \frac{1}{2} \\ 0 & otherwise \end{cases}$$

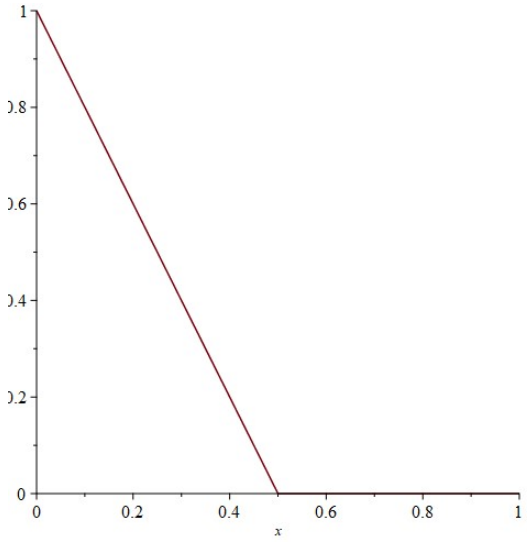


Figure 21: Maple Global Basis Function p[1] (Two-Element Model)

For global basis function p[2] (the interior node), in the span of node one and node two, use the second half of the local basis function of element one. In the span of node two and node three use first half of local basis function of element two. Otherwise, use zero.

$$p_2 := \begin{cases} 2x & 0 \le x \le \frac{1}{2} \\ -2x+2 & \frac{1}{2} \le x \le 1 \\ 0 & otherwise \end{cases}$$

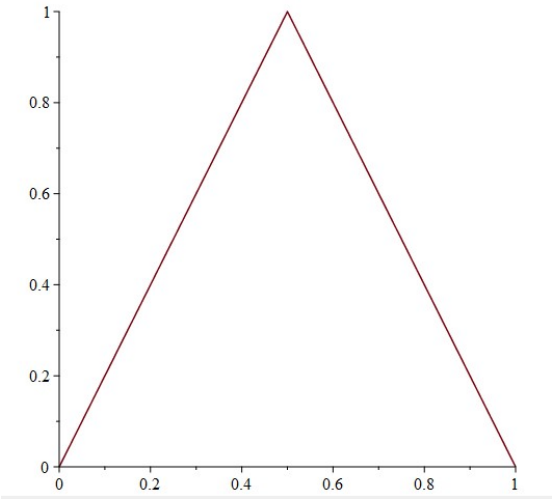


Figure 22: Maple Global Basis Function p[2] (Two-Element Model)

For global basis function p[3], the span of node one to node two, zero. For the span of node two to node three, use local basis function for node two in element two.

$$p_3 := \begin{cases} 2x - 1 & \frac{1}{2} \le x \le 1 \\ 0 & otherwise \end{cases}$$

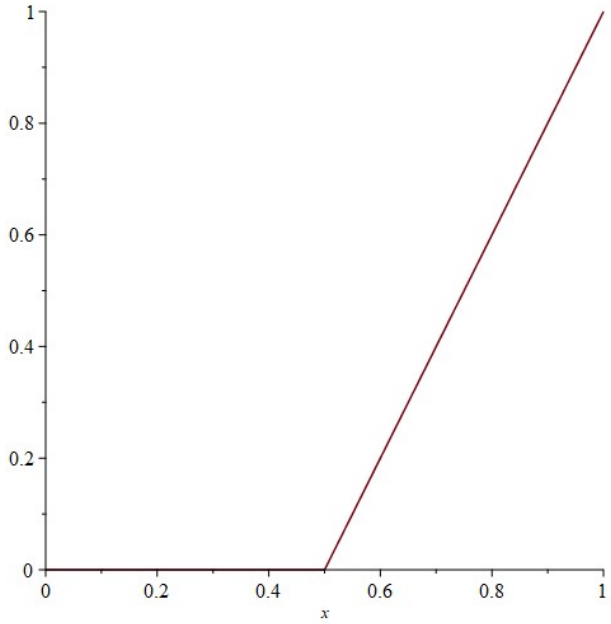


Figure 23: Maple Global Basis Function p[3] (Two-Element Model)

All these global basis functions combined will look as below

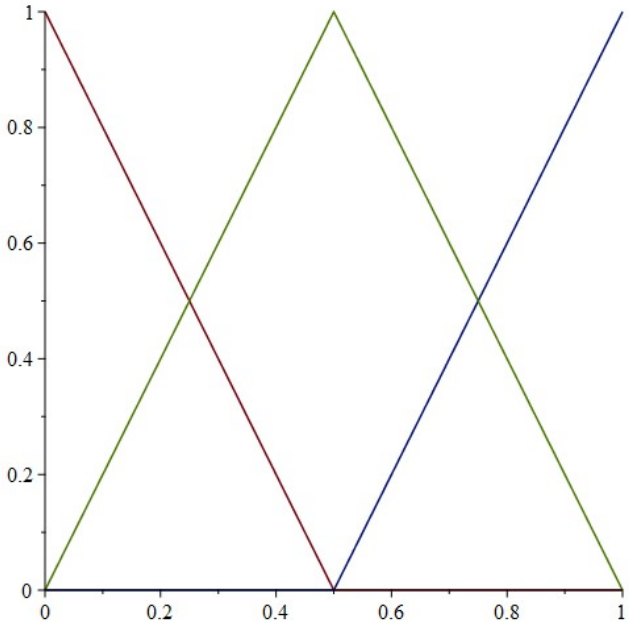


Figure 24: Maple Global Basis Functions (Two-Element) Model

We can see in red is p[1], in green is p[2] and in blue is p[3]. Now we have our global basis function, we will use these as our test functions for the Galerkin approximation. We can write the Galerkin approximation as a Linear Combination of the global finite element basis functions and its' derivative as shown below (uG).

$$> \; uG := u[1]\cdot p[1] + u[2]\cdot p[2] + u[3]\cdot p[3] : duGdx := diff(uG, x) :$$

$$uG := u_1 \left(\begin{cases} -2x+1 & 0 \le x \le \frac{1}{2} \\ 0 & otherwise \end{cases} \right) + u_2 \left(\begin{cases} 2x & 0 \le x \le \frac{1}{2} \\ -2x+2 & \frac{1}{2} \le x \le 1 \\ 0 & otherwise \end{cases} \right)$$

$$+ u_3 \left(\begin{cases} 2x-1 & \frac{1}{2} \le x \le 1 \\ 0 & otherwise \end{cases} \right)$$

$$duGdx := u_1 \left(\begin{cases} 0 & x < 0 \\ undefined & x = 0 \\ -2 & x < \frac{1}{2} \\ undefined & x = \frac{1}{2} \\ 0 & \frac{1}{2} < x \end{cases} \right) + u_2 \left(\begin{cases} 0 & x < 0 \\ undefined & x = 0 \\ 2 & x < \frac{1}{2} \\ undefined & x = \frac{1}{2} \\ -2 & x < 1 \\ undefined & x = 1 \\ 0 & 1 < x \end{cases} \right) + u_3 \left(\begin{cases} 0 & x < \frac{1}{2} \\ undefined & x = \frac{1}{2} \\ 2 & x < 1 \\ undefined & x = 1 \\ 0 & 1 < x \end{cases} \right)$$

Figure 25: Maple Galerkin Approximation Sum and its' Derivative (Two-Element Model)

We can now make the test functions identical to the global finite element basis functions instead of using a guess oscillating function like in Milestone 1. This can be seen in the Maple equation below

$$> \; dvdx[1] := diff(v[1], x) : dvdx[2] := diff(v[2], x) : dvdx[3] := diff(v[3], x) :$$

Figure 26: Maple Galerkin Approximation Equations and their Derivative (Two-Element Model)

Using the variational equation, we need to represent three equations due to the three nodes of the two-element model. At node 1 and node 3, the equation will be zero since we are at nodes that contain a zero deflection at the boundaries (Dirilecht Boundary Conditions). For equation two, it will represent node two and is as shown below as the variational equation for the simply supported beam over nodes one through three but using the second global basis function as the test function.

```
> eq1 := u[1] = 0;
                                   eq1 := u_1 = 0

> eq2 := int(E·I1·duGdx·dvdx[2], x = x[1]..x[3]) = int(M·v[2], x = x[1]..x[3]);
          eq2 := -208333.3334 u_1 + 416666.6668 u_2 - 208333.3334 u_3 = -52.08333333

> eq3 := u[3] = 0;
                                   eq3 := u_3 = 0

> solve({eq1, eq2, eq3}, {u[1], u[2], u[3]}); assign(%)
                        {u_1 = 0., u_2 = -0.0001250000000, u_3 = 0.}
```

Figure 27: Maple Discrete Variational Formulations (Two-Element Model)

After obtaining the equations, we can solve for the deflection constants using linear algebra and plot the results of the span of the beam with the exact solution for comparison below

```
> plot({uex(x), uG}, x = 0..L);
```

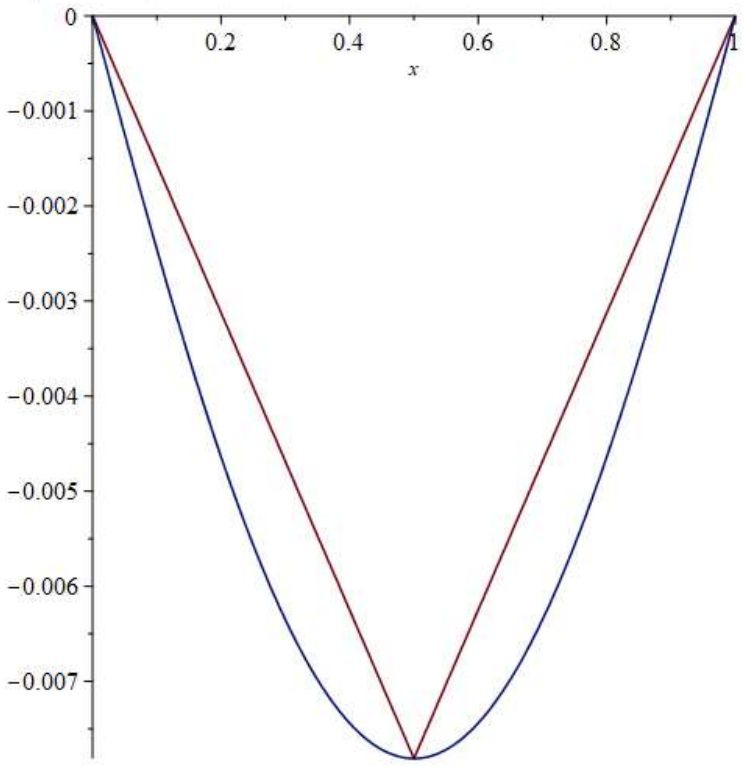
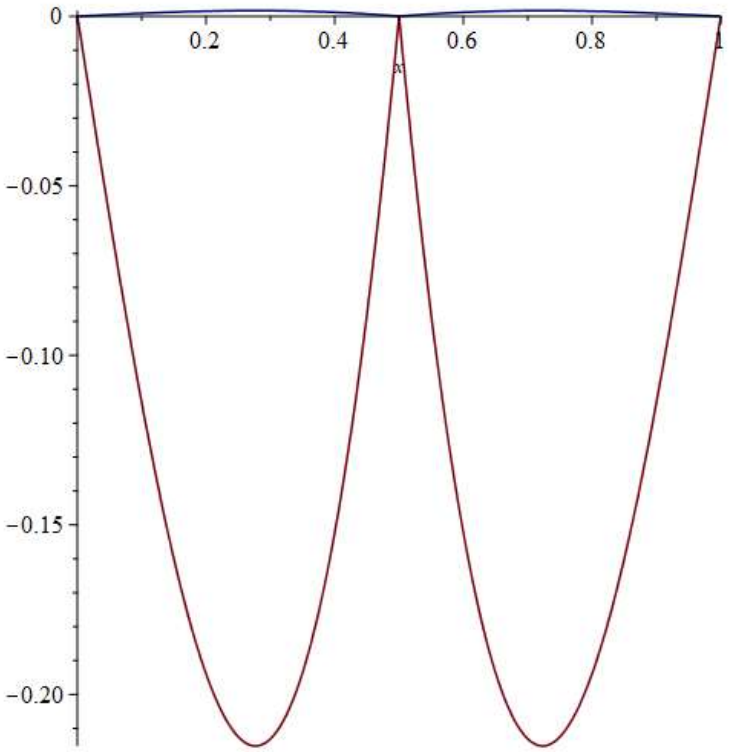


Figure 28: Exact Solution vs Two Element Model Galerkin Approximation Plot from Node 1 to Node 3

We can use Maple to compute the local and global error between the Galerkin approximation and the exact solution, which is shown below.

Computing the Local Error

```
> el := abs(uex(x) - uG) : elr := el/uexMax : plot({el, elr}, x = 0..L);
```




```
> Aex := evalf(int(uex(x), x = 0..L)); AG := evalf(int(uG, x = 0..L)); Aex - AG;
```

$$Aex := -0.004999999999$$

$$AG := -0.003906249999$$

$$-0.001093750000$$

Computing the Global Error as the Energy Norm of the Error

```
> evalf(sqrt(int((1/2)*diff(el, x)^2, x = 0..L))); evalf(sqrt(int((1/2)*diff(elr, x)^2, x = 0..L)));
```

$$0.005451183520$$

$$0.6977514906$$

Figure 29: Exact Solution vs Two Element Model Galerkin Approximation Local and Global Error

Just as we did with the admissible test functions, we will increase the number of test functions in the variational form by increasing the number of nodes. This will allow us to analyze the accuracy increase, though we will increase the computation time. A four element and eight element model can be seen below using for loop and do functions in maple to reduce the amount of equations to write. We will also use an evaluation function to evaluate the max deflection of the beam.

Galerkin FEM

N=Number of Elemens of uniform size h1

> $N := 4 : h1 := \frac{L}{N} :$

Global Finite Element Basis Functions - First Order Lagrange Polynomial

> **for** i **from** 1 **to** $N+1$ **do** $x[i] := (i-1) \cdot h1$ **od**:

> $p[1] := piecewise\left(x[1] \le x \le x[2], \frac{(x - x[2])}{(x[1] - x[2])}, 0\right)$:

Translation: Global Basis Function 1 = piecewise function. When x is between node 1 and node 2, use the defined first order Lagrange Polynomial, otherwise the function will equal zero. Left side x=0 (node 1) is a boundary node, so no GBF.

> **for** i **from** 2 **to** N **do** $p[i] := piecewise\left(x[i-1] \le x \le x[i], \frac{(x - x[i-1])}{(x[i] - x[i-1])}, x[i] \le x \le x[i+1], \frac{(x - x[i+1])}{(x[i] - x[i+1])}, 0\right)$ **od**:

Translation: (For 4 elements or 5 Nodes)

Global Basis function 2= piecewise function. When x is between node 1 and node 2, use Lagrange Polynomial (x-x[1])/(x[2]-x[1]), when x is between node 2 and 3 use Lagrange Polynomial (x-x[3])/(x[2]-x[3]), otherwise the function will equal zero.

Global Basis function 3= piecewise function. When x is between node 2 and node 3, use Lagrange Polynomial (x-x[2])/(x[3]-x[2]), when x is between node 3 and 4 use Lagrange Polynomial (x-x[4])/(x[3]-x[4]), otherwise the function will equal zero.

Global Basis function 4= piecewise function. When x is between node 3 and node 4, use Lagrange Polynomial (x-x[3])/(x[4]-x[3]), when x is between node 4 and 5 use Lagrange Polynomial (x-x[5])/(x[4]-x[5]), otherwise the function will equal zero.

>

> $p[N+1] := piecewise\left(x[N] \le x \le x[N+1], \frac{(x - x[N])}{(x[N+1] - x[N])}, 0\right)$:

> $plot(\{p[index]\$index = 1..N+1\}, x = 0..L);$

All GBF Plotted Together (N=4)

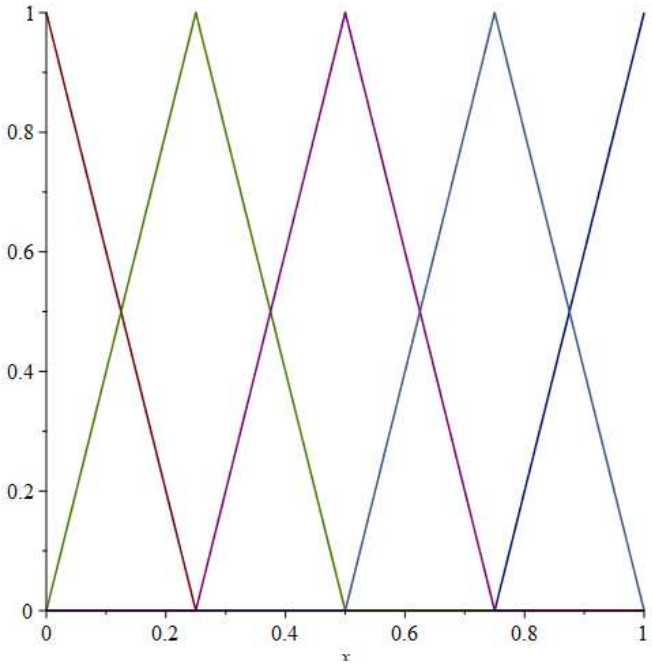
Red = p[1]

Green = p[2]

Purple = p[3]

Light Blue = p[4]

Blue = p[5]



Galerkin Approximation as a Linear Combination of the Global Finite Element Basis Functions

```
> uG := sum(u[n]·p[n], n = 1..N+1):
> duGdx := diff(uG, x):
```

Test Functions Identical to the Global Finite Element Basis Functions (Instead of using a guessed oscillating (sine) function like in Milestone 1)

```
> for j from 1 to N+1 do v[j] := p[j] od:
> for j from 1 to N+1 do dvdx[j] := diff(v[j], x) od:
```

Discrete Variational Formulation

Note that the Dirichlet-type BCs at x=0 and x=1 must be enforced (Essential BCs)

Note also that the Variational formulation is Integrated over the span of the Two Contiguous Elements (Because the Global FE Basis Function for any node connecting Two Contiguous Elements is non-zero ONLY within the elemens connected by the Node)

```
> eq[1] := u[1] = u0; Boundary Node and Condition (Left Side)
```

$$eq_1 := u_1 = 0 \tag{7}$$

```
> for j from 2 to N do eq[j] := int(E·I1·duGdx·dvdx[j], x = x[j-1]..x[j+1]) = int(M·v[j], x = x[j-1]
      ..x[j+1]) od;
```

$$eq_2 := -6.666666668\ 10^6\ u_1 + 1.333333334\ 10^7\ u_2 - 6.666666668\ 10^6\ u_3 = -22135.41667$$

$$eq_3 := -6.666666668\ 10^6\ u_2 + 1.333333334\ 10^7\ u_3 - 6.666666668\ 10^6\ u_4 = -29947.91667$$

$$eq_4 := -6.666666668\ 10^6\ u_3 + 1.333333334\ 10^7\ u_4 - 6.666666668\ 10^6\ u_5 = -22135.41667 \tag{8}$$

```
> eq[N+1] := u[N+1] = uL; Boundary Node and Condition (Right Side)
```

$$eq_5 := u_5 = 0 \tag{9}$$

The Finite Element Equations (A System of Linear Algebraic Equations)

```
> for i from 1 to N+1 do eq[i]od;
```

$$u_1 = 0$$

$$-6.666666668\ 10^6\ u_1 + 1.333333334\ 10^7\ u_2 - 6.666666668\ 10^6\ u_3 = -22135.41667$$

$$-6.666666668\ 10^6\ u_2 + 1.333333334\ 10^7\ u_3 - 6.666666668\ 10^6\ u_4 = -29947.91667$$

$$-6.666666668\ 10^6\ u_3 + 1.333333334\ 10^7\ u_4 - 6.666666668\ 10^6\ u_5 = -22135.41667$$

$$u_5 = 0 \tag{10}$$

Solving the System of Linear Algebraic Equations

```
> solve({eq[index]$index = 1..N+1}, {u[index]$index = 1..N+1}); assign(%)
```

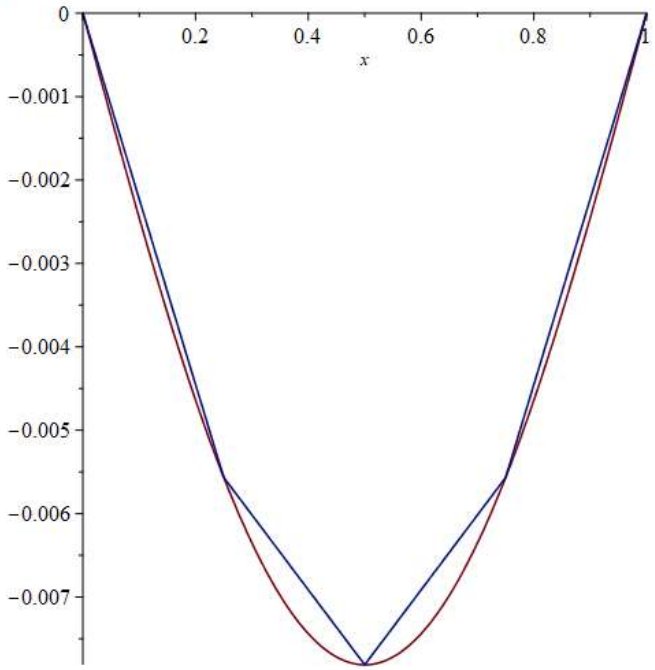
Deflection (u) at node 1 is 0 (BC), at node 2 is -0.005566, at node 3 is -0.007812, at node 4 is -.005566, and at node 5 is 0 (BC).

We can see that node 1 and 5 are equal (BCs) and node 2 and 4 are equal due to symmetry.

$$\{u_1 = 0.,\ u_2 = -0.005566406244,\ u_3 = -0.007812499991,\ u_4 = -0.005566406244,\ u_5 = 0.\} \tag{11}$$

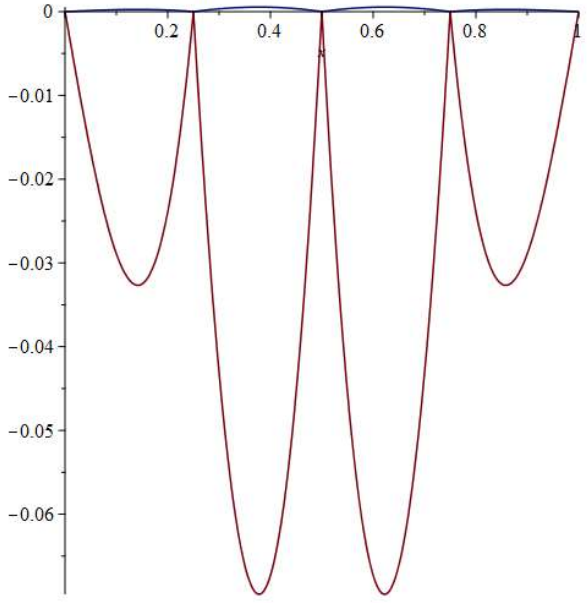
Comparing the Galerkin FE Approximation with the Exact Solution

```
> plot({uex(x), uG}, x = 0..L);
```



Computing the Local Error

```
> e1 := abs(uex(x) − uG) : e1r := e1/uexMax : plot({e1, e1r}, x = 0..L);
```




```
> Aex := evalf(int(uex(x), x = 0..L)); AG := evalf(int(uG, x = 0..L)); Aex − AG;
```

$$Aex := -0.004999999999$$

$$AG := -0.004736328120$$

$$-0.000263671879$$

Computing the Global Error as the Energy Norm of the Error

```
> evalf(sqrt(int((1/2)·diff(e1, x)^2, x = 0..L))); evalf(sqrt(int((1/2)·diff(e1r, x)^2, x = 0..L)));
```

$$0.002768835782$$

$$0.3544109801$$

Figure 30: Maple Four Element Galerkin Approximation Analysis

Galerkin FEM

N=Number of Elemens of uniform size h1

> $N := 8 : h1 := \frac{L}{N} :$

Global Finite Element Basis Functions - First Order Lagrange Polynomial

> **for** i **from** 1 **to** $N+1$ **do** $x[i] := (i-1)\cdot h1$ **od**:

> $p[1] := piecewise\left(x[1] \le x \le x[2], \frac{(x-x[2])}{(x[1]-x[2])}, 0\right)$:

Translation: Global Basis Function 1 = piecewise function. When x is between node 1 and node 2, use the defined first order Lagrange Polynomial, otherwise the function will equal zero. Left side x=0 (node 1) is a boundary node, so no GBF.

> **for** i **from** 2 **to** N **do** $p[i] := piecewise\left(x[i-1] \le x \le x[i], \frac{(x-x[i-1])}{(x[i]-x[i-1])}, x[i] \le x \le x[i+1], \frac{(x-x[i+1])}{(x[i]-x[i+1])}, 0\right)$ **od**:

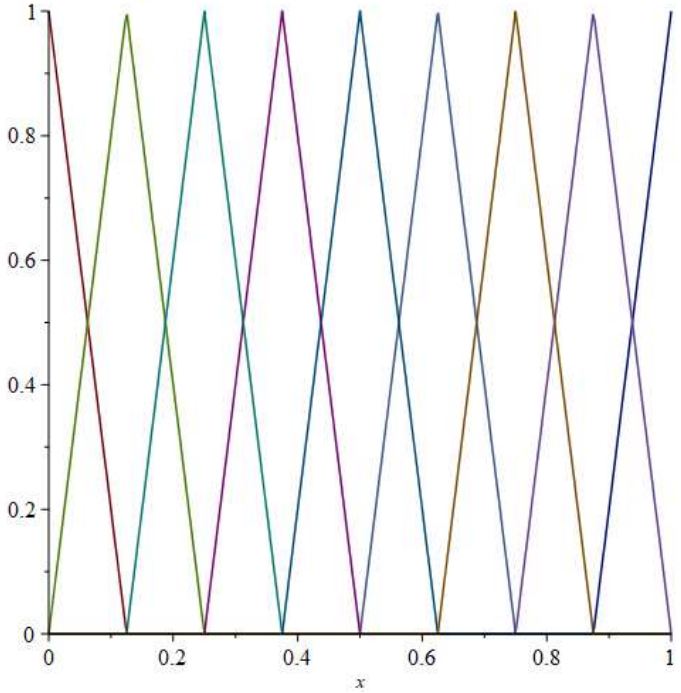
Translation: (For 4 elements or 5 Nodes)
Global Basis function 2= piecewise function. When x is between node 1 and node 2, use Lagrange Polynomial (x-x[1])/(x[2]-x[1]), when x is between node 2 and 3 use Lagrange Polynomial (x-x[3])/(x[2]-x[3]), otherwise the function will equal zero.
Global Basis function 3= piecewise function. When x is between node 2 and node 3, use Lagrange Polynomial (x-x[2])/(x[3]-x[2]), when x is between node 3 and 4 use Lagrange Polynomial (x-x[4])/(x[3]-x[4]), otherwise the function will equal zero.
Global Basis function 4= piecewise function. When x is between node 3 and node 4, use Lagrange Polynomial (x-x[3])/(x[4]-x[3]), when x is between node 4 and 5 use Lagrange Polynomial (x-x[5])/(x[4]-x[5]), otherwise the function will equal zero.

>

> $p[N+1] := piecewise\left(x[N] \le x \le x[N+1], \frac{(x-x[N])}{(x[N+1]-x[N])}, 0\right)$:

> $plot(\{p[index]\$index = 1..N+1\}, x = 0..L);$

All GBF Plotted Together (N=8)



Galerkin Approximation as a Linear Combination of the Global Finite Element Basis Functions

> $uG := sum(u[n] \cdot p[n], n = 1..N+1)$:

> $duGdx := diff(uG, x)$:

Test Functions Identical to the Global Finite Element Basis Functions (Instead of using a guessed oscillating (sine) function like in Milestone 1)

> **for** j **from** 1 **to** $N+1$ **do** $v[j] := p[j]$ **od**:

> **for** j **from** 1 **to** $N+1$ **do** $dvdx[j] := diff(v[j], x)$ **od**:

Discrete Variational Formulation

Note that the Dirichlet-type BCs at x=0 and x=1 must be enforced (Essential BCs)

Note also that the Variational formulation is Integrated over the span of the Two Contiguous Elements (Because the Global FE Basis Function for any node connecting Two Contiguous Elements is non-zero ONLY within the elemens connected by the Node)

> $eq[1] := u[1] = u0$; **Boundary Node and Condition (Left Side)**

$$eq_1 := u_1 = 0 \tag{3}$$

> **for** j **from** 2 **to** N **do** $eq[j] := int(E \cdot I1 \cdot duGdx \cdot dvdx[j], x = x[j-1]..x[j+1]) = int(M \cdot v[j], x = x[j-1]..x[j+1])$ **od**;

$$eq_2 := -1.333333334\,10^7\,u_1 + 2.666666667\,10^7\,u_2 - 1.333333334\,10^7\,u_3 = -6673.177083$$

$$eq_3 := -1.333333334\,10^7\,u_2 + 2.666666667\,10^7\,u_3 - 1.333333334\,10^7\,u_4 = -11555.98958$$

$$eq_4 := -1.333333334\,10^7\,u_3 + 2.666666667\,10^7\,u_4 - 1.333333334\,10^7\,u_5 = -14485.67708$$

$$eq_5 := -1.333333334\,10^7\,u_4 + 2.666666667\,10^7\,u_5 - 1.333333334\,10^7\,u_6 = -15462.23958$$

$$eq_6 := -1.333333334\,10^7\,u_5 + 2.666666667\,10^7\,u_6 - 1.333333334\,10^7\,u_7 = -14485.67708$$

$$eq_7 := -1.333333334\,10^7\,u_6 + 2.666666667\,10^7\,u_7 - 1.333333334\,10^7\,u_8 = -11555.98958$$

$$eq_8 := -1.333333334\,10^7\,u_7 + 2.666666667\,10^7\,u_8 - 1.333333334\,10^7\,u_9 = -6673.177083 \tag{4}$$

> $eq[N+1] := u[N+1] = uL$; **Boundary Node and Condition (Right Side)**

$$eq_9 := u_9 = 0 \tag{5}$$

The Finite Element Equations (A System of Linear Algebraic Equations)

> **for** i **from** 1 **to** $N+1$ **do** $eq[i]$**od**;

$$u_1 = 0$$

$$-1.333333334\,10^7\,u_1 + 2.666666667\,10^7\,u_2 - 1.333333334\,10^7\,u_3 = -6673.177083$$

$$-1.333333334\,10^7\,u_2 + 2.666666667\,10^7\,u_3 - 1.333333334\,10^7\,u_4 = -11555.98958$$

$$-1.333333334\,10^7\,u_3 + 2.666666667\,10^7\,u_4 - 1.333333334\,10^7\,u_5 = -14485.67708$$

$$-1.333333334\,10^7\,u_4 + 2.666666667\,10^7\,u_5 - 1.333333334\,10^7\,u_6 = -15462.23958$$

$$-1.333333334\,10^7\,u_5 + 2.666666667\,10^7\,u_6 - 1.333333334\,10^7\,u_7 = -14485.67708$$

$$-1.333333334\,10^7\,u_6 + 2.666666667\,10^7\,u_7 - 1.333333334\,10^7\,u_8 = -11555.98958$$

$$-1.333333334\,10^7\,u_7 + 2.666666667\,10^7\,u_8 - 1.333333334\,10^7\,u_9 = -6673.177083$$

$$u_9 = 0 \tag{6}$$

Solving the System of Linear Algebraic Equations

```
> solve({eq[index]$index=1..N+1}, {u[index]$index=1..N+1}); assign(%)
```

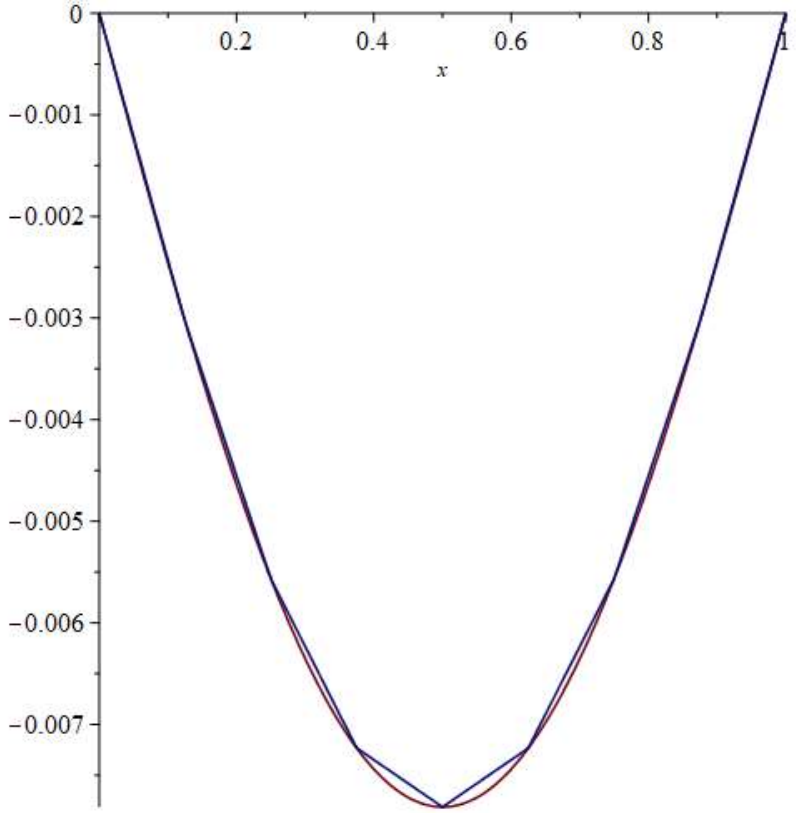
Deflection (u) at node 1 is 0 (BC), at node 2 is -0.005566, at node 3 is -0.007812, at node 4 is -.005566, and at node 5 is 0 (BC).

We can see that node 1 and 5 are equal (BCs) and node 2 and 4 are equal due to symmetry.

$$\{u_1 = 0.,\ u_2 = -0.003033447278,\ u_3 = -0.005566406273,\ u_4 = -0.007232666046,\ u_5 = -0.007812500033,\ u_6 = -0.007232666046,\ u_7 = -0.005566406273,\ u_8 = -0.003033447278,\ u_9 = 0.\} \quad (7)$$

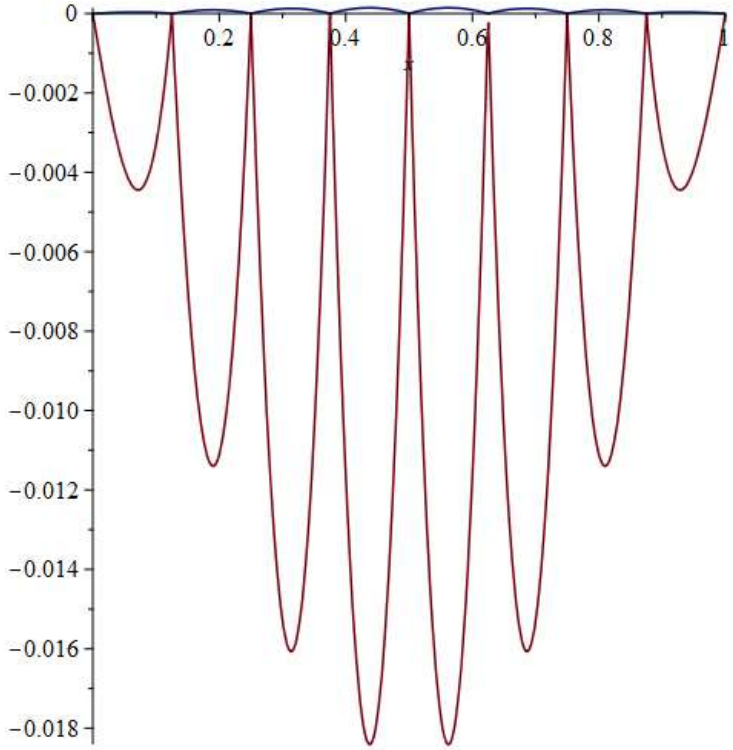
Comparing the Galerkin FE Approximation with the Exact Solution

```
> plot({uex(x), uG}, x=0..L);
```



Computing the Local Error

```
> e1 := abs(uex(x) - uG) : e1r := e1/uexMax : plot({e1, e1r}, x = 0..L);
```




```
> Aex := evalf(int(uex(x), x = 0..L)); AG := evalf(int(uG, x = 0..L)); Aex - AG;
```

$$Aex := -0.004999999999$$

$$AG := -0.004934692403$$

$$-0.000065307596$$

Computing the Global Error as the Energy Norm of the Error

```
> evalf(sqrt(int((1/2)*diff(e1, x)^2, x = 0..L))); evalf(sqrt(int((1/2)*diff(e1r, x)^2, x = 0..L)));
```

$$0.001393991965$$

$$0.1784309715$$

Figure 31: Maple Eight-Element Galerkin Approximation Analysis

Finite Element Galerkin Approximation using Local and Global Basis Functions (Cantilever Beam)

Using the same methods for the Finite Element Galerkin approximation for the simply supported beam will allow us to develop the approximation using the cantilever beam. We know that though the differential equation is the same, the moment equation, boundary conditions, and variational formulas differ. To fully cover how to develop the Galerkin approximation with global basis functions, we will make a three-element model (four nodes). This will enable a more detailed explanation of the treatment of the interior nodes.

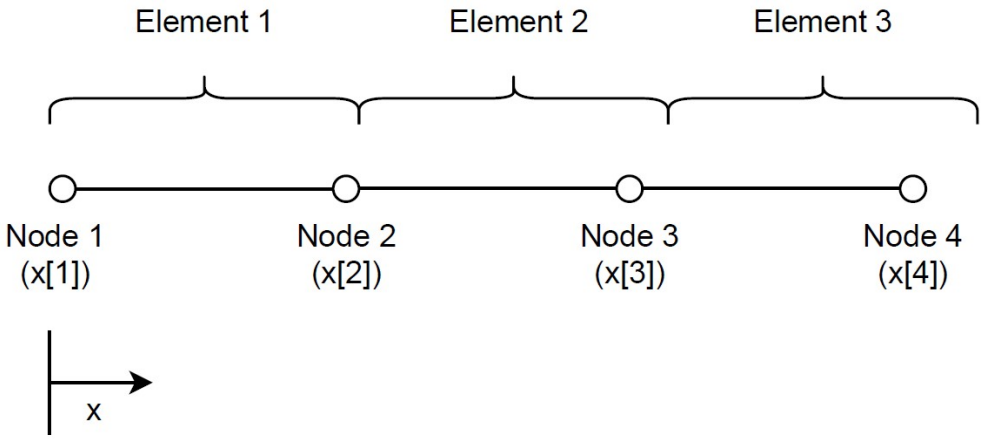


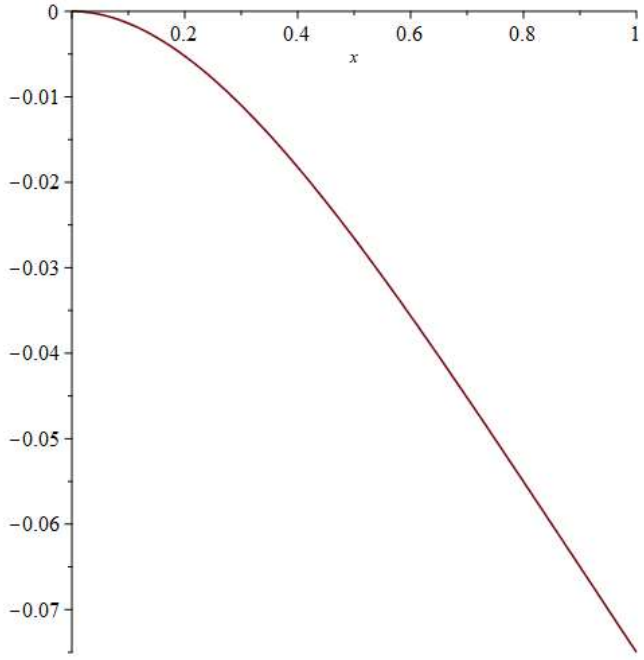
Figure 32: Three Element Model (Four Nodes) – Cantilever Beam

Maple coding of the cantilever beam can be seen in Figures 7 and 8, but repeated below with values implemented for the constants as well as the moment and slope at x=L.

```
> restart;
> q := -1e6 : L := 1 : E := 2e11 : h := .1 : b := .1 : I1 := b*h^3/12 :
> M := -q*(L-x)^2/2;
                              M := 500000.0000 (1 - x)^2
> de1 := -E*I1*(d^2/dx^2)(uex(x)) = M;
                    de1 := -1.666666667 10^6 (d^2/dx^2) uex(x) = 500000.0000 (1 - x)^2
> u0 := 0; s0 := (q*L^3/(6*E*I1));
                                   u0 := 0
                                s0 := -0.1000000000
> ss0 := -q*x/(6*E*I1)*(3*L^2 - 3*L*x + x^2);
                          ss0 := 0.1000000000 x (x^2 - 3 x + 3)
> BCA := uex(0) = u0; BCB := D(uex)(L) = s0;
                                 BCA := uex(0) = 0
                              BCB := D(uex)(1) = -0.1000000000
> dsolve({de1, BCA, BCB}, uex(x));
                  uex(x) = -125000000 (x - 1)^4/5000000001 - x/10 + 125000000/5000000001
```

Figure 33: Maple Code Cantilever Beam Calculation of Exact Solution

```
> assign(%) :
>
> plot(uex(x), x = 0 ..L);
```




```
> dudxL := evalf(subs(x = L, diff(uex(x), x)));
```

$$dudxL := -0.1000000000$$

Figure 34: Maple Code Cantilever Beam Plot of Exact Solution and Slope at x=L

3 Element Model

```
> N := 3 : h := L/3 : x[1] := 0; x[2] := h; x[3] := 2·h; x[4] := L :
```

$$x_1 := 0$$

$$x_2 := \frac{1}{3}$$

$$x_3 := \frac{2}{3}$$

```
> p11 := (x − x[2])/(x[1] − x[2]); p12 := (x − x[1])/(x[2] − x[1]);
```

$$p11 := -3x + 1$$

$$p12 := 3x$$

```
> p21 := (x − x[3])/(x[2] − x[3]); p22 := (x − x[2])/(x[3] − x[2]);
```

$$p21 := -3x + 2$$

$$p22 := 3x - 1$$

```
> p31 := (x − x[4])/(x[3] − x[4]); p32 := (x − x[3])/(x[4] − x[3]);
```

$$p31 := -3x + 3$$

$$p32 := 3x - 2$$

```
> plot({p11, p12}, x = x[1] ..x[2]) :
> plot({p21, p22}, x = x[2] ..x[3]) :
> p[1] := piecewise(x[1] ≤ x ≤ x[2], p11, x[2] ≤ x ≤ x[3], 0) :
> p[2] := piecewise(x[1] ≤ x ≤ x[2], p12, x[2] ≤ x ≤ x[3], p21, 0) :
> p[3] := piecewise(x[2] ≤ x ≤ x[3], p22, x[3] ≤ x ≤ x[4], p31, 0) :
> p[4] := piecewise(x[3] ≤ x ≤ x[4], p32, 0) :
> plot({p[1], p[2], p[3], p[4]}, x = x[1] ..x[4]);
```

Figure 35: Maple Code Cantilever Beam – Local and Global Basis Function Construction

What we can see is that global basis function one and four for the end nodes are similar to the global basis functions for the simply supported beam, but global basis functions two and three as interior nodes will differ from the end nodes. These global basis functions can be described as for global basis function p[2], the span of node one to node two, use the second half of element one. For the span of node two and node three, use first half of element one. Otherwise use zero. This is exactly similar to the p[2] of the two-element model of the simply supported beam. We have an additional interior node that we must consider. This will be constructed the same as p[2] but for the next element and consequent nodes. Global basis function p[3], the span of node two to node three, use the second half of element two. For the span of node three and node four, use first half of element three. Otherwise use zero. The plot of the global basis functions can be seen below

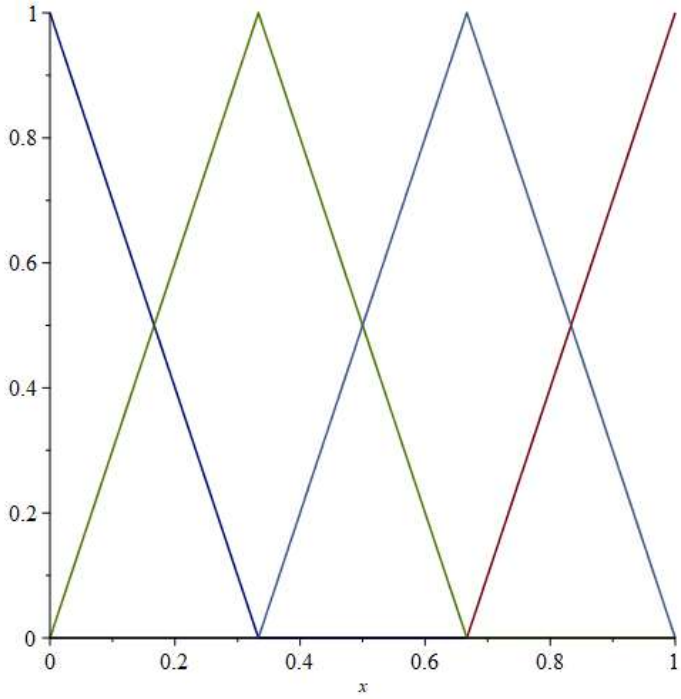


Figure 36: Maple Plot of Global Basis Functions – Three-Element Cantilever Beam

Further into the code, we can similarly see the construction and analysis of the Galerkin approximation to the exact solution of the cantilever beam.

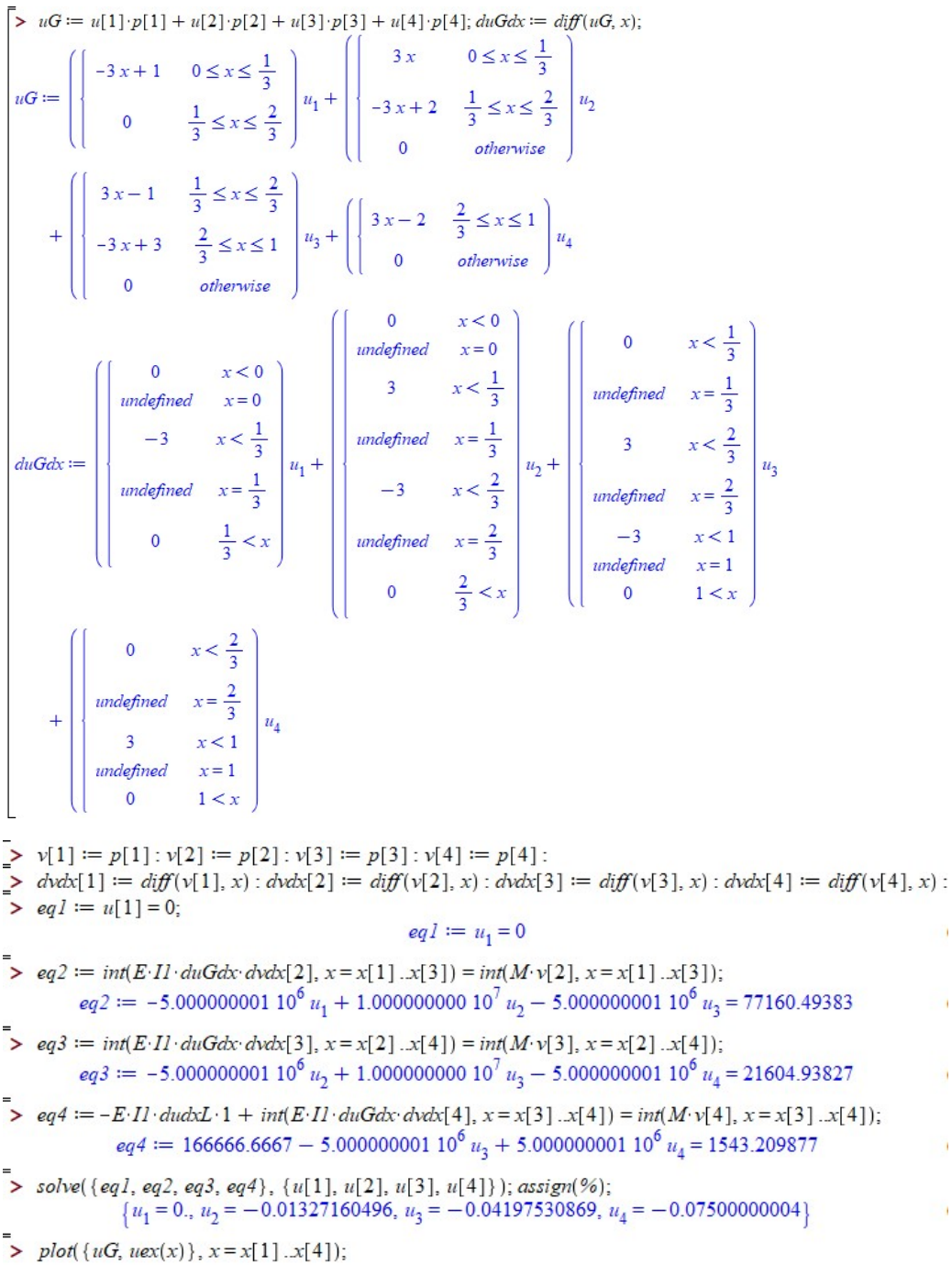


Figure 37: Maple Coding of Galerkin Approximation for Three – Element Cantilever Beam

As we can see, there is an added variational equation (eq4). With the added element, we add a node. Equation 1 represents the dirilecht boundary condition at the fixed point of the cantilever beam, so it will be equal to zero. Equation 2 and equation 3 are interior nodes of the model and requrie the implementation of the variational formula, but you will notice that they are only pertinent over the span of nodes of elements that are shared with the node (i.e. eq3 for node 3 is integrated oever node two and node 4 since node three connects to element two and three, which contain nodes two, three, and four.

Eq4 differs from all of the equations because it represent node four on the free end of the model. This boundary condition is a neumann boundary condition which means that the boundary condition is aderivative, which we have the derivative of the deflection. Looking back at the variational formula at this neumann boundary condition, we have Eq 1.27. This includes the derivative at the boundary and the test function at the boundary. This can be seen integrated into the eq4 of the maple code. This equation is also integrated only over the range of node three and four. The plot of the exact solution vs the Galerkin approximation is shown below.

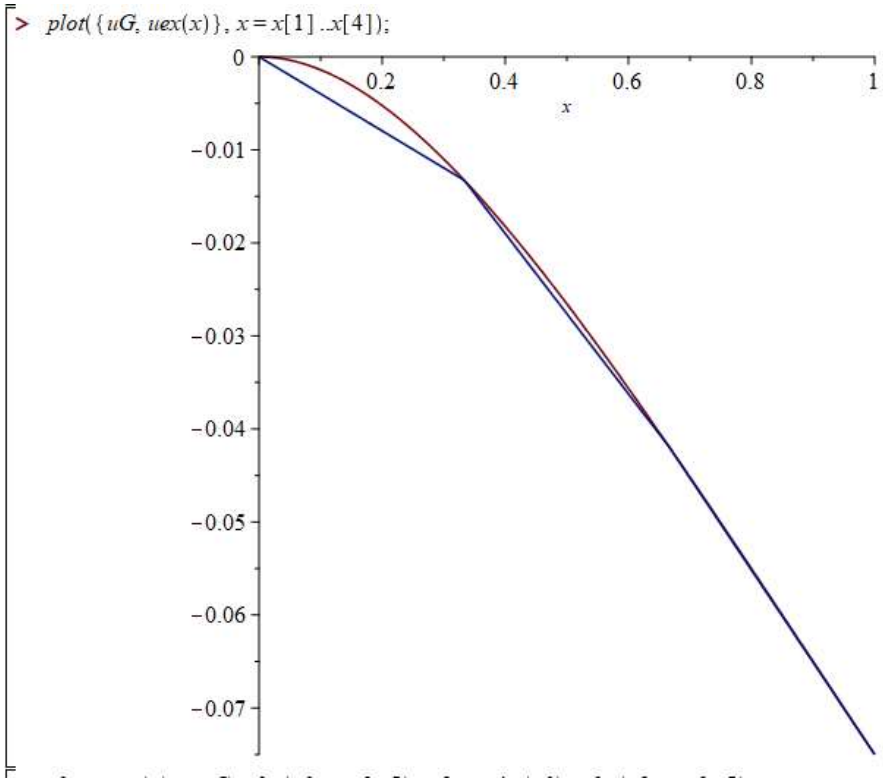
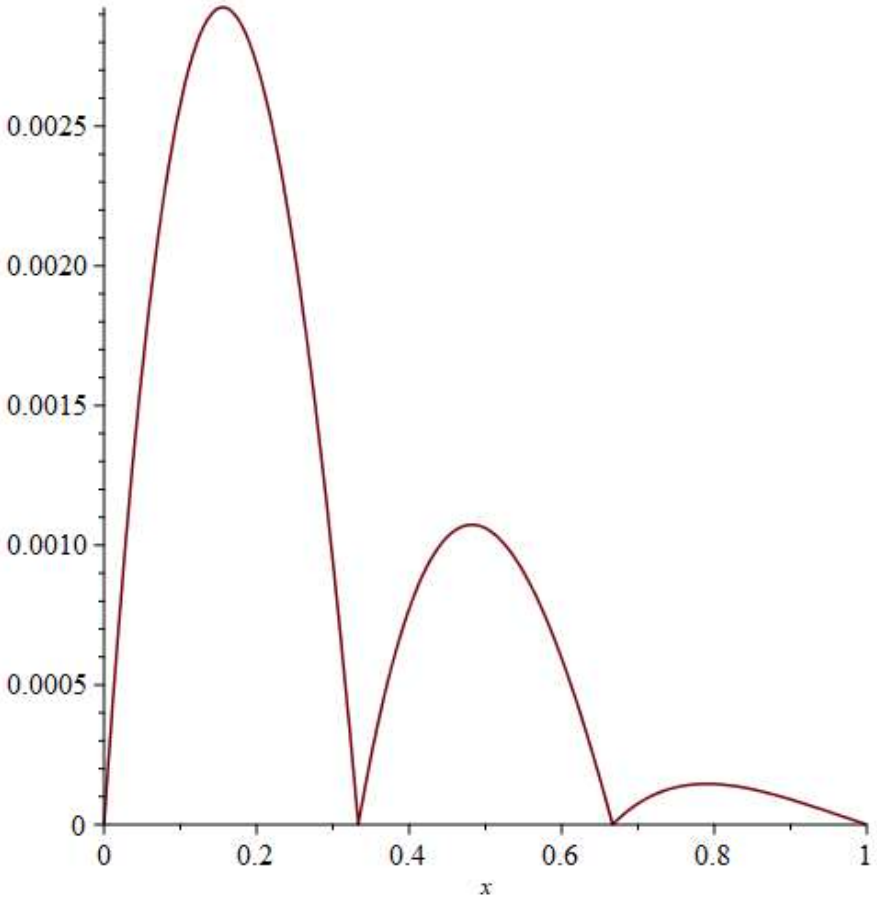


Figure 38: Maple Plot – Exact Solution vs Galerkin Approximation – Three – Element Cantilever

We will again evaluate the error exhibited from the Galerkin approximation and the exact solution. We first plot the difference between the exact and the Galerkin approximation as shown in Figure 35, and then obtain exact values of the global error in the exact solution and the Galerkin approximation. The Energy Norm Error is also analyzed as seen in equation eN.

```
> e1 := uex(x) - uG : plot(e1, x = 0 ..L) : e2 := abs(e1) : plot(e2, x = 0 ..L);
```



Global error

```
> Aex := evalf(int(uex(x), x = 0 ..L)); AG := evalf(int(uG, x = 0 ..L)); (AG - Aex)/Aex;
```

$$Aex := -0.03000000000$$

$$AG := -0.03091563789$$

$$0.03052126300$$

Energy Norm Error (Goes smaller and smaller as you add more elements)

```
> eN := evalf(sqrt(1/2 · int(e2^2, x = 0 ..L)));
```

$$eN := 0.0009276390262$$

Figure 39: Maple Plot – Exact vs Galerkin Error (Three-Element Cantilever)

Finite Element Galerkin Approximation using COMSOL (Simply Supported Beam)

Obtaining an understanding of the coding occurring in the background of many FEA software packages, we can now move onto using a software package to compare our Maple coding. This problem will be analyzed using a package named COMSOL. This is a very user-friendly and powerful FEA package that will allow for the study of 1D, 2D and 3D Finite Element model solutions. Defining the differential formulas and exact solutions will be the bulk of the initial set up, as well as defining our geometry and meshing conditions. The software is capable of conducting the coding that we have been completing by hand.

We first assure that we select a 1D space dimension. We then must select the problem type we are working on. There are many to choose from, but since a solid mechanics problem-set is not defined, we can choose the coefficient form PDE which will allow the manipulation of the equation to fit our problem at hand. Since our model does not depend on time, we will select a stationary study type. We must now define the geometry of our model by using intervals. We can define the amount of nodes and elements by using the intervals. The models that we have produced using Maple are the two element and an eight element, which will show as the models in COMSOL as defined in the intervals.

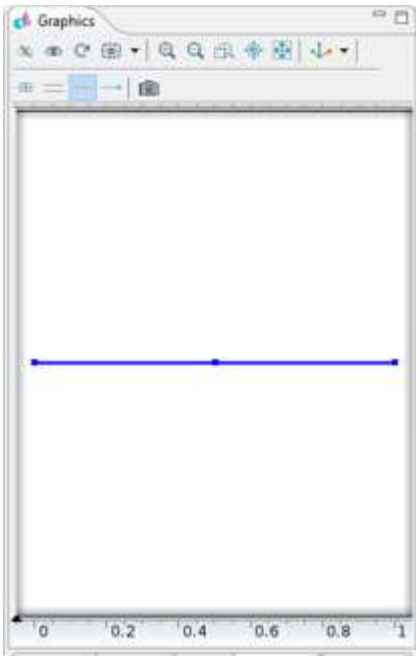


Figure 40: COMSOL Two Element Model

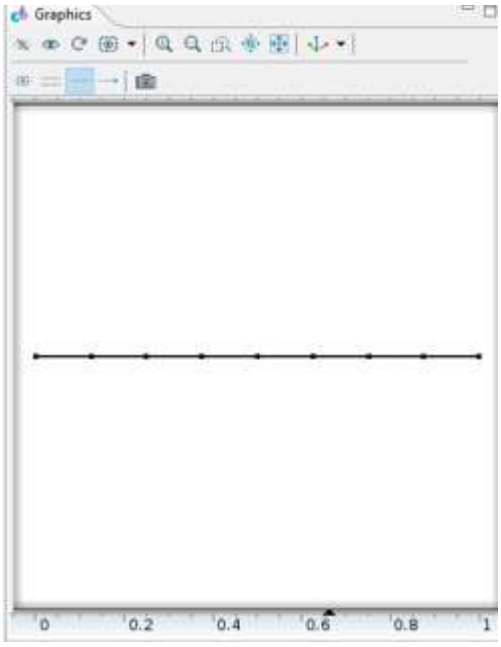


Figure 41: COMSOL Eight Element Model

We then move forward by defining the coefficients and source term that will allow the general differential form to take the form of our defined differential formulation.

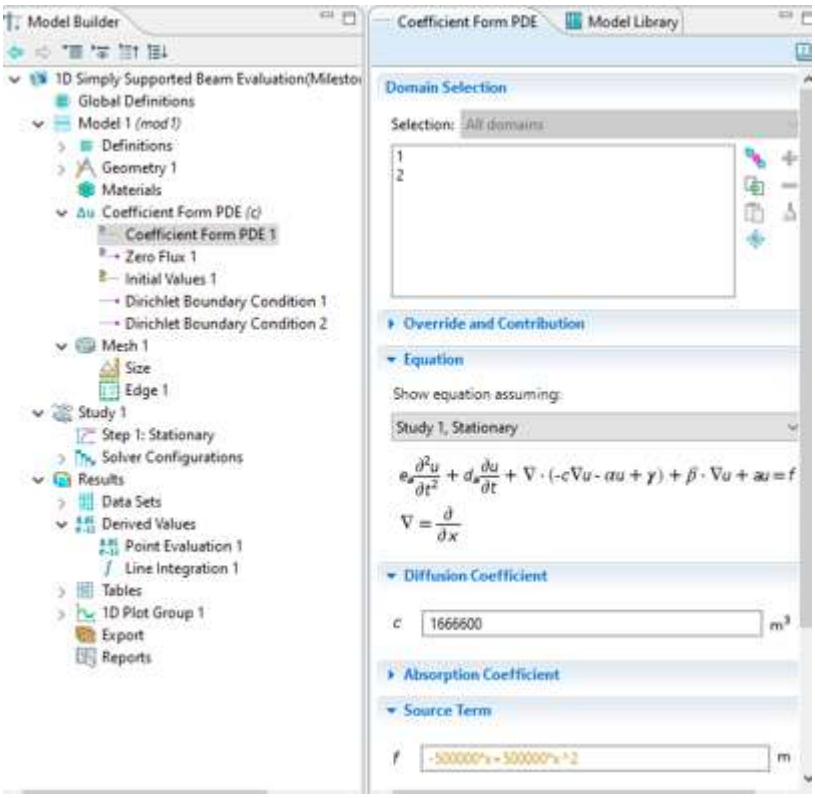


Figure 42: COMSOL PDE Definition

We must also define our shape functions through the Lagrange Linear discretization. Boundary conditions in our case will be enforced by selecting Dirichlet Boundary Conditions since the deflection at both ends is zero. If we were to model the cantilever beam with the derivative of the deflection at the free end, then we would select "flux" and enter the appropriate value.

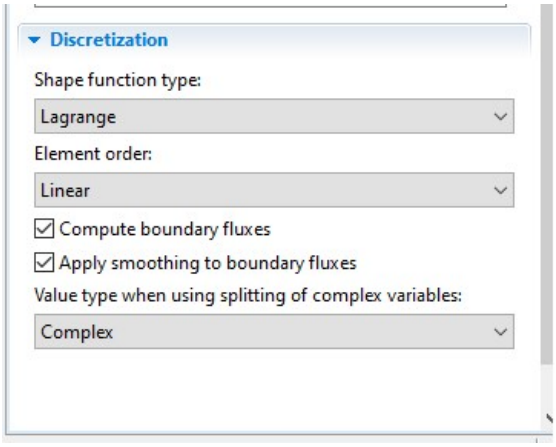


Figure 43: COMSOL Discretization Definitions

We can then compute the solution and obtain the graph for the exact solution and the maximum deflection of the beam at the center (simply supported beam).

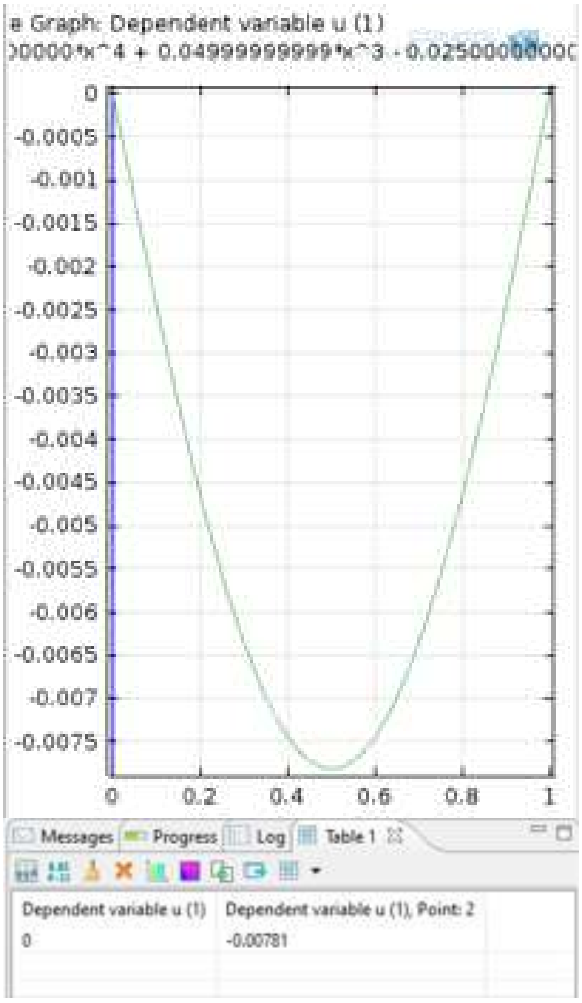


Figure 44: COMSOL Simply Supported Beam Exact Solution

Now that we have plotted and computed the exact solution, we can use the meshing properties in COMSOL that will use the shape functions defined by a linear lagrange discretization just as we did in the Maple code. This can be viewed as the Galerkin approach. We must define the element size as follows for an eight element model (See Figure 42).

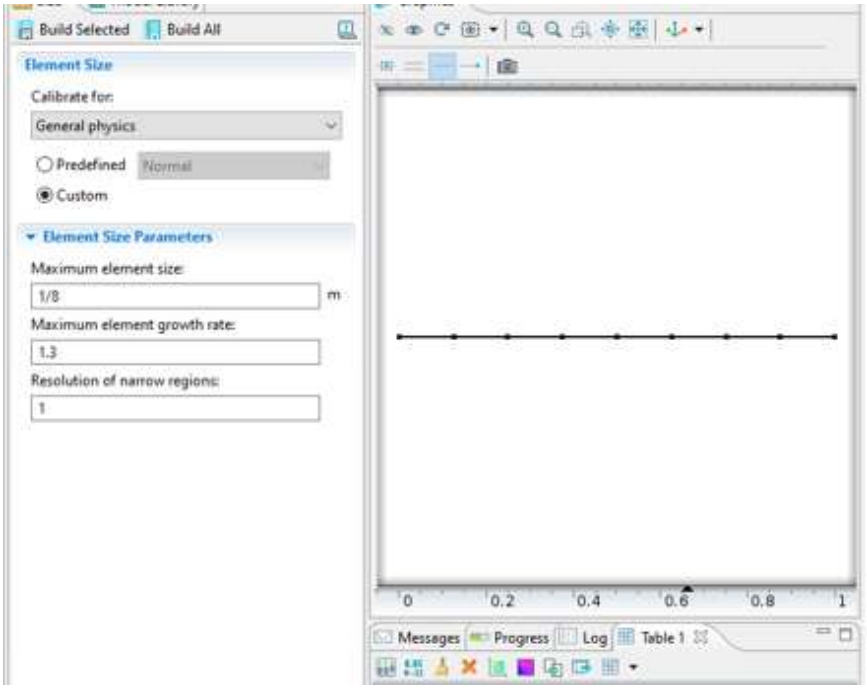


Figure 45: COMSOL Mesh Size Definition (8 Element Model)

When we run the study again, and plot the results, we obtain a plot that contains both the exact solution and the approximation with the eight element meshing.

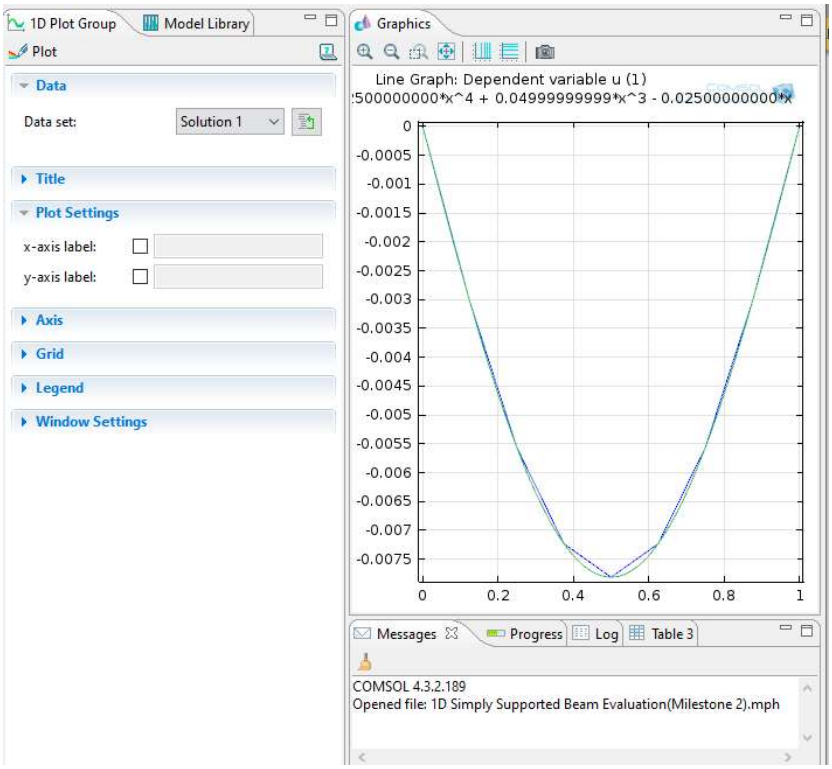


Figure 46: COMSOL Plot Meshing Solution vs Exact Solution (Simply Supported Beam)

As we have looked at the error of the galerkin approximation in the Maple coding, we can also view this in the COMSOL solution as shown below

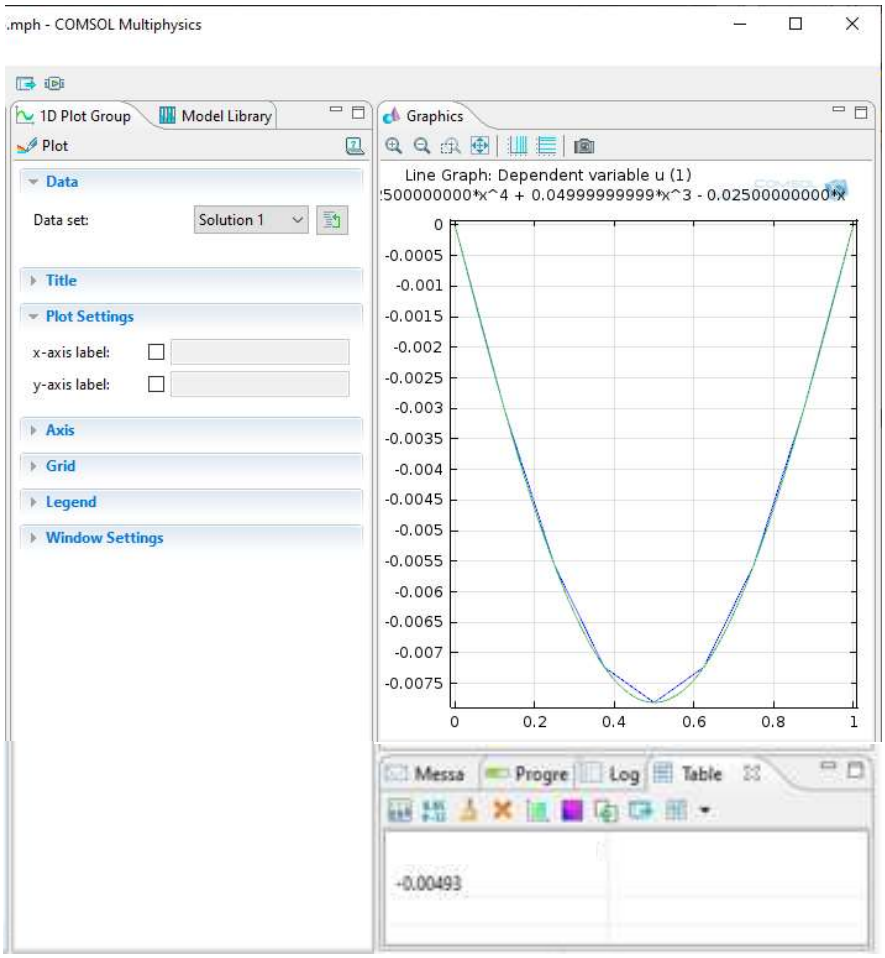


Figure 47: COMSOL Meshing Error

Finite Element - Inconel 625 Annealed Plate 2D (Cantilever Beam)

For a transition to a more realistic problem, Inconel 625 will be used in a simple 1D finite element system as demonstrated above. The dimensions of the beam are scaled up to a one meter length beam for more realistic take. We are more interested in the material properties in this problem that can translate throughout. The Maple programming is as shown below for a cantilever beam with a point load on the end

```
> restart;
```

Navin_Ian MANE 4240 Milestone 3, 2D Galerkin Approximation

Bending Moment Equation of a beam from Strength of Materials

Incoloy 625 Annealed Plate at RT

```
>
> de1 := E·I1·d^2/dx^2(uex(x)) = P·x;
```

$$de1 := EI1\left(\frac{d^2}{dx^2}\,uex(x)\right) = Px$$

```
> u0 := 0;
```

$$u0 := 0$$

```
>
> BCA := uex(L) = u0; BCB := D(uex)(L) = u0;
```

$$BCA := uex(L) = 0$$

$$BCB := D(uex)(L) = 0$$

```
> s1 := dsolve({de1, BCA, BCB}, uex(x)); assign(%);
```

$$s1 := uex(x) = \frac{Px^3}{6\,EI1} - \frac{PL^2x}{2\,EI1} + \frac{PL^3}{3\,EI1}$$

```
> subs(x = 0, uex(x));
```

$$\frac{PL^3}{3\,EI1}$$

```
> P := -1e6 : L := 1 : E := 8.14e10 : h := .1 : b := .1 : I1 := b·h^3/12 : rho := 8440 : nu := 0.278 :
```

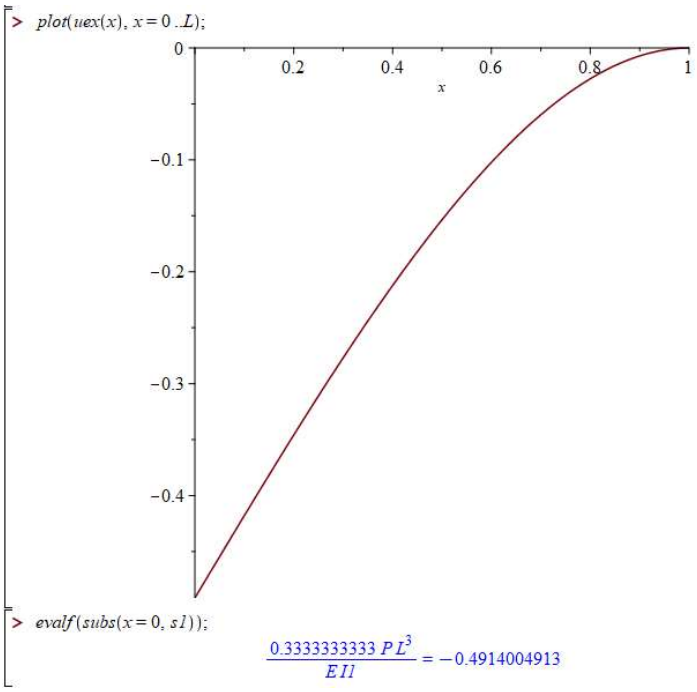


Figure 48: 1D Exact Solution Cantilever Inconel 625 Annealed Plate at RT

Moving to a two-dimensional approximation, we use Timoshenko beam theory rather than the Euler Bernoulli.

As stated in the plate assumptions, we consider

1.) Deflection of the midplane is small compared to the thickness
2.) The midplane remains unstrained subsequent to bending
3.) Plane shear strains are zero. Out of plane strain is omitted
4.) Out of plane normal stress is neglected

The difference with Timoshenko's theory is that in bending of the beam, rotation between the cross section and neutral axis occurs and therefore creates shear deformation. Neither of these effects are considered in the Euler Bernoulli Theory, which means that the Timenshenko beam will have a larger deflection when compared to the deflection of Euler due to a decreased stiffness of the beam. These effects are more apparent in larger/longer beams since shear forces that oppose eachother are further away. As the beam gets shorter, these effects are not as apparent. The Timoshenko beam will approach the Euler beam when the ratio of the shear stress and strain approach infinity. This makes the beam stiffer when considering shear.

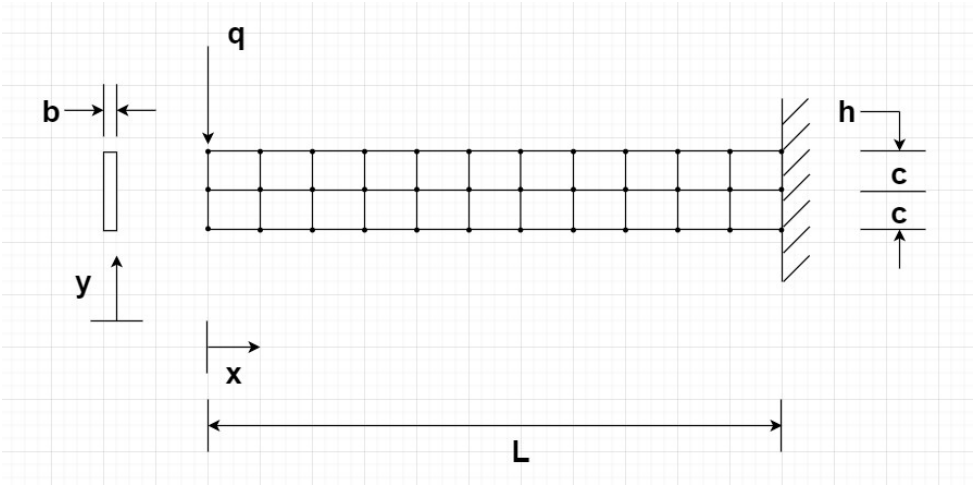


Figure 49: 2D Cantilever Beam Nodal Network

The solution can be obtained though polynomials for two dmensional problems when using Timoshenko [12]. The differential equation is as shown as derived from the biharmonic equation with no lateral load

$$\nabla^4 \Phi = 0 \tag{1.33}$$

$$\frac{\partial^4 \Phi}{\partial x^4} + 2\frac{\partial^4 \Phi}{\partial x^2 \partial y^2} + \frac{\partial^4 \Phi}{\partial y^4} = 0 \tag{1.34}$$

Utilizing a stress function to the fourth degree polynomial, we have

$$\Phi_4 = \frac{a_4}{4*3}x^4 + \frac{b_4}{3*2}x^3 y + \frac{c_4}{2}x^2 y^2 + \frac{d_4}{3*2}xy^3 + \frac{e_4}{4*3}y^4 \tag{1.35}$$

When the polynomial is substitued into Eq 1.34, we find that

$$e_4 = -(2c_4 + a_4) \tag{1.36}$$

Solving for the components of stress with the soleved for coefficients, we have

$$\sigma_x = \frac{\partial^2 \Phi_4}{\partial y^2} = c_4 x^2 d_4 xy - (2c_4 + a_4)y^2 \tag{1.37}$$

$$\sigma_y = \frac{\partial^2 \Phi_4}{\partial x^2} = a_4 x^2 + b_4 xy + c_4 y^2 \tag{1.38}$$

$$\tau_{xy} = \frac{\partial^2 \Phi_4}{\partial x^2 \partial y^2} = -\frac{b_4}{2}x^2 - 2c_4 xy - \frac{d_4}{2}y^2 \tag{1.39}$$

Combining with pure shear and yielding a pure shear of -b2 on the stresses above, gives

$$\sigma_x = \frac{\partial^2 \Phi_4}{\partial y^2} = d_4 xy \tag{1.40}$$

$$\sigma_y = \frac{\partial^2 \Phi_4}{\partial x^2} = 0 \tag{1.41}$$

$$\tau_{xy} = \frac{\partial^2 \Phi_4}{\partial x^2 \partial y^2} = -b_2 - \frac{d_4}{2} y^2 \tag{1.42}$$

Since the cantilever beam is not subject to longitudinal forces , then the shear at +c and -c is

$$(\tau_{xy})_{y=\pm C} = -b_2 - \frac{d_4}{2} c^2 = 0 \tag{1.43}$$

And d4 can be found to be -2b2/c^2. B_2 can be solved for after summing the shearing foces over the ends (+c to -c)and setting equal to the end load P.

$$b_2 = \frac{3}{4} * \frac{P}{c} \tag{1.44}$$

Finally we can solve for the stresses by substituing in d4 and b2 which follow the Euler stresses

$$\sigma_x = \frac{\partial^2 \Phi_4}{\partial y^2} = -\frac{3}{2} * \frac{P}{c^3} xy \tag{1.45}$$

$$\sigma_y = \frac{\partial^2 \Phi_4}{\partial x^2} = 0 \tag{1.46}$$

$$\tau_{xy} = \frac{\partial^2 \Phi_4}{\partial x^2 \partial y^2} = -\frac{3P}{4c} * (1 - \frac{y^2}{c^2}) \tag{1.47}$$

To get to the dispalcements in the x and y direction, we must apply Hookes law

$$\in_x = \frac{\partial u}{\partial x} = \frac{\sigma_x}{E} = -\frac{Pxy}{EI}, \in_y = \frac{\partial v}{\partial y} = \frac{\nu \sigma_x}{E} = -\frac{\nu Pxy}{EI}, \tau_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial y} = \frac{\tau_{xy}}{G} = -\frac{P}{2IG}(c^2 - y^2) \tag{1.48}$$

And then

$$u = -\frac{Px^2 y}{2EI} + f(y),\ v = \frac{\nu P x y^2}{2EI} + f(x) \tag{1.49}$$

Solving for f(y) and f(x) by substituting into Eq. 1.47.

$$-\frac{Px^2}{2EI} + \frac{df(y)}{dy}, + \frac{\nu P y^2}{2E} + \frac{df(x)}{dx} = -\frac{P}{2IG}(c^2 - y^2) \tag{1.50}$$

Making the first term as a constant e and the second as a constant d, we can solve for f(y) and f(x)

$$f(y) = -\frac{\nu P y^3}{6EI} + \frac{Py^3}{6IG} + ey + g,\ f(x) = \frac{Px^3}{6EI} + dx + h \tag{1.51}$$

Plugging into 1.49,

$$u = -\frac{Px^2 y}{2EI} - \frac{\nu P y^3}{6EI} + \frac{Py^3}{6IG} + ey + g,\ v = \frac{\nu P x y^2}{2EI} + \frac{Px^3}{6EI} + dx + h \tag{1.52}$$

We now assume for the beam to remain rigid, the cross section centroid at the end is anchored. This means that the deflection in the x and y direction are zero at the end of the beam = L and at the center of the beam in the y axis. This means that g=0 and h=-PL^3/6EI – dL. If we then consider

that rotation is removed in the xy plane, then we can solve for d and e and finally our deflections in the x and y direction to be

$$u = -\frac{Px^2y}{2EI} - \frac{\nu Py^3}{6E} + \frac{Py^3}{6I} + \left(\frac{PL^2}{2EI} - \frac{Pc^2}{2IG}\right)y \tag{1.53}$$

$$v = \frac{\nu Pxy^2}{2E} + \frac{Px^3}{6EI} + \frac{PL^2x}{2EI} - \frac{PL^3}{3EI} \tag{1.54}$$

The utilization of these equations can be used in Maple to solve for the two dimensional Timenshenko theory

Navin_Ian MANE 4240 Milestone 3, 2D Timenshenko Approximation

Bending Moment Equation of a beam from Strength of Materials

Incoloy 625 Annealed Plate at RT

```
> restart;
> u := -P·x^2·y/(2·E·I1) - nu·P·y^3/(6·E·I1) + P·y^3/(6·I1·G) + (P·L^2/(2·E·I1) - P·c^2/(2·I1·G))·y;
```

$$u := -\frac{P x^2 y}{2 E I1} - \frac{\nu P y^3}{6 E I1} + \frac{P y^3}{6 I1 G} + \left(\frac{P L^2}{2 E I1} - \frac{P c^2}{2 I1 G}\right) y$$

```
> v := nu·P·x·y^2/(2·E·I1) + P·x^3/(6·E·I1) - P·L^2·x/(2·E·I1) + P·L^3/(3·E·I1);
```

$$v := \frac{\nu P x y^2}{2 E I1} + \frac{P x^3}{6 E I1} - \frac{P L^2 x}{2 E I1} + \frac{P L^3}{3 E I1}$$

```
>
> w := 0;
```

$$w := 0$$

```
> subs({y = 0, x = 0}, v);
```

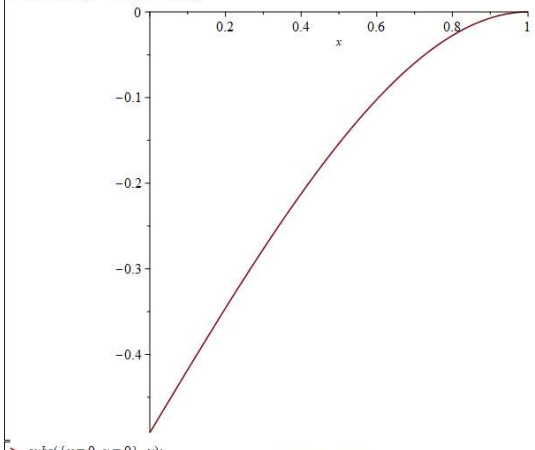
$$\frac{P L^3}{3 E I1}$$

```
> P := -1e6 : L := 1 : E := 8.14e10 : h := .1 : b := .1 : I1 := b·h^3/12 : rho := 8440 : nu := 0.278 : G := E/(2·(1 + nu)) : c := h/2 : B := b/2 :
> u; v;
```

$$0.7371007370\, x^2 y - 0.5597051597\, y^3 - 0.7323906633\, y$$

$$-0.2049140049\, x y^2 - 0.2457002457\, x^3 + 0.7371007370\, x - 0.4914004913$$

```
> plot(subs(y = 0, v), x = 0..L);
```




```
> subs({x = 0, y = 0}, v);
```

$$-0.4914004913$$

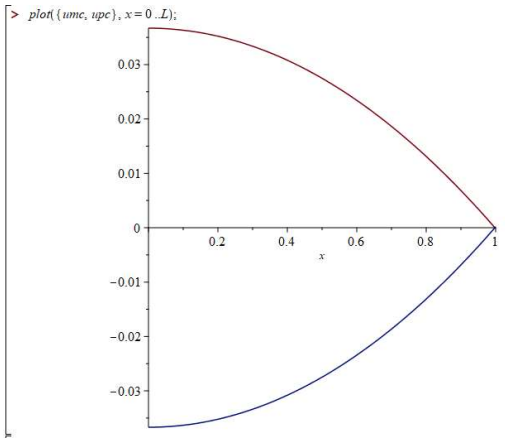
```
> umc := subs(y = -c, u);
```

$$umc := -0.03685503685\, x^2 + 0.03668949630$$

```
> upc := subs(y = c, u);
```

$$upc := 0.03685503685\, x^2 - 0.03668949630$$

Figure 50: 2D Exact Solution (Timoshenko) Inconel 625 Annealed Plate at RT



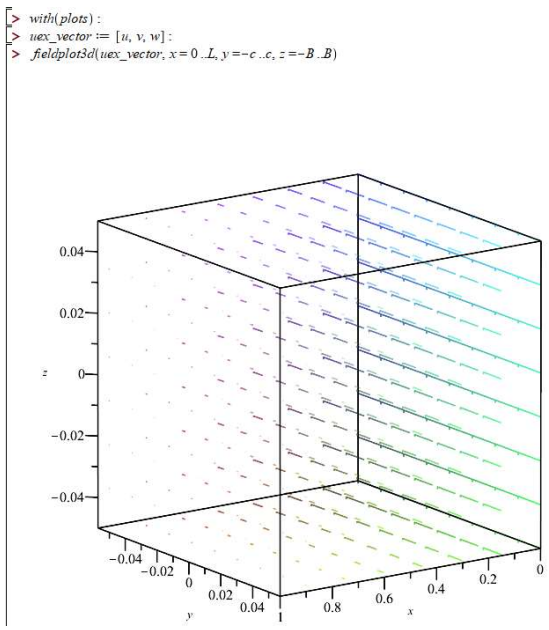


Figure 51: 2D Plot Cantilever Inconel 625 Annealed Plate at RT

This can now be compared to the finite element software COMSOL where the setup starts with selecting a 2D workspace and setting the parameters

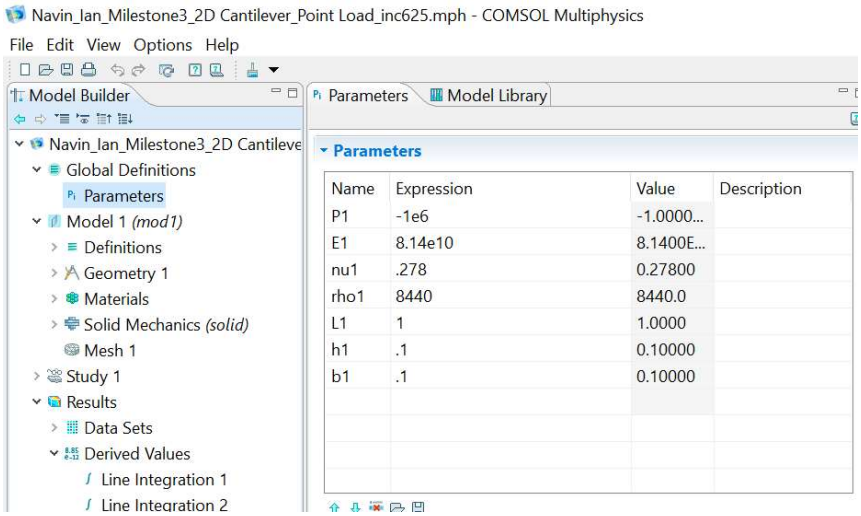


Figure 52: 2D Cantilever – Inconel 625 Annealed Plate at RT Parameters

Then to use the parameters to build the shape of the model

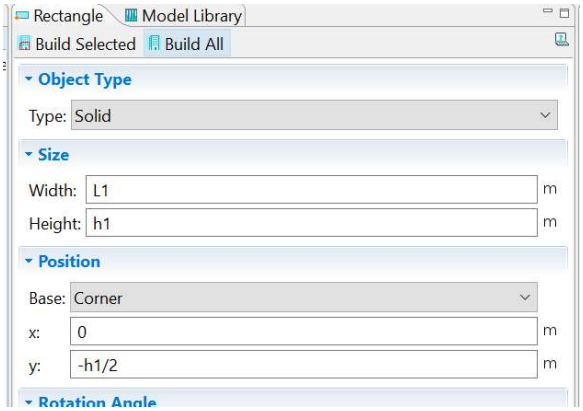


Figure 53: 2D Plot Cantilever Inconel 625 Annealed Plate at RT

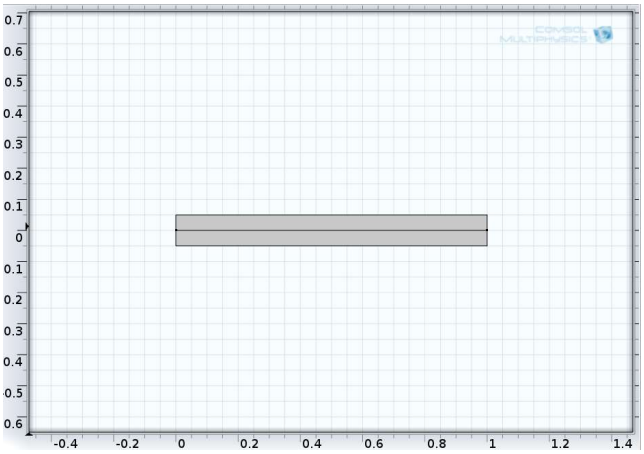


Figure 54: 2D Cantilever Beam Build

Now that I have the model, I will set the boundary conditions on the end and apply the load as shown below

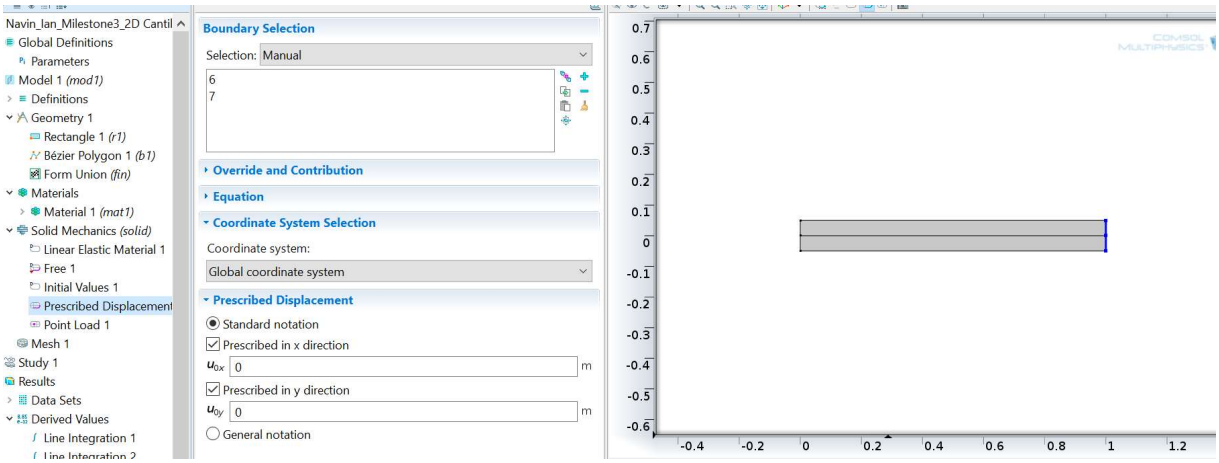


Figure 55: 2D Cantilever Beam Displacement Settings

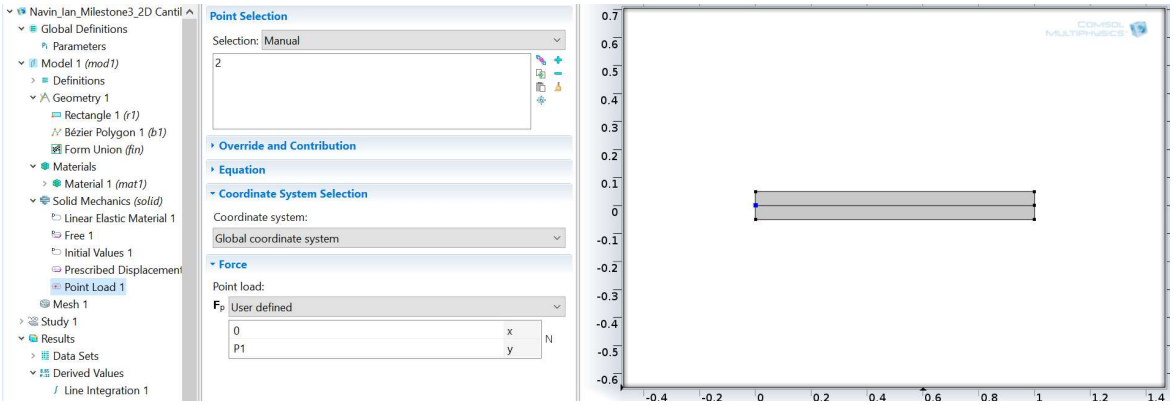


Figure 56: 2D Cantilever Beam Point Load Application

To set the global material parameters to the domain, we specific the domains for the material to be applied to and input the values of young's modulus, poisons ratio and the density

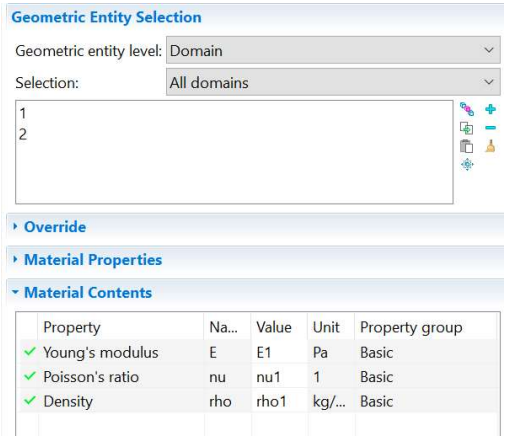


Figure 57: 2D Cantilever Beam Material Property Application

We are also conducting a Solid Mechanics model, so this is selected. Plane stress is also considered for the 2D approximation since we are not concerned with the thickness of beam therefore it does not need to be considered. We still input the thickness. In order to study the effects of the discretization, we will select a linear discretization and a quadratic discretization for comparison.

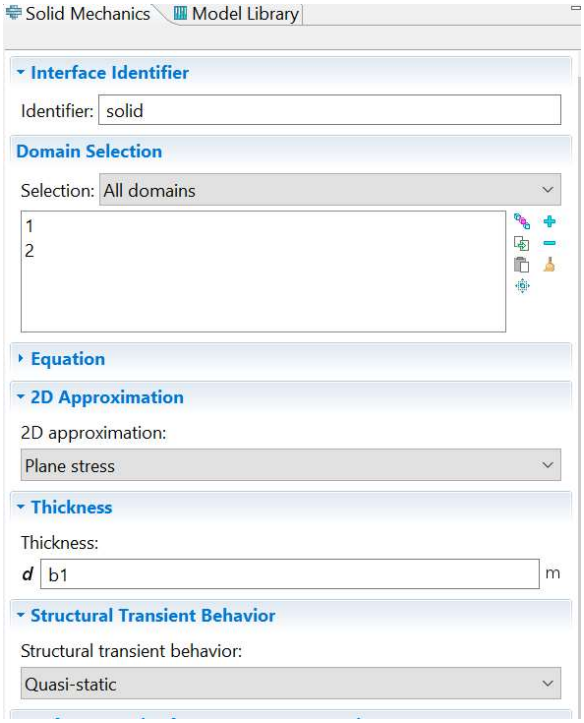


Figure 58: 2D Cantilever Beam Solid Mechanics Study Application

Meshing is the next step for the model, which can be modified depending on the study. To obtain the best mesh, we must conduct a mesh extension study that will show when our solution converges. To start, we will use a linear discretization method. We will use a mapped quadratic mesh on this 2D model. For drastic purposes, we start with a model with two elements across the span and two in the height.

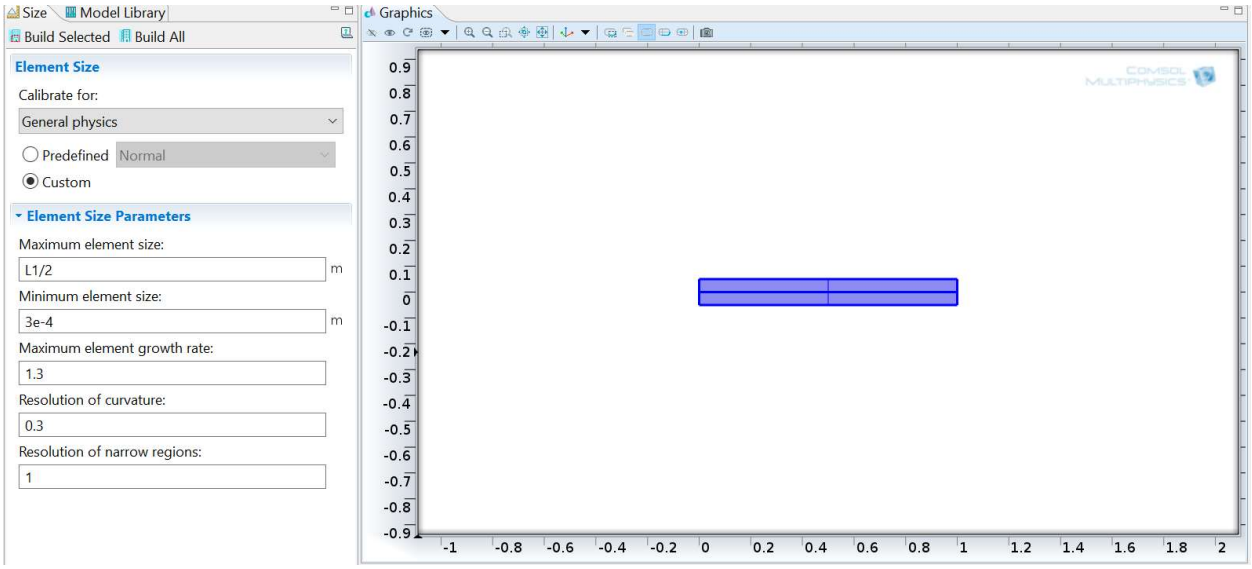


Figure 59: 2D Cantilever Beam Solid Mechanics Coarse Mesh

Plotting the mesh quality on the surface, we can see that this is not an ideal mesh since the elements shown are towards the bottom of the scale.

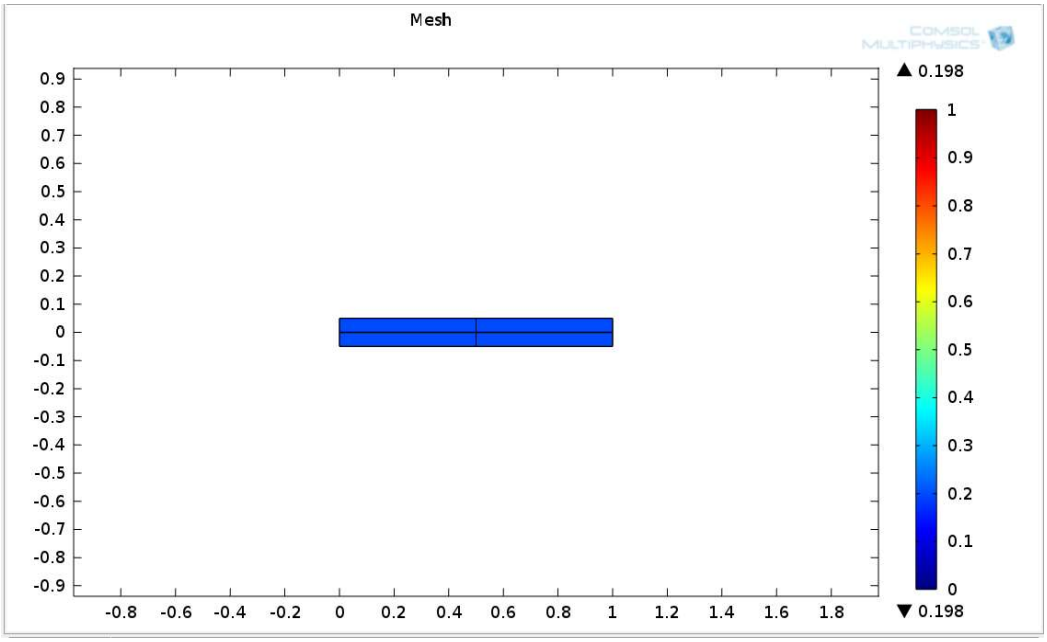


Figure 60: 2D Cantilever Beam Solid Mechanics Coarse Mesh Quality

Below is the displacement field plot in the y direction showing the value obtained.

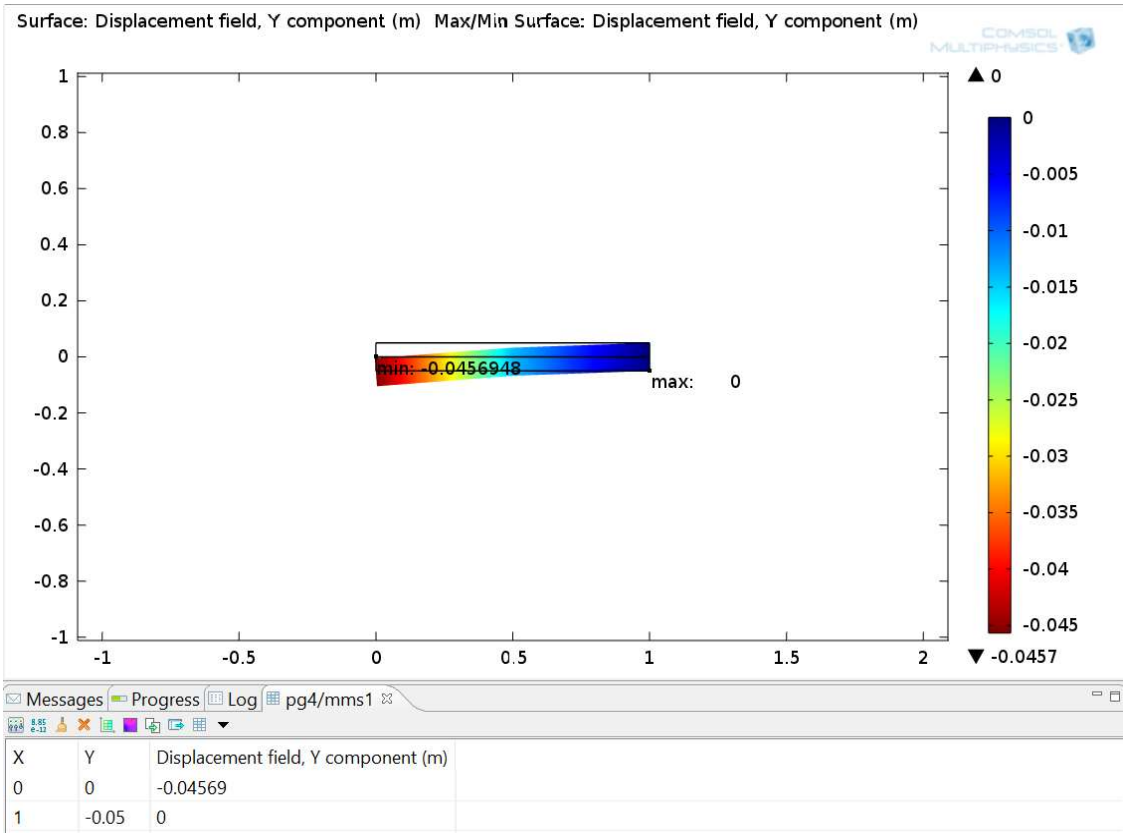


Figure 61: 2D Cantilever Beam Solid Mechanics Coarse Mesh – Displacement Field Plot (Y) with Point Evaluation (Linear Discretization)

This result is far away from the exact solution, so we can start refining the mesh. The results can be seen in the table below for Linear Discretization.

Mesh Study										
Cantilever Beam (Point Load) - 2D										
Discretization - Linear										
Mesh Characteristics		Study		1	2	3	4	5	6	7	8
		Number of Divisions	Height	2	2	2	6	10	50	100	200
			Length	2	10	20	50	100	500	1000	2000
		DOF		34	146	286	1914	6222	151102	602202	2404402
Results		Solve Time(s)		1	1	1	1	1	4	11	52
		Displacement (m)		-0.04569	-0.03489	-0.44095	-0.48536	-0.49203	-0.49456	-0.49469	-0.49477
Maple Solution	-0.4914	Percent Error		90.70%	92.90%	10.27%	1.23%	0.13%	0.64%	0.67%	0.69%

Figure 62: 2D Cantilever Beam Mesh Study Table (Linear Discretization)

For comparison to the coarse mesh, we can use a quadratic discretization. The mesh will look the same and the element quality is also the same. The displacement results are shown below

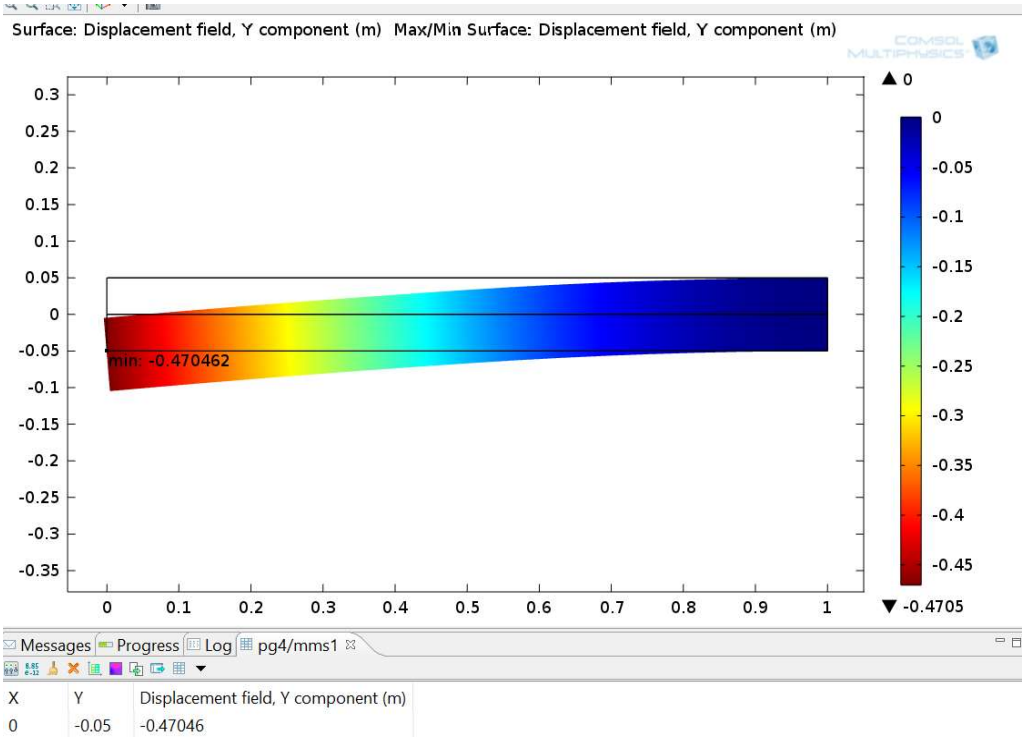


Figure 63: 2D Cantilever Beam Solid Mechanics Coarse Mesh – Displacement Field Plot (Y) with Point Evaluation (Quadratic Discretization)

On the coarse mesh, the answer is already much closer to the exact solution. Below is the mesh extension table for the quadratic discretization.

Mesh Study										
Cantilever Beam (Point Load) - 2D										
Discretization - Quadratic										
Mesh Characteristics		Study		1	2	3	4	5	6	7	8
		Number of Divisions	Height	2	2	2	6	10	50	100	200
			Length	2	10	20	50	100	500	1000	2000
		DOF		86	390	770	5326	17442	427202	1704402	6808802
Results		Solve Time(s)		1	1	1	1	1	13	39	196
		Displacement (m)		-0.47046	-0.49345	-0.49409	-0.49449	-0.49459	-0.49475	-0.49479	-0.49485
Maple Solution	-0.4914	Percent Error		4.26%	0.42%	0.55%	0.63%	0.65%	0.68%	0.69%	0.70%

Figure 64: 2D Cantilever Beam Mesh Study Table (Quadratic Discretization)

Plotting these two discretization mesh studies is shown below comparing the degrees of freedom found in each mesh and comparing the to displacement in the y direction.

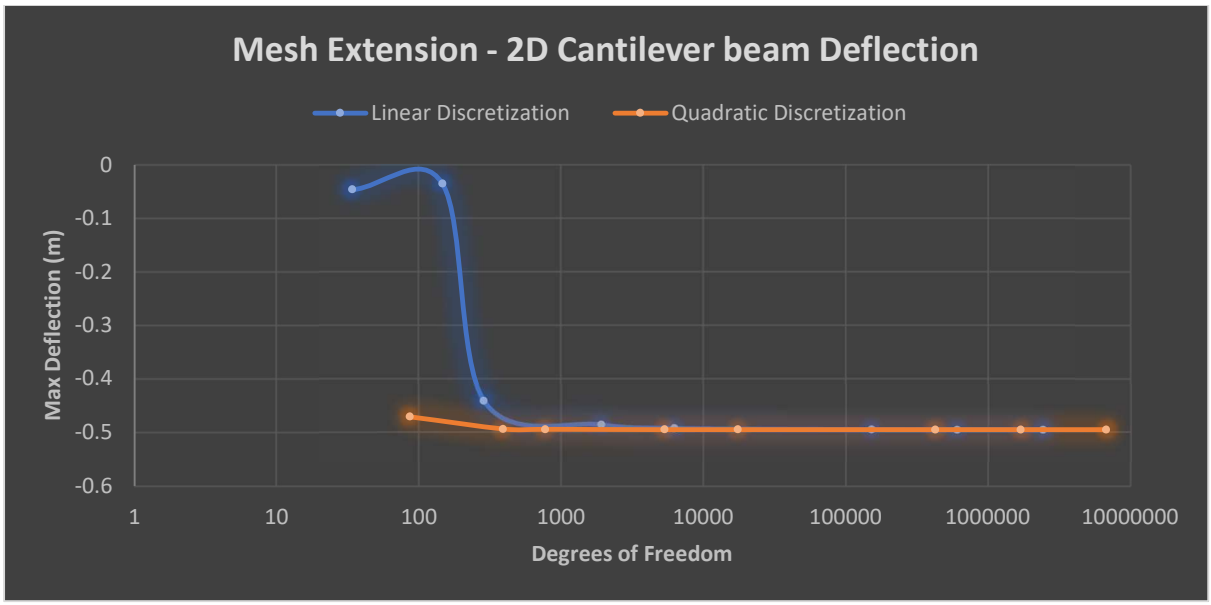


Figure 65: 2D Cantilever Beam Mesh Study Convergence Plot

The converged solution plots from COMSOL are below

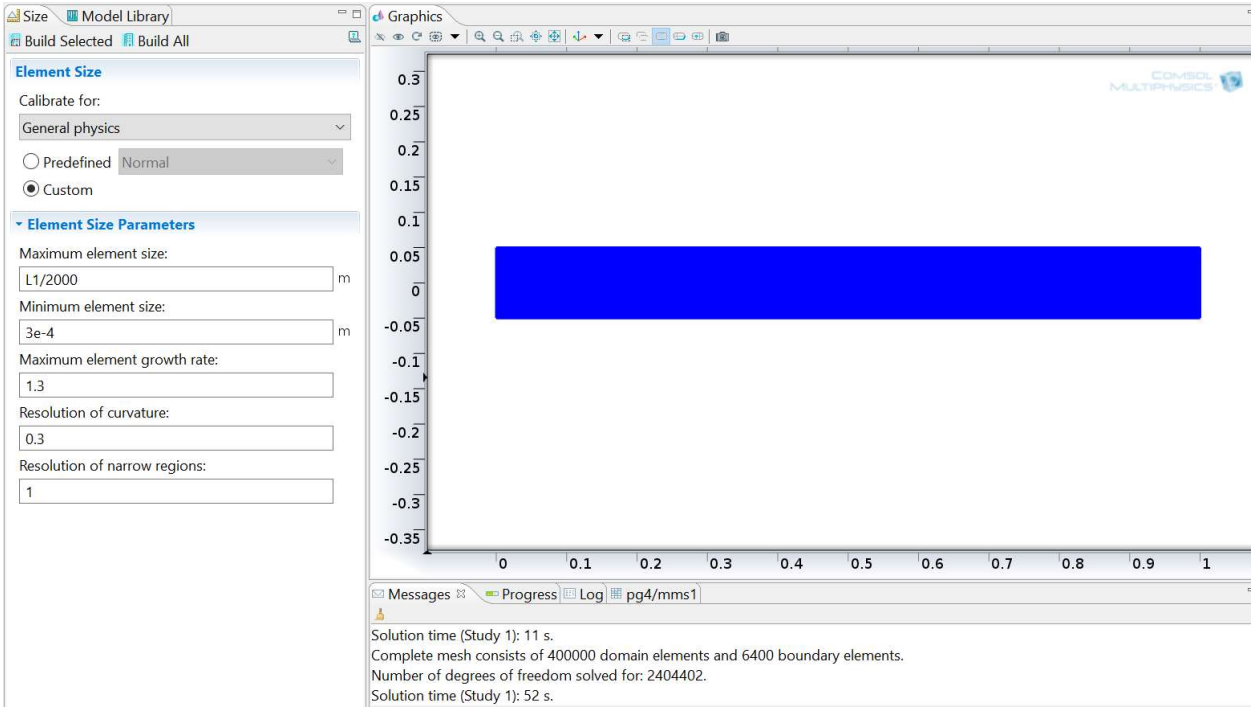


Figure 66: 2D Cantilever Beam Fine Mesh Plot

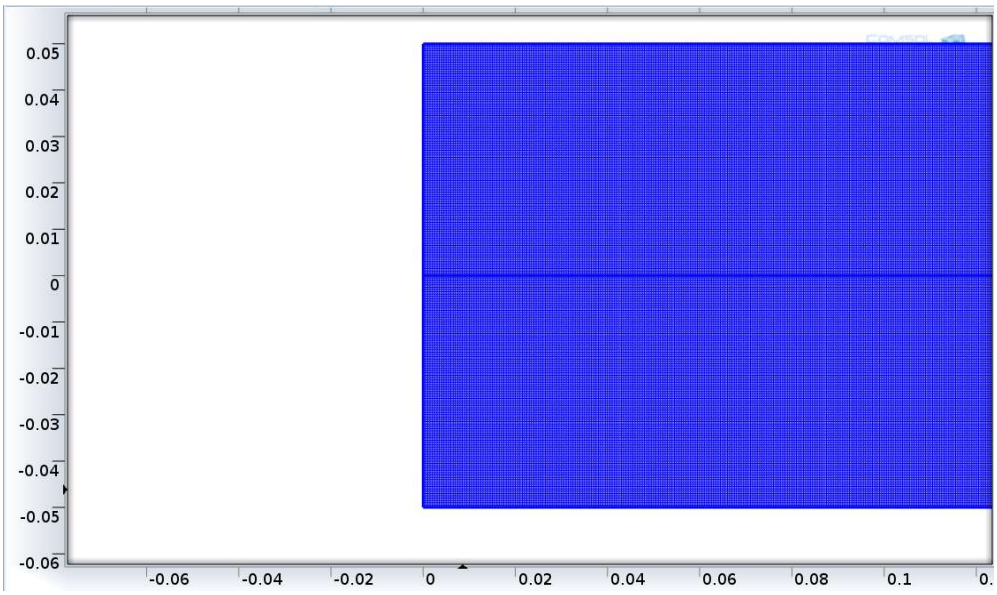


Figure 67: 2D Cantilever Beam Fine Mesh Plot (Zoom)

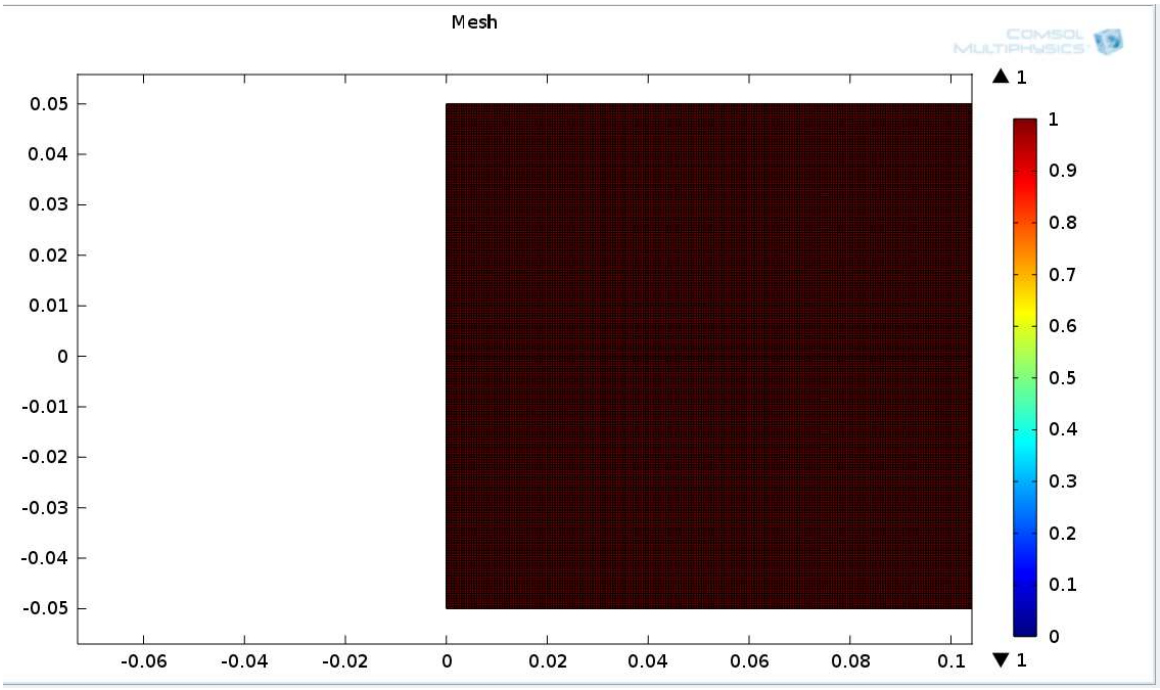


Figure 68: 2D Cantilever Beam Fine Mesh Quality Plot (Zoom)

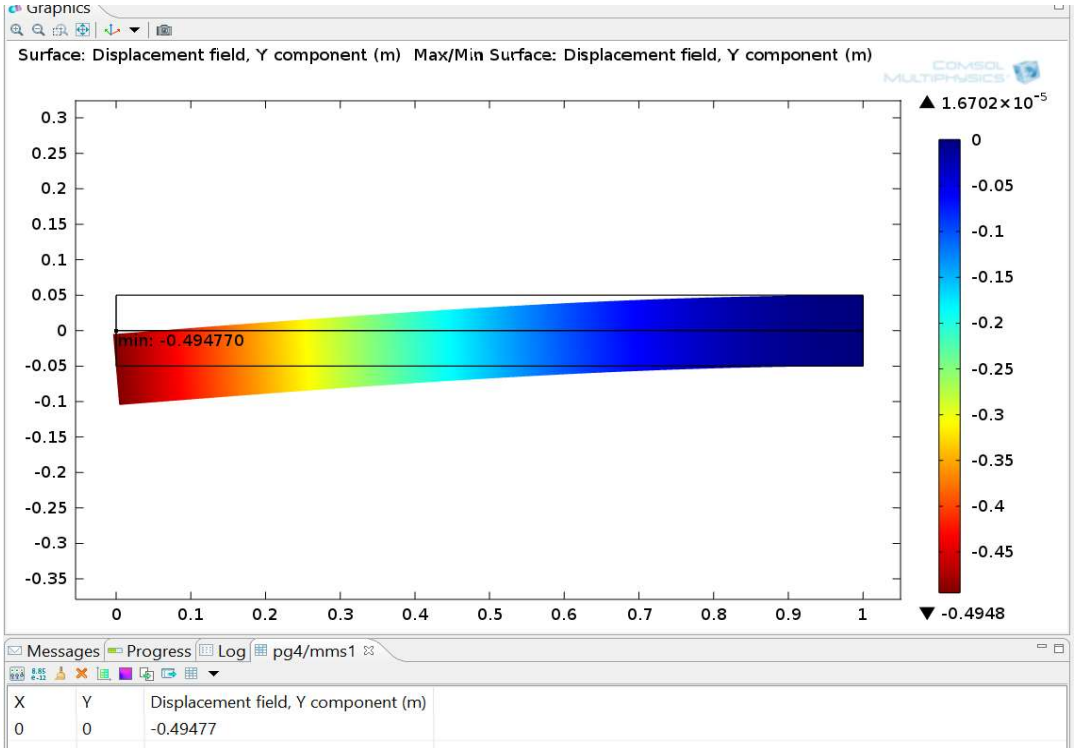


Figure 69: 2D Cantilever Beam Fine Mesh Displacement Field Plot (Y) and Point Evaluation (Linear Discretization)

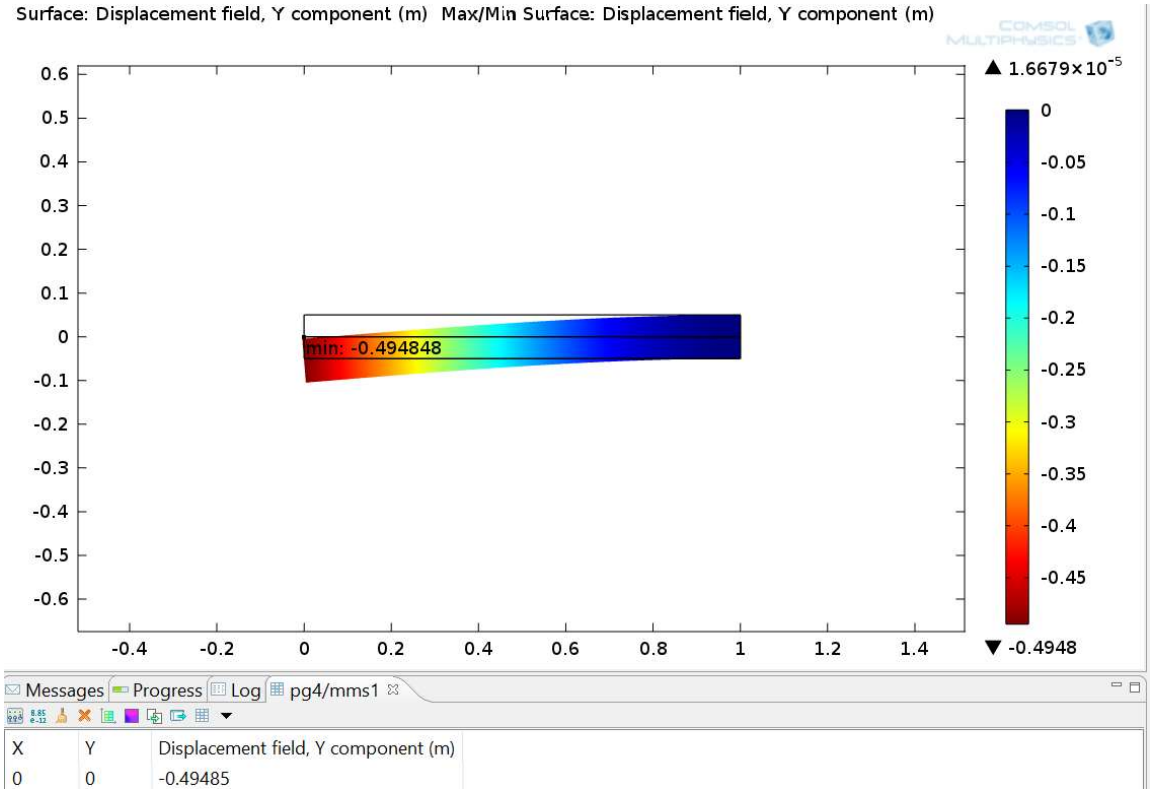


Figure 70: 2D Cantilever Beam Fine Mesh Displacement Field Plot (Y) and Point Evaluation (Quadratic Discretization)

At the boundary is where we have our fixed end, so of interest is to distribute a finer mesh towards the boundary which looks as below

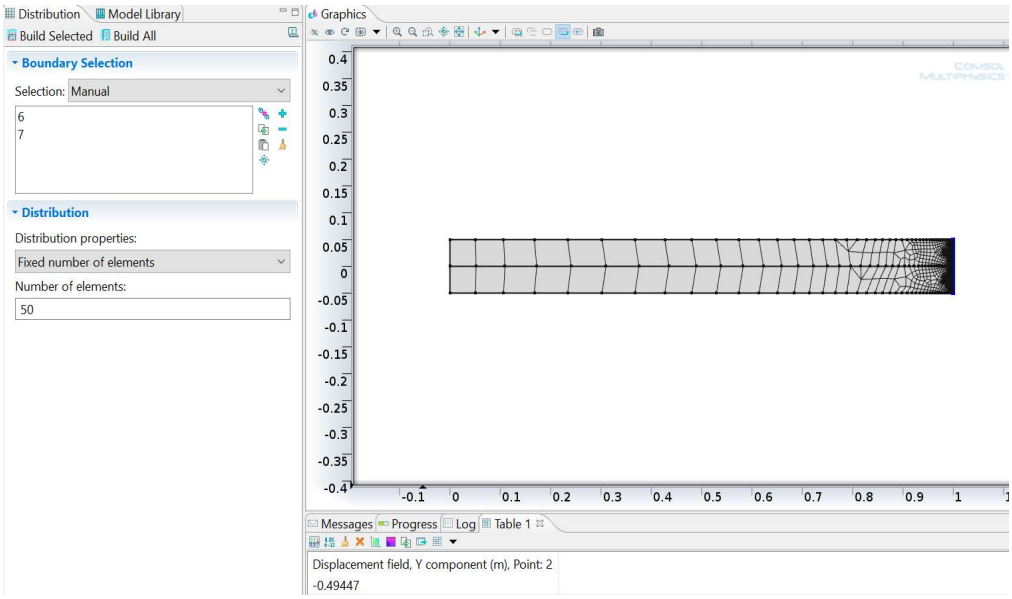


Figure 71: 2D Cantilever Beam Quad Mesh (Distributed Refinement to Fixed End)

Computing the solution, we can view a 2D plot of the von Mises Stress in N/m^2 and the beam being deflected at the end. As expected, the stresses are at a max on the top and bottom surface and fade toward the free end of the beam.

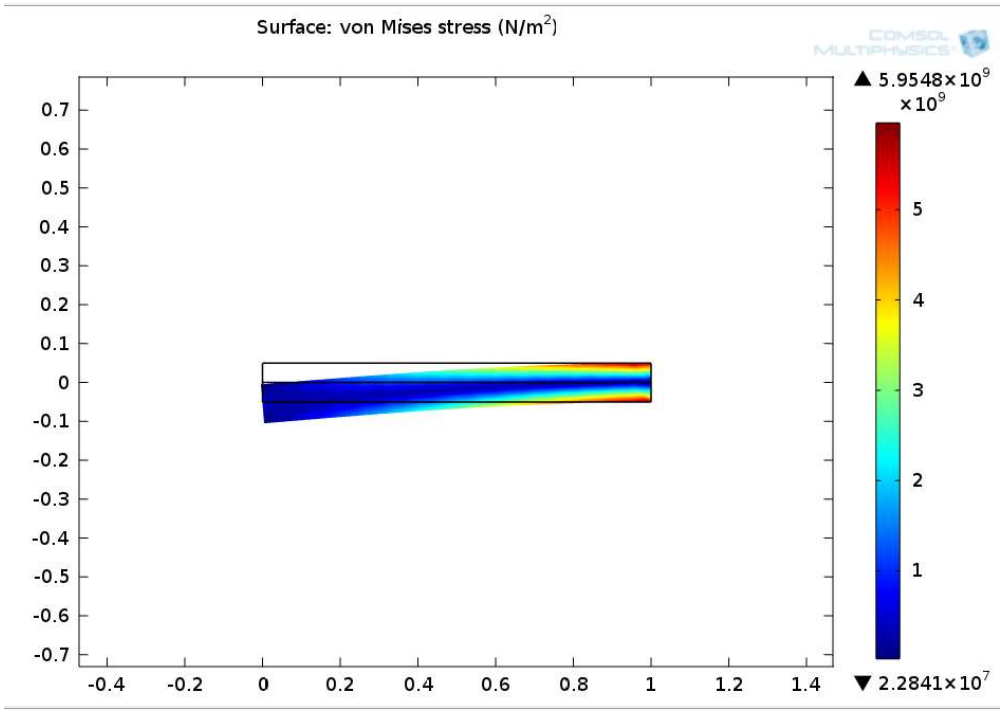


Figure 72: 2D Cantilever Beam von Mises Stress Plot

As seen in the displacement plots above, the deflection is most apparent at the left end and no deflection at the fixed end, as expected.

Finally, for more comparison, we can plot the displacement in a line graph by selecting the center line of the model in a 1D plot.

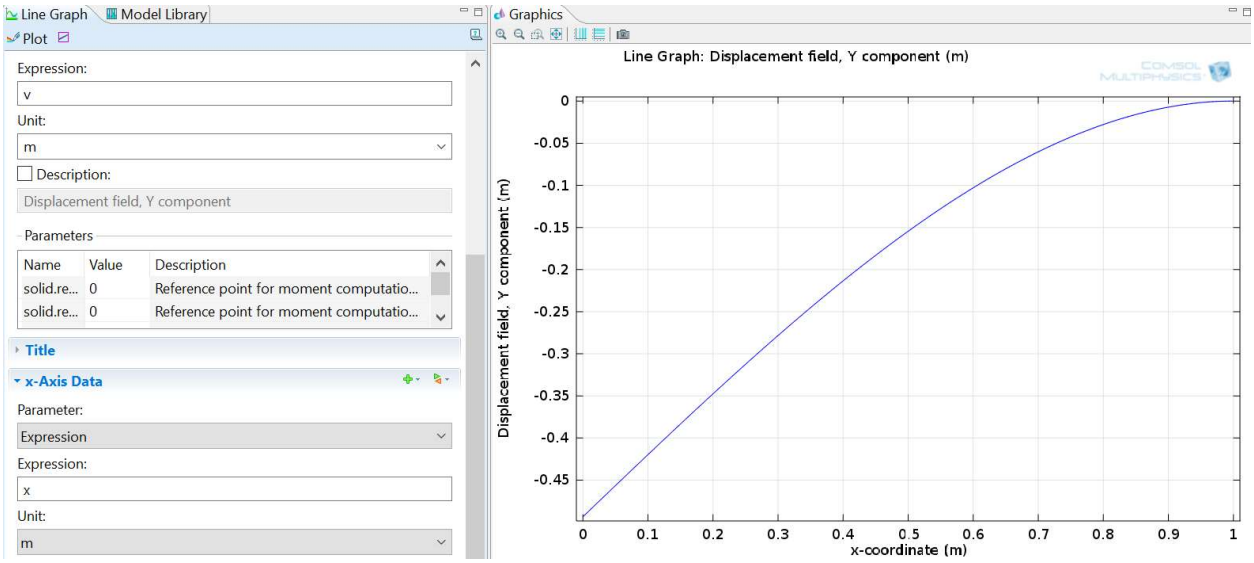


Figure 73: 2D Cantilever Beam Displacement Field (Y) Point Evaluation

Moving to the 3D cantilever beam evaluation in COMSOL, we start by selecting a 3D workspace and set the same parameters. We will build the model in the same way, but the thickness will now be considered in the model as shown below

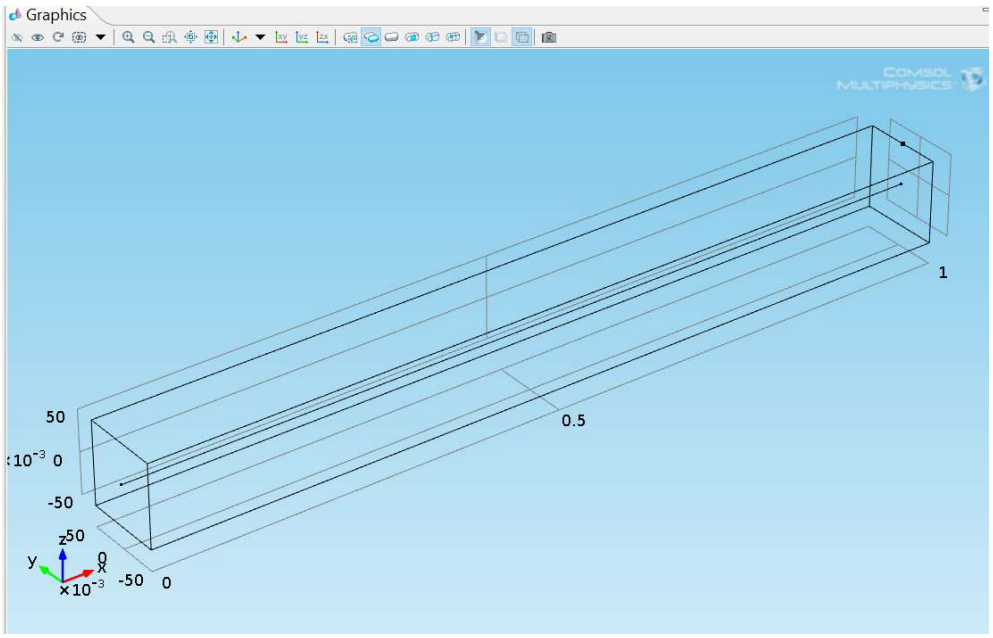


Figure 74: 3D Cantilever Beam Model

Setting the point load and the fixed end is similar to the 2D but is shown as below

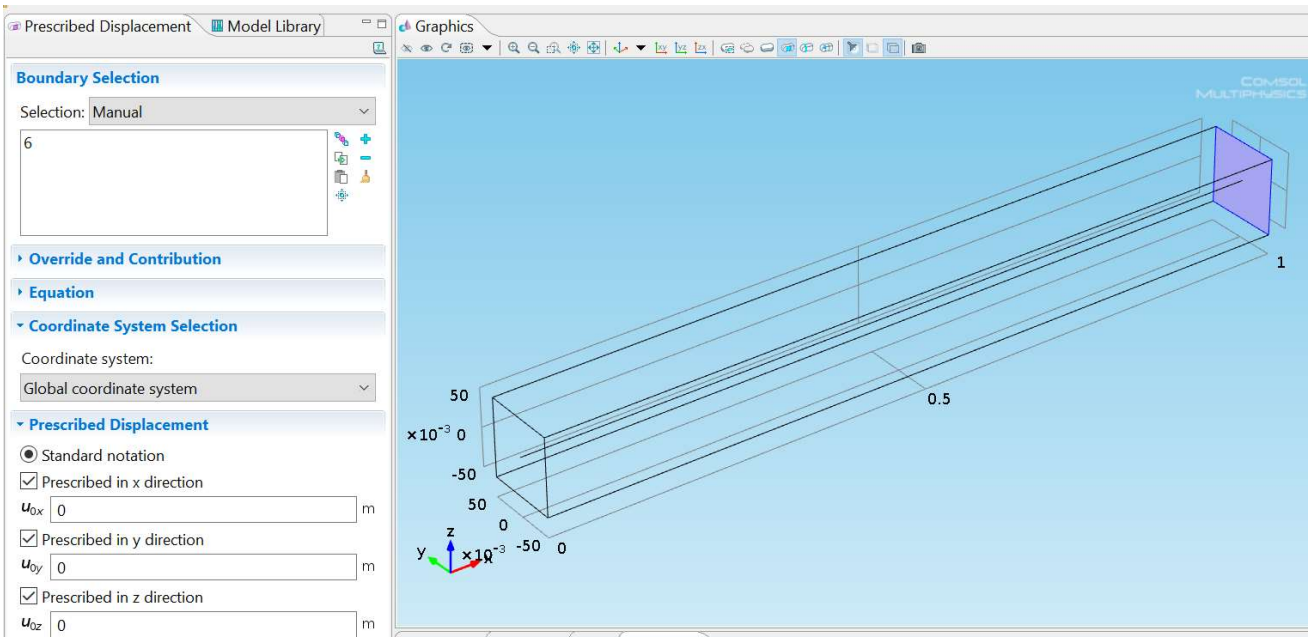


Figure 75: 3D Cantilever Beam Fixed End Application

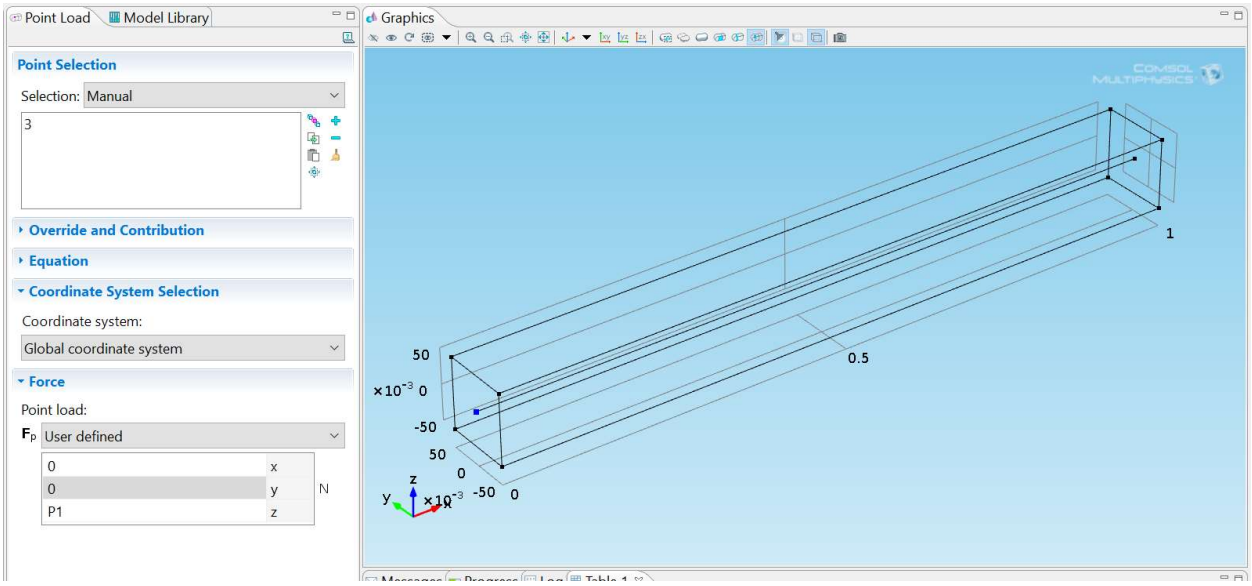


Figure 76: 3D Cantilever Beam Point Load Application (Z Direction)

Meshing is completed in the same manner as the 2D model and to start, we will evaluate the mesh settings with an extension study of the linear, quadratic discretization, as well as a cubic to see a comparison for future studies and their mesh quality (1=best, 0=worst) for comparison and we can see the displacement in the z direction in the table below the models. These mesh are also done using the physics controlled mesh to observe its' usefulness in future studies.

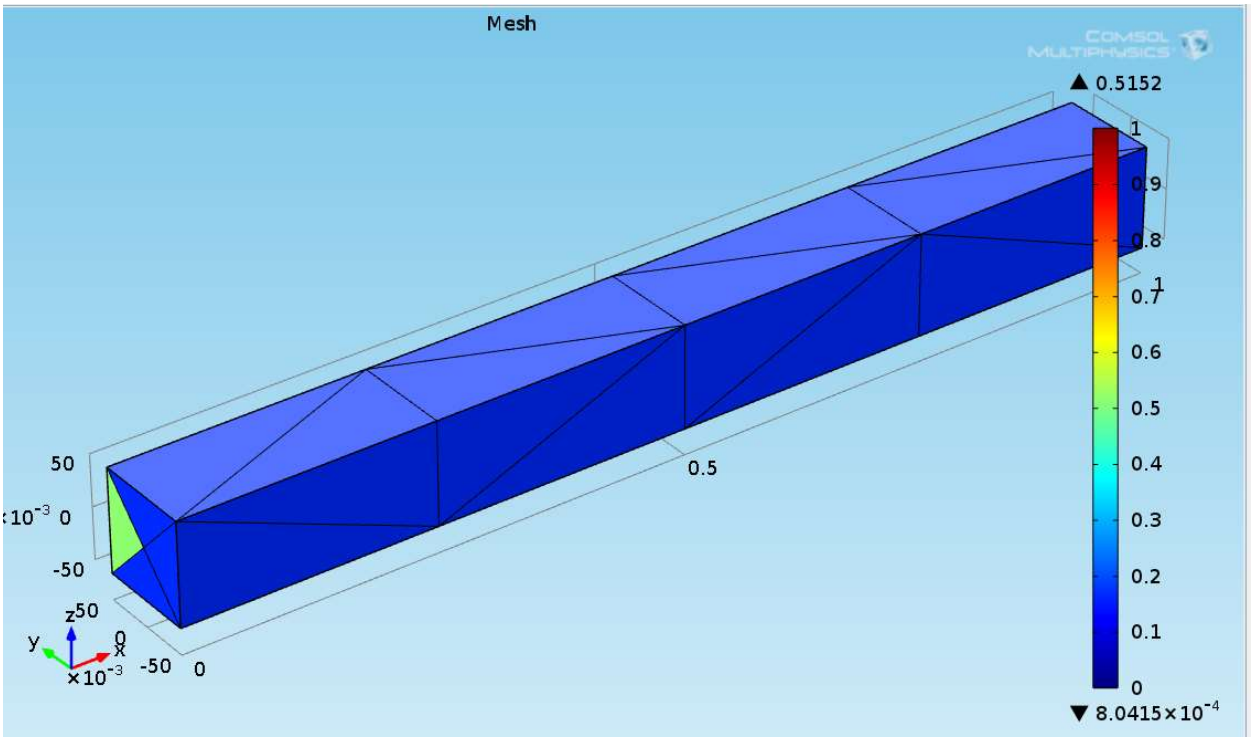


Figure 77: 3D Cantilever Beam Point Load Extra Coarse Mesh Quality (Linear Discretization)

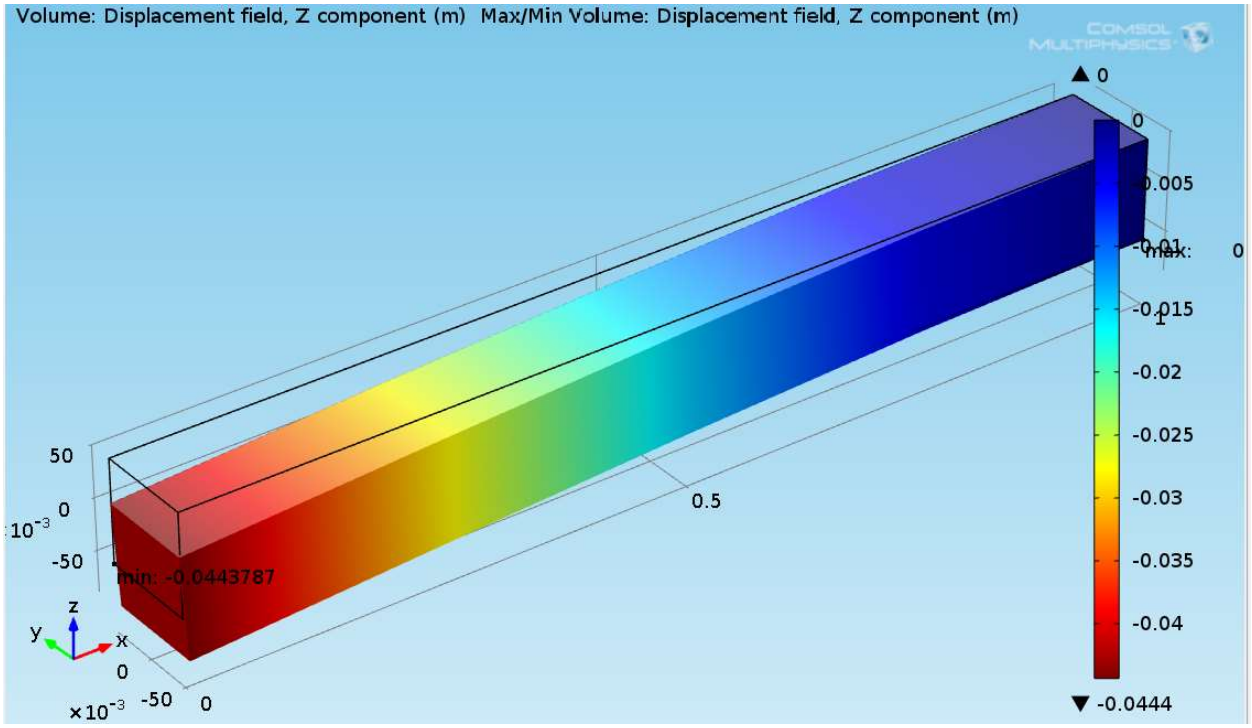


Figure 78: 3D Cantilever Beam Point Load Extra Coarse Mesh Displacement Field (Z) and Point Evaluation (Linear Discretization)

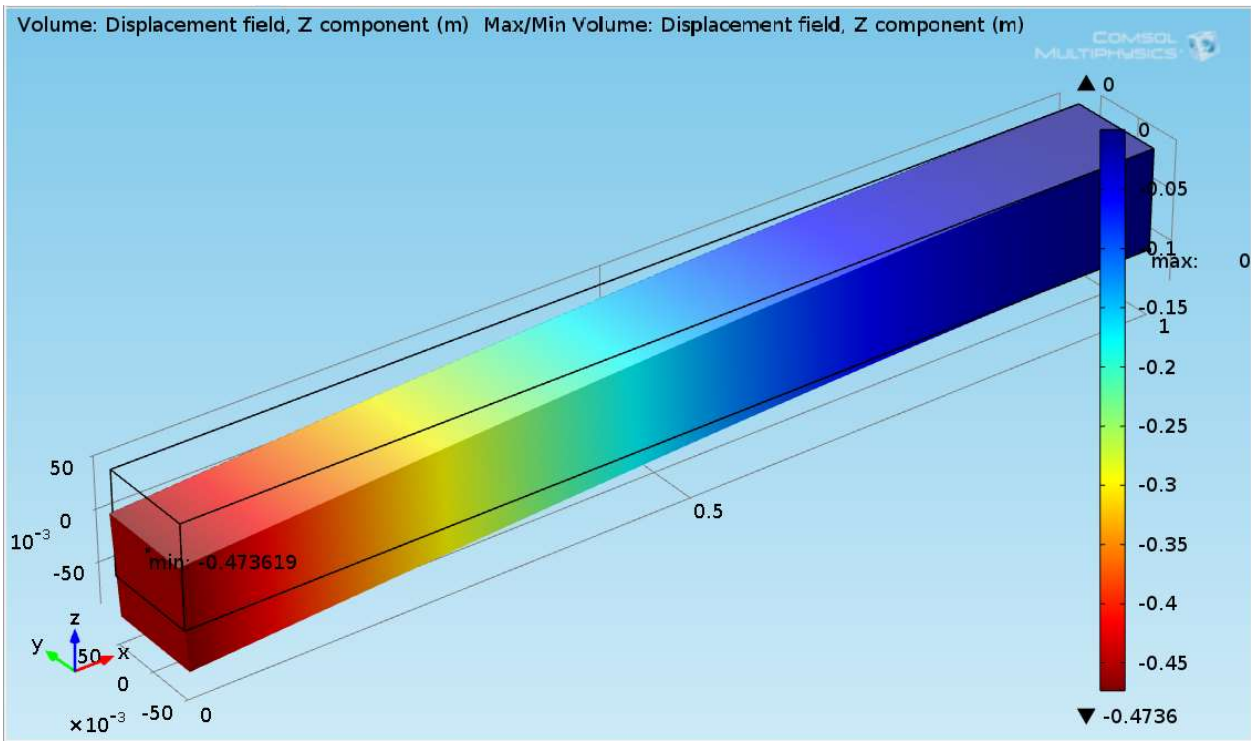


Figure 79: 3D Cantilever Beam Point Load Extra Coarse Mesh Displacement Field (Z) and Point Evaluation (Quadratic Discretization)

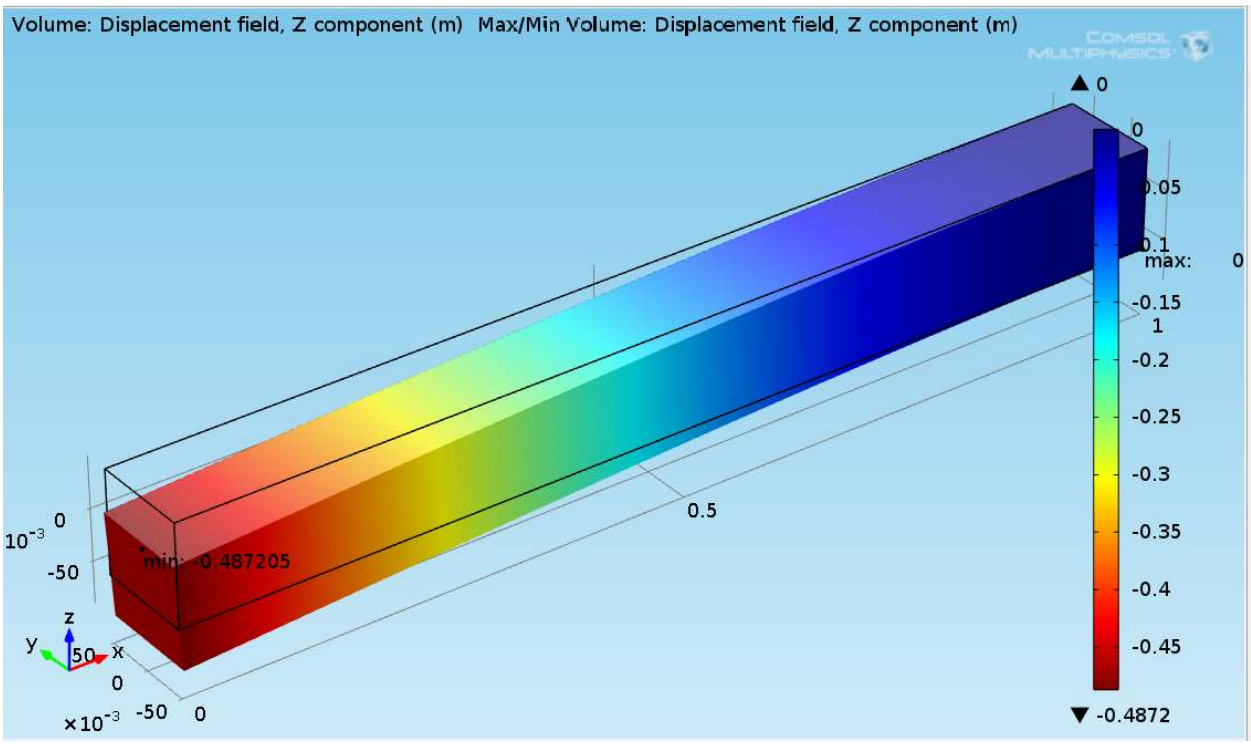


Figure 80: 3D Cantilever Beam Point Load Extra Coarse Mesh Displacement Field (Z) and Point Evaluation (Cubic Discretization)

The mesh study tables for each finite element basis functions can be seen below.

Mesh Study								
Cantilever Beam (Point Load) - 3D								
Discretization - Linear								
Mesh Characteristics		Study	1	2	3	4	5	6
		Physcis Controlled Mesh Tetrahedral	Extra Coarse	Coarse	Normal	Fine	Extra Fine	Extremely Fine
		DOF	78	147	285	447	2865	12027
Results		Solve Time(s)	2	1	1	2	1	4
		Displacement (m)	-0.04438	-0.155	-0.26686	-0.34586	-0.43705	-0.4719
Maple Solution	-0.4914	Percent Error	90.97%	68.46%	45.69%	29.62%	11.06%	3.97%

Figure 81: 3D Cantilever Beam Mesh Study Table (Linear Discretization)

Mesh Study								
Cantilever Beam (Point Load) - 2D								
Discretization - Quadratic								
Mesh Characteristics		Study	1	2	3	4	5	6
		Physcis Controlled Mesh Tetrahedral	Extra Coarse	Coarse	Normal	Fine	Extra Fine	Extremely Fine
		DOF	372	747	1539	2421	19224	87078
Results		Solve Time(s)	2	1	1	1	3	11
		Displacement (m)	-0.47362	-0.48826	-0.48909	-0.49169	-0.49414	-0.49543
Maple Solution	-0.4914	Percent Error	3.62%	0.64%	0.47%	0.06%	0.56%	0.82%

Figure 82: 3D Cantilever Beam Mesh Study Table (Quadratic Discretization)

Mesh Study								
Cantilever Beam (Point Load) - 2D								
Discretization - Cubic								
Mesh Characteristics		Study	1	2	3	4	5	6
		Physcis Controlled Mesh Tetrahedral	Extra Coarse	Coarse	Normal	Fine	Extra Fine	Extremely Fine
		DOF	1044	2139	4485	7053	61068	284307
Results		Solve Time(s)	1	1	2	2	8	58
		Displacement (m)	-0.4872	-0.4917	-0.49182	-0.49312	-0.49659	-0.49849
Maple Solution	-0.4914	Percent Error	0.85%	0.06%	0.09%	0.35%	1.06%	1.44%

Figure 83: 3D Cantilever Beam Mesh Study Table (Cubic Discretization)

We can see how, as in the 2D cantilever study, increasing the finite element basis function order converges much quicker with less computation time with more degrees of freedom at mesh refinement stages.

The most refined mesh and its quality can be viewed below, followed by each discretization method's results. It is seen that the mesh quality is indeed improved, with many elements towards the top of the scale.

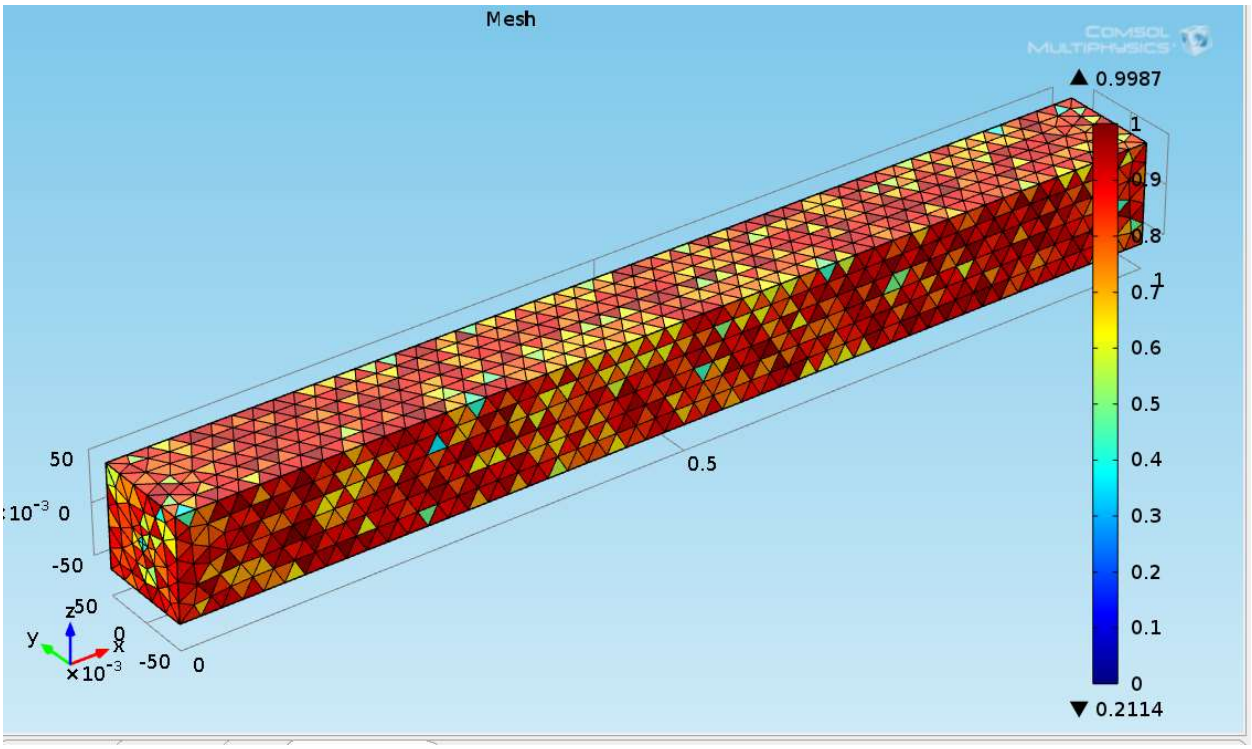


Figure 84: 3D Cantilever Beam Point Load Extremely Fine Mesh Quality

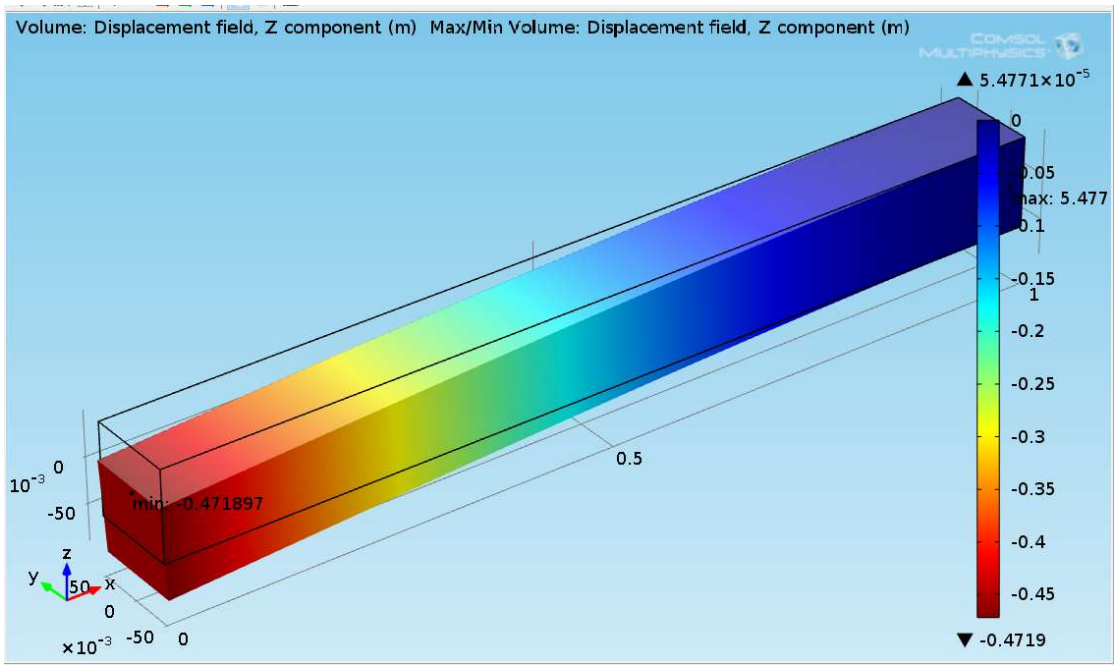


Figure 85: 3D Cantilever Beam Point Load Extremely Fine Mesh Displacement Field (Z) and Point Evaluation (Linear Discretization)

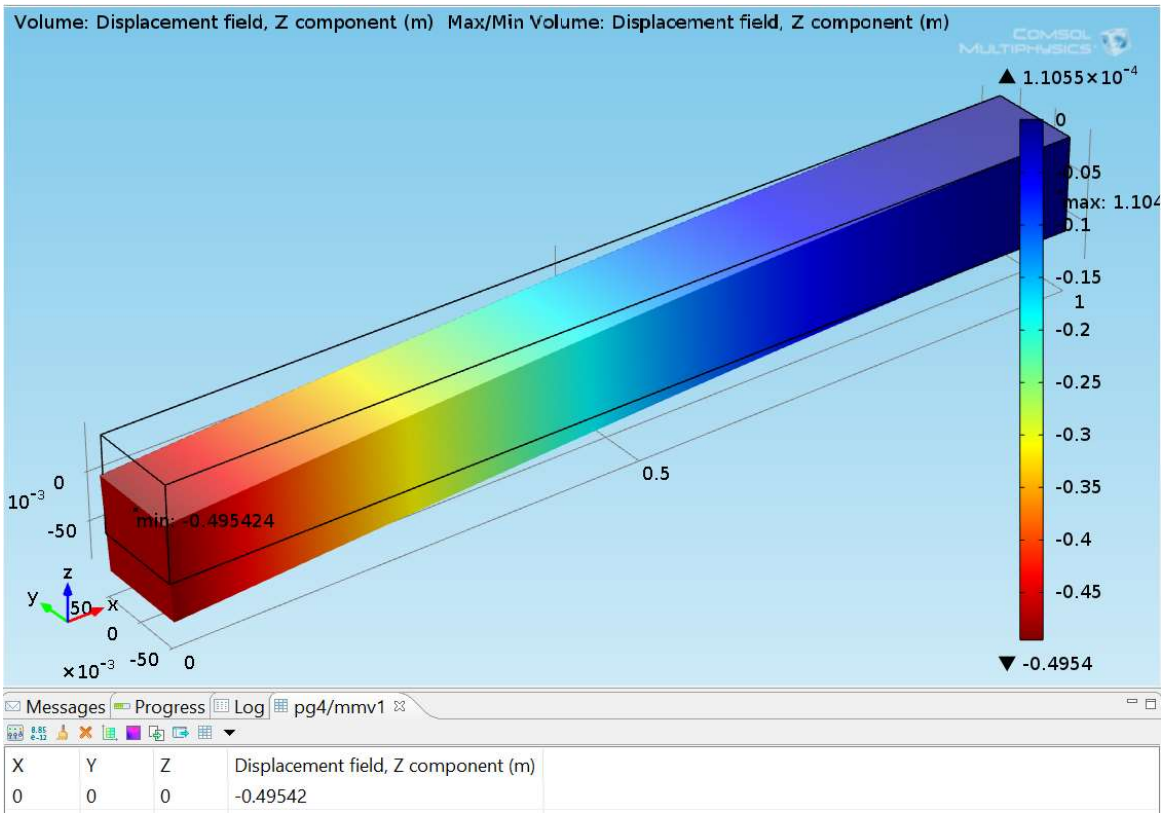


Figure 86: 3D Cantilever Beam Point Load Extremely Fine Mesh Displacement Field (Z) and Point Evaluation (Quadratic Discretization)

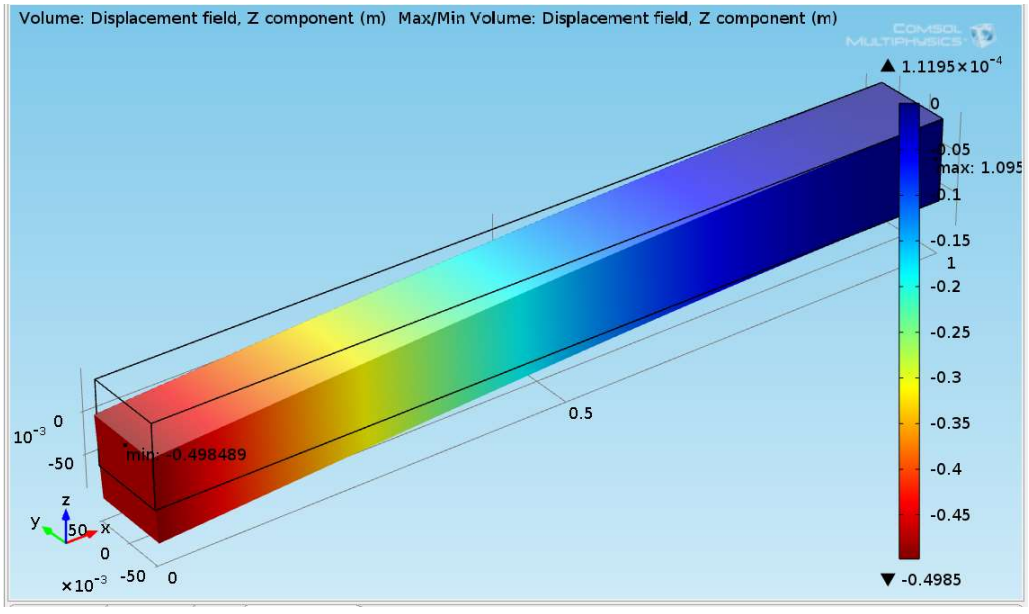


Figure 87: 3D Cantilever Beam Point Load Extremely Fine Mesh Displacement Field (Z) and Point Evaluation (Cubic Discretization)

To finalize the mesh study, the mesh convergence plot is produced

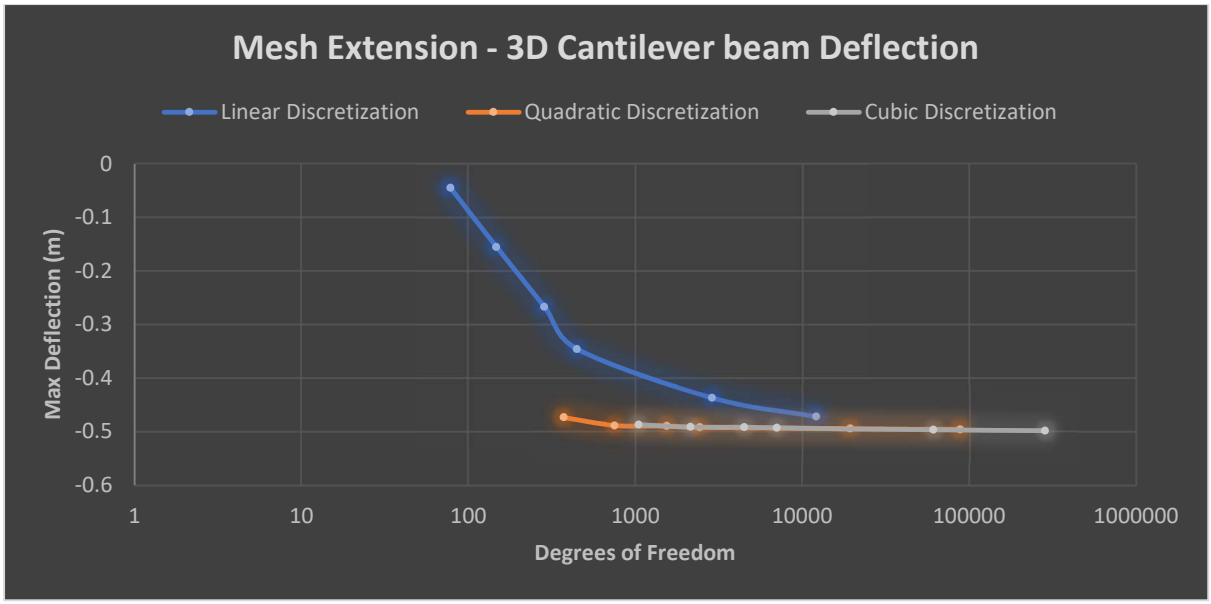


Figure 88: 3D Cantilever Beam Point Load Mesh Convergence Plot

Computing the study, we can view the von Mises stress (N/m^2) in the beam model

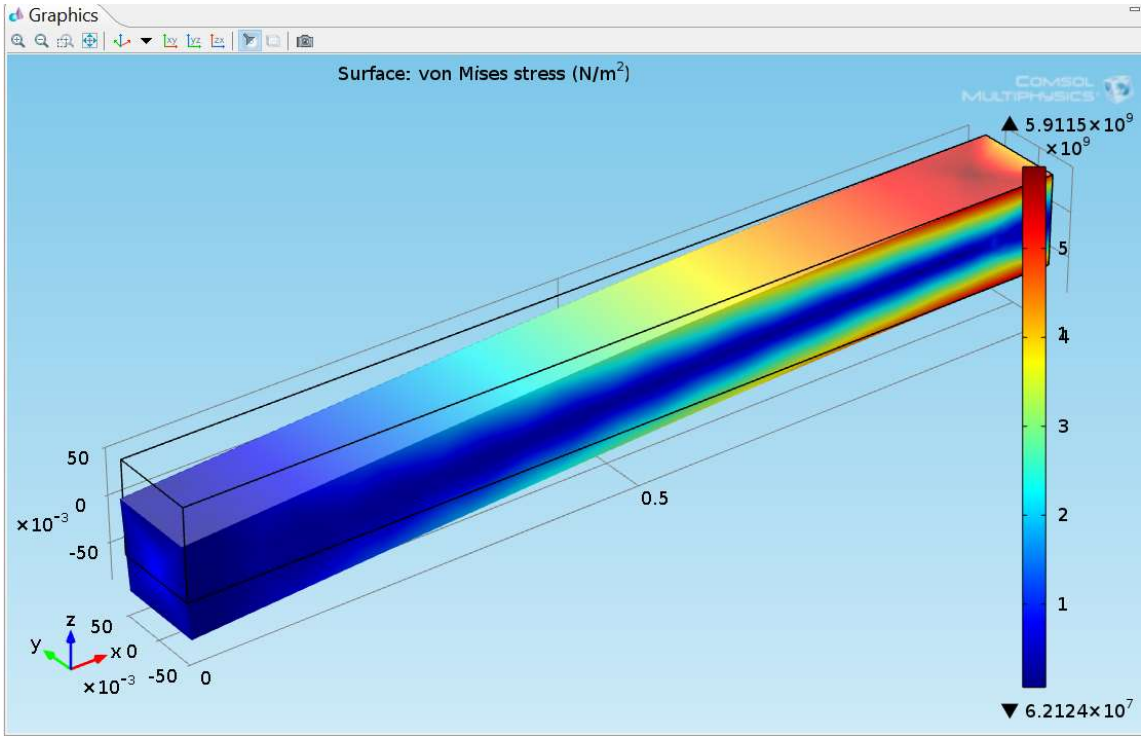


Figure 89: 3D Cantilever Beam von Mises Stress Plot

Evaluating the 1D line plot of the displacement field, we obtain the plot

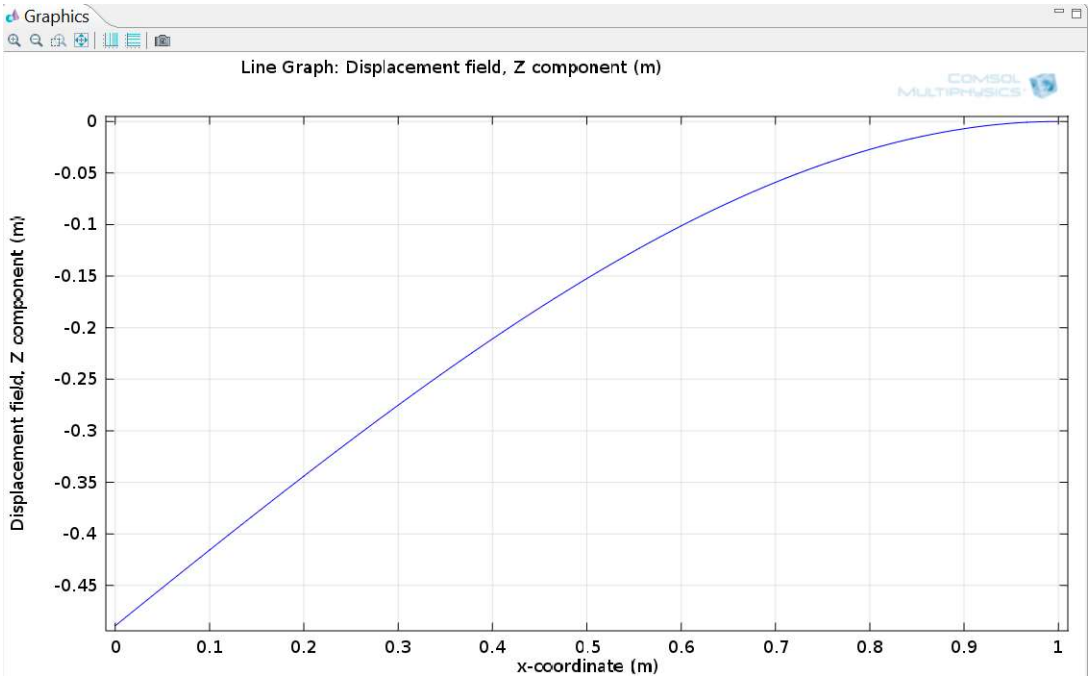


Figure 90: 3D Cantilever Beam, 1D Line Plot Displacement Field (Z)

Finite Element - Inconel 625 Annealed Plate 2D (Simply Supported)

For the transition to a more realistic problem for a simply supported system, Inconel 625 will be used in a simple 1D finite element system as demonstrated above. The beam is scaled up to represent a more realistic system of a beam with a length of one meter. Later in the plate theory, we will use dimensions more suitable to situations that I encounter in the field for rupture discs. The Maple programming of the exact solution using Euler Bernoulli Beam theory is shown below.

```
> restart;
```

Navin_Ian MANE 4240 Milestone 3, 1D Galerkin Approximation

Bending Moment Equation of a beam from Strength of Materials
Incoloy 625 Annealed Plate at RT

```
> E := 8.14e10 : b := .1 : h := .1 : I1 := b*h^3/12 : L := 1 : q := -1e6 : rho := 8440 : nu := 0.278 :
```

```
> M := q*L*x/2 - q*x^2/2;
```

$$M := -500000.0000\,x + 500000.0000\,x^2$$

Differential Equation for the deflection u(x) of the beam

```
> de1 := -E*I1*diff(uex(x), x$2) = M;
```

$$de1 := -678333.3333\,\frac{d^2}{dx^2}uex(x) = -500000.0000\,x + 500000.0000\,x^2$$

```
> u0 := 0 : uL := 0;
```

$$uL := 0$$

Boundary Conditions for a Simply Supported Beam

```
> BCA := uex(0) = u0; BCB := uex(L) = uL;
```

$$BCA := uex(0) = 0$$

$$BCB := uex(1) = 0$$

Exact Solution

```
> S1 := dsolve({de1, BCA, BCB}, uex(x)); assign(%) :
```

$$S1 := uex(x) = -\frac{1250000000}{20349999999}x^4 + \frac{2500000000}{20349999999}x^3 - \frac{1250000000}{20349999999}x$$

```
> s1 := evalf(solve(diff(uex(x), x) = 0, x)) : xMax := s1[1]; uexMax := subs(x = xMax, uex(x));
```

$$xMax := 0.5000000000$$

$$uexMax := -0.01919533169$$

Figure 91: 1D Exact Solution Simply Supported Beam – Inconel 625 Annealed Plate RT

Now to move to a 2D model, we will consider a beam that looks like drawn below with nodes spanning in both directions and the model being made up of elements.

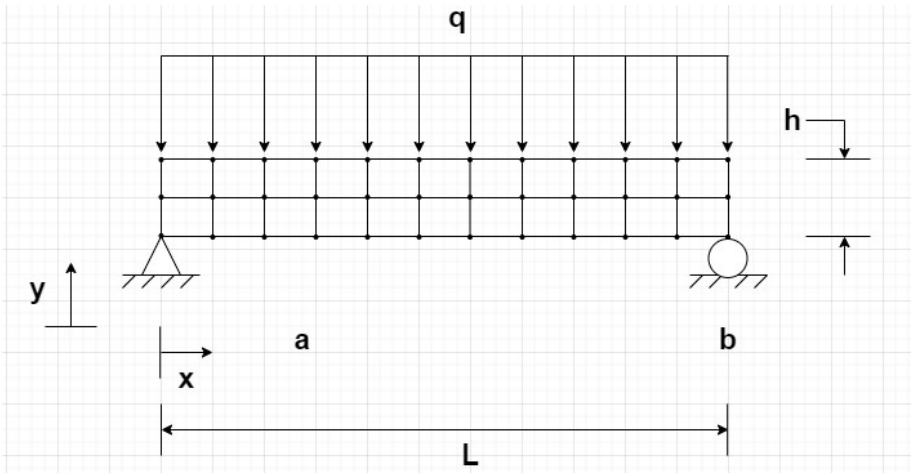


Figure 92: 2D Simply Supported Beam Visual Layout (Nodal Network/Elements/Mesh)

Utilization of Timoshenko's Beam Thoery and Theory of Elasticity, we can determine the exact solution of the 2D beam and compare to exact solution of the Eueler Bernoulli beam theory in 1D.

The beam that we are considering is subject to a uniform transverse distributed load and axial effects are not considered. The cross section is also considered to be very thin, in which case we have plane stress. Plane stress is applied to plates with thin cross sections (z-direction) in comparison to the length (x-direction) and height (y-direction). Referencing a cartesian coordinate system of x,y,z, we obsever zero stress in the z direction. We also observe that z strain is non-zero.

$$\sigma_z = 0, \tau_{xz} = 0, \tau_{yz} = 0 \tag{1.55}$$

$$\gamma_{xz} = 0, \gamma_{yz} = 0 \tag{1.56}$$

In other words, we assume that these components listed in 1.55 and 1.56 do not depend on z or vary through the thickness.

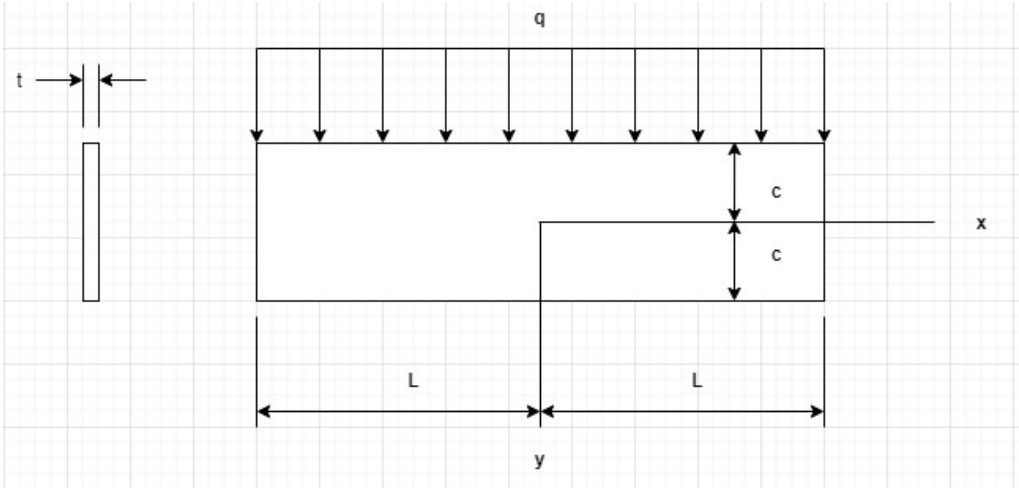


Figure 93: Simply Supported Plate with Uniform Load

The solution can be obtained though polynomials for two dmensional problems when using Timoshenko [13]. The differential equation is as shown as derived from the biharmonic equation

$$\nabla^4 \Phi = 0 \tag{1.57}$$

$$\frac{\partial^4 \Phi}{\partial x^4} + 2\frac{\partial^4 \Phi}{\partial x^2 \partial y^2} + \frac{\partial^4 \Phi}{\partial y^4} = 0 \tag{1.58}$$

We consider the following conditions at the edges of the beam

$$(\tau_{xy})_{y=\pm c} = 0, (\sigma_y)_{y=+c} = 0, (\sigma_y)_{-c} = -q, \int_{-c}^{c} \tau_{xy} dy = \pm qL\,, \int_{-c}^{c} \sigma_x dy = 0\,, \int_{-c}^{c} \sigma_x y dy = 0 \tag{1.59}$$

Utilizing a stress function to the fifth degree polynomial, we have

$$\Phi_5 = \frac{a_5}{5*4}x^5 + \frac{b_5}{4*3}x^4 y + \frac{c_5}{3*2}x^3 y^2 + \frac{d_5}{3*2}x^2 y^3 + \frac{e_5}{4*3}x y^4 + \frac{f_5}{5*4}y^5 \tag{1.60}$$

And substituting into eq (1.58), we satisfy it when

$$e_5 = -(2c_5 + 3a_5), f_5 = -\frac{1}{3}(b_5 + 2d_5) \tag{1.61}$$

Solving for the components of stress with the soleved for coefficients, we have

$$\sigma_x = \frac{\partial^2 \Phi_5}{\partial y^2} = \frac{c_5}{3}x^3 + d_5 x^2 y - (2c_5 + 3a_5)x y^2 - \frac{1}{3}(b_5 + 2d_5)y^3 \tag{1.62}$$

$$\sigma_y = \frac{\partial^2 \Phi_5}{\partial x^2} = a_5 x^3 + b_5 x^2 y + c_5 x y^2 + \frac{d_5}{3}y^3 \tag{1.63}$$

$$\tau_{xy} = -\frac{\partial^2 \Phi_5}{\partial x^2 \partial y^2} = \frac{b_5}{3}x^3 - c_5 x^2 y - d_5 x y^2 + \frac{1}{3}(2c_5 + 3a_5)y^3 \tag{1.64}$$

Considering all coefficients are equation to zero except d5, then we obtain simplified equations of

$$\sigma_x = d_5(x^2 y - \frac{2}{3}y^3) \tag{1.65}$$

$$\sigma_y = \frac{d_5}{3}y^3 \tag{1.66}$$

$$\tau_{xy} = -d_5 x y^2 \tag{1.67}$$

In this problem, we rid of the tensile stresses along side of y=c by inserting a compresion sigma y = a2 and stresses sigmay=b3*y and taoxy = -b3*x. This transforms eq. 1.65, 1.66, and 1.67 to

$$\sigma_x = d_5(x^2 y - \frac{2}{3}y^3) \tag{1.68}$$

$$\sigma_y = \frac{d_5}{3}y^3 + b_3 y + a_2 \tag{1.69}$$

$$\tau_{xy} = -d_5 x y^2 - b_3 x \tag{1.70}$$

Inserting into eq. 1.36, we find the coefficients to be

$$a_2 = -\frac{q}{2}, b_3 = \frac{3}{4} * \frac{q}{c}, d_5 = -\frac{3}{4} * \frac{q}{c^3} \tag{1.71}$$

Plugging these coefficients back into 1.65, 1.66, and 1.67 and observing the moment of ineria is equal to 2c^3/3

$$\sigma_x = -\frac{3}{4} * \frac{q}{c^3}\left(x^2 y - \frac{2}{3}y^3\right) = \frac{q}{2I}\left(x^2 y - \frac{2}{3}y^3\right) \tag{1.72}$$

$$\sigma_y = -\frac{3q}{4c^3}\left(\frac{1}{3}y^3 - c^2 y + \frac{2}{3}c^3\right) = -\frac{q}{2I}\left(\frac{1}{3}y^3 - c^2 y + \frac{2}{3}c^3\right) \tag{1.73}$$

$$\tau_{xy} = -\frac{3q}{4c^3}((c^2 - y^2)x) = -\frac{q}{2I}((c^2 - y^2)x) \tag{1.74}$$

To also satisfy the conditions of the problem, we elminate the couples at the ends of the beam using pure bending sigmax=d3y, sigmay=taoxy=0 and solve for coefficient d3 at both ends of the beam

$$\int_{-c}^{c} \sigma_x y dy = \int_{-c}^{c}\left[-\frac{3}{4} * \frac{q}{c^3\left(L^2 y - \frac{2}{3}y^3\right)} + d_3 y\right] y dy = 0 \tag{1.75}$$

Where

$$d_3 = \frac{3}{4} * \frac{q}{c}\left(\frac{l^2}{c^2} - \frac{2}{5}\right) \tag{1.76}$$

Which yields the stress in the x direction to be

$$\sigma_x = -\frac{3}{4} * \frac{q}{c^3}\left(x^2 y - \frac{2}{3}y^3\right) + \frac{3}{4} * \frac{q}{c}\left(\frac{l^2}{c^2} - \frac{2}{5}\right) y = \frac{q}{2I}(L^2 - x^2)y + \frac{q}{2I}\left(\frac{2}{3}y^3 - \frac{2}{5}c^2 y\right) \tag{1.77}$$

This equation had two parts to the expression. The first portion is recognizable for the stresses in the Euler Bernoulli beam thoery which depends on x, whereas the second portion of the equation does not depend on x and is the correction used in the Timoshenko theory. Equation 1.77 is considered the exact solution when the normal forces at either ends of the beam are distributed as to $\frac{3}{4} * \frac{q}{c^3}\left(\frac{2}{3}y^3 - \frac{2}{5}c^2 y\right)$. When there are no normal forces at either ends of the beam, then we can use eq. 1.77.

Solving for the displacements in the x and y direction; u and v respectively

$$u = \frac{q}{2EI}\left[\left(L^2 x - \frac{x^3}{3}\right) y + x\left(\frac{2}{3}y^3 - \frac{2}{5}c^2 y\right) + v * x\left(\frac{1}{3}y^3 - c^2 y + \frac{2}{3}c^3\right)\right] \tag{1.78}$$

$$v = -\frac{q}{2EI}\left\{\frac{y^4}{12} - \frac{c^2 y^2}{2} + \frac{2}{3}c^3 y + v\left[\frac{(L^2 - x^2) * y^2}{2} + \frac{y^4}{6} - \frac{1}{5}c^2 y^2\right]\right\} - \frac{q}{2EI}\left[\frac{L^2 x^2}{2} - \frac{x^4}{12} - \frac{1}{5}c^2 x^2 + \left(1 + \frac{1}{2}v\right)c^2 x^2\right] + \delta \tag{1.79}$$

When the centroid is located at x=0 and y=0, the displacement in the x direction is zero and the y direction is equal to δ. The neutral axis of the beam is not as expected at the centerline due to the

compressive stress (σy)y=0=-q/2, the centerline has a tensile strain of nu*q/2E which yields (u)y=0=nu*q*x/2E.

The deflection in the y direction becomes

$$(v)_{y=0} = \delta\ -\frac{q}{2EI}\left[\frac{L^2x^2}{2} - \frac{x^4}{12} - \frac{1}{5}\ c^2x^2 + \left(1+\frac{1}{2}\nu\right)c^2x^2\right] \tag{1.80}$$

When we assume that the deflection at the ends of the beam are zero, then the deflection can be calculated as

$$\delta = \frac{5}{24} * \frac{qL^4}{EI}\left[1 + \frac{12}{5} * \frac{c^2}{L^2}\left(\frac{4}{5} + \frac{\nu}{2}\right)\right] \tag{1.81}$$

Where $\frac{5}{24} * \frac{qL^4}{EI}$ is equivalent to the deflection found for Euler with the assumption that the cross sections remain plane during bending. The second term is the correction of Timoshenko that accounts for the shearing force in the beam.

The 2D model is calculated as shown below using Maple

2D Model

```
> restart;
```

Incoloy 625 Annealed Plate at RT

```
> P := -1e6 : L := 0.5 : E := 8.14e10 : h := .1 : b := .1 : II := b*h^3/12 : rho := 8440 : nu := 0.278 : G := E/(2*(1+nu)) : c := h/2 : d := 5/24*P*(L)^4/(E*II)*(1+12/5*c^2/L^2*(4/5+nu/2)); dl := 5/24*P*(L)^4/(E*II);
```

$$d := -0.01962791769$$

$$dl := -0.01919533170 \quad (7)$$

d1 is the deflection derived from elementary analysis, assuming that cross sections of the beam remain plane during bending

d is the correction called the effect of shear force

From Timoshenko Theory of Elasticity Delfection in x, y, and z direction (u,v,w) - 2D Model

```
> u := P/(2*E*II)*(((L)^2*x-x^3/3)*y+x*(2/3*y^3-2/5*c^2*y)+nu*x*(1/3*y^3-c^2*y+2/3*c^3));
```

$$u := -0.7371007370\left(0.25\,x-\frac{1}{3}x^3\right)y-0.7371007370\,x\left(\frac{2}{3}y^3-0.001000000000\,y\right)$$
$$-0.2049140049\,x\left(\frac{1}{3}y^3-0.002500000000\,y+0.00008333333333\right) \quad (8)$$

```
> v := -P/(2*E*II)*(y^4/12-c^2*y^2/2+2/3*c^3*y+nu*((L)^2-x^2)*y^2/2+y^4/6-1/5*c^2*y^2))-P/(2*E*II)*((L)^2*x^2/2-x^4/12-1/5*c^2*x^2+(1+1/2*nu)*c^2*x^2)+d
```

$$v := 0.09557739560\,y^4-0.001023832924\,y^2+0.00006142506140\,y+0.1024570024\left(-x^2+0.25\right)y^2$$
$$+0.09386793610\,x^2-0.06142506140\,x^4-0.01962791769 \quad (9)$$

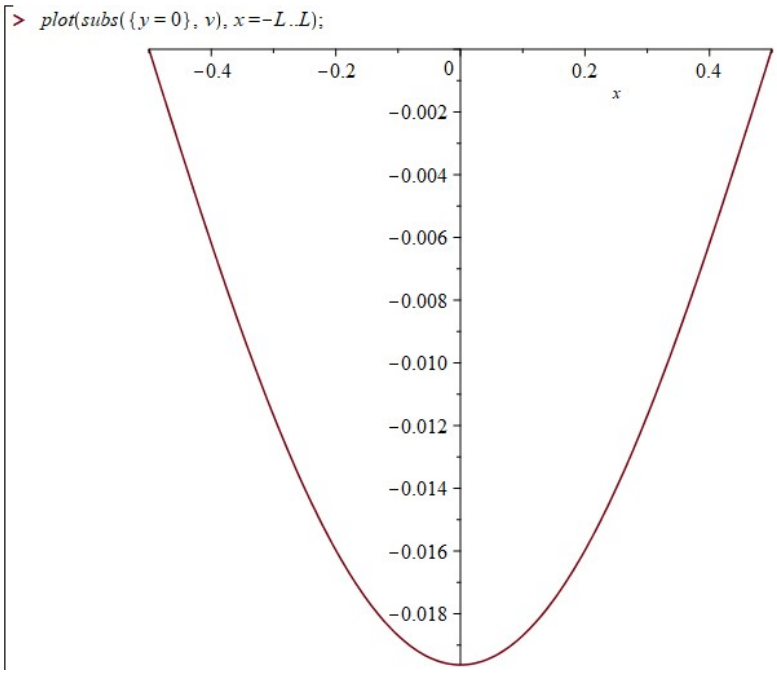
```
> w := 0;
```

$$w := 0 \quad (10)$$

```
> subs({x=0, y=0}, u); subs({x=0, y=0}, v); subs({x=0, y=0}, w);
```

$$-0.$$
$$-0.01962791769$$
$$0 \quad (11)$$

Figure 94: 2D Solution (Timoshenko) Simply Supported Beam – Inconel 625 Annealed Plate RT



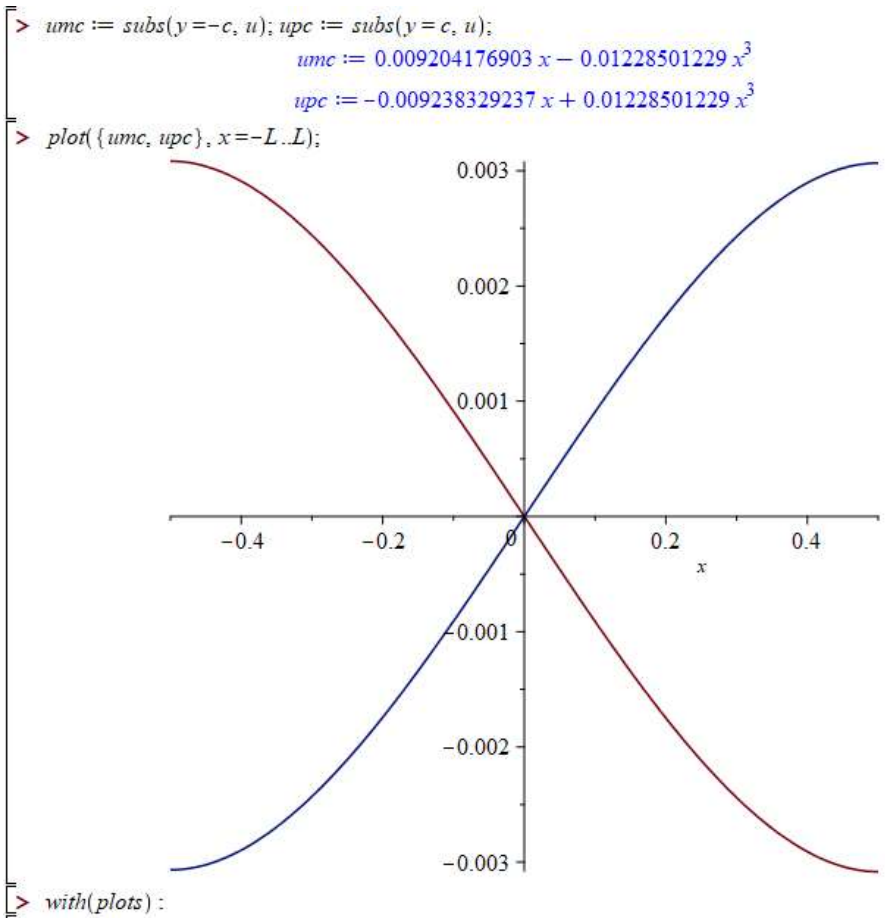


Figure 95: 2D Solution, 1D Deflection Plots - Simply Supported Beam – Inconel 625 Annealed Plate RT

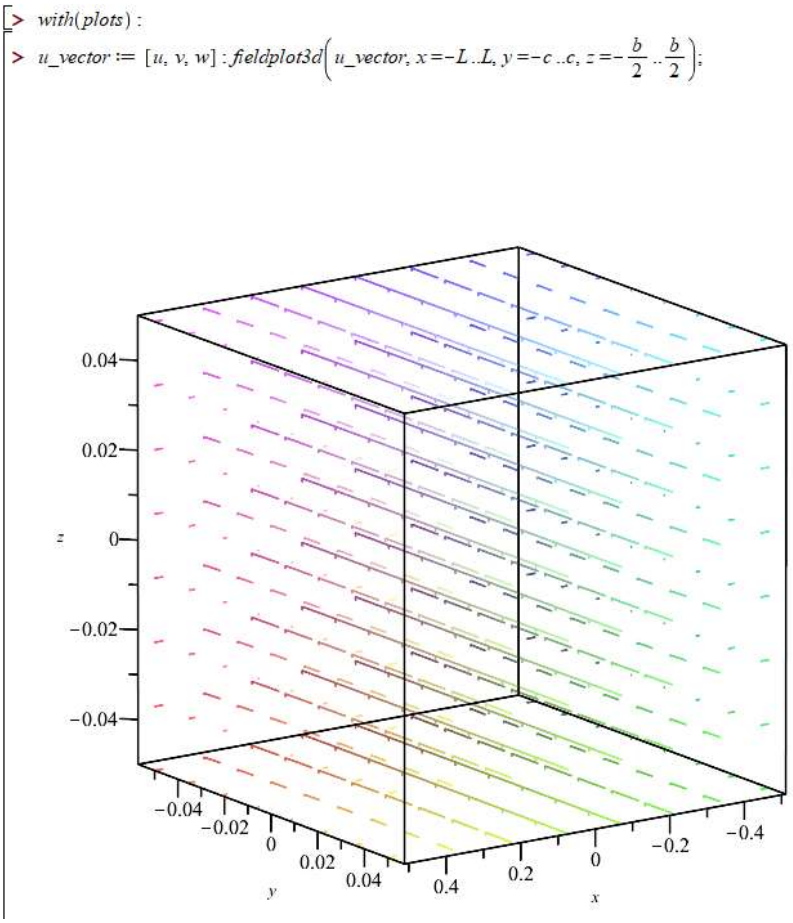


Figure 96: 2D Plots - Simply Supported Beam – Inconel 625 Annealed Plate RT

Further we compare the maple results with a 2Dimensional analysis using finite element software COMSOL which has proven reliable in the 1D evaluations. Set up and analysis can be see below

First the parameters are set using the defined material properties of Inconel 625 and geometry of the beam

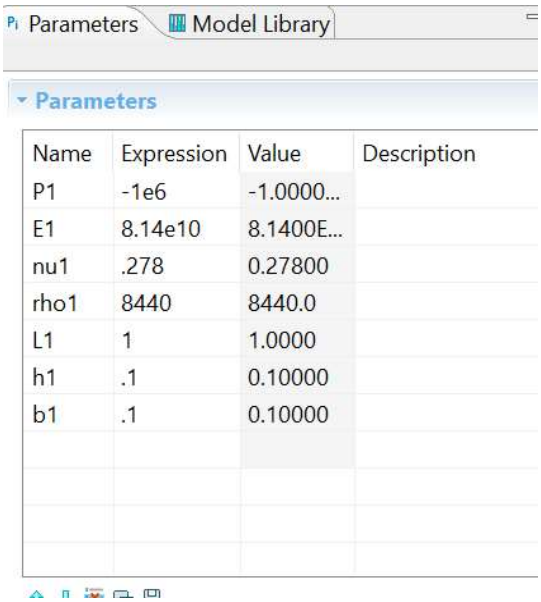


Figure 97: Simply Supported Beam Parameters COMSOL

The parameters can be used to build the model as shown

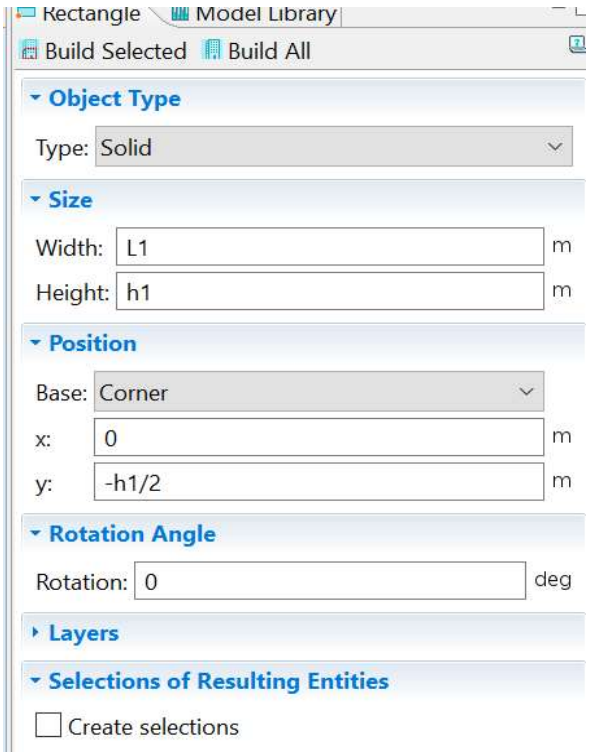


Figure 98: 2D Simply Supported Beam Model Parameters

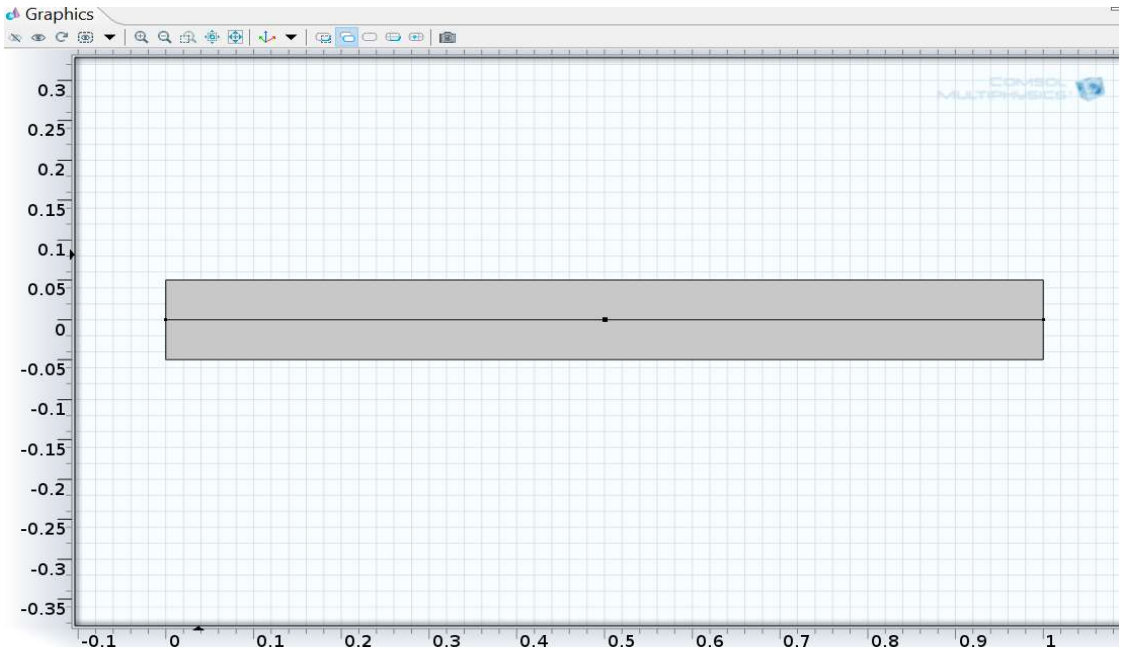


Figure 99: 2D Beam Model

The beam domains need the material properties applied using the parameters set at the beginning

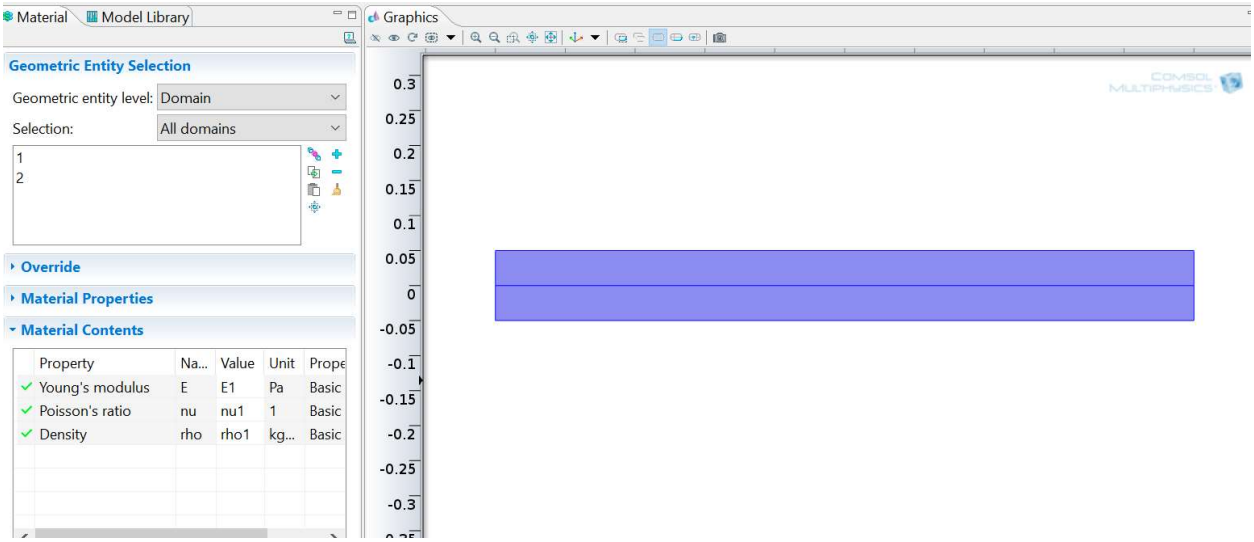


Figure 100: 2D Beam Model Material Parameter Application

We consider plane stress since we are not dependent on the z direction for the thin beam that we consider, but still specify the thickness in the 2D approximation

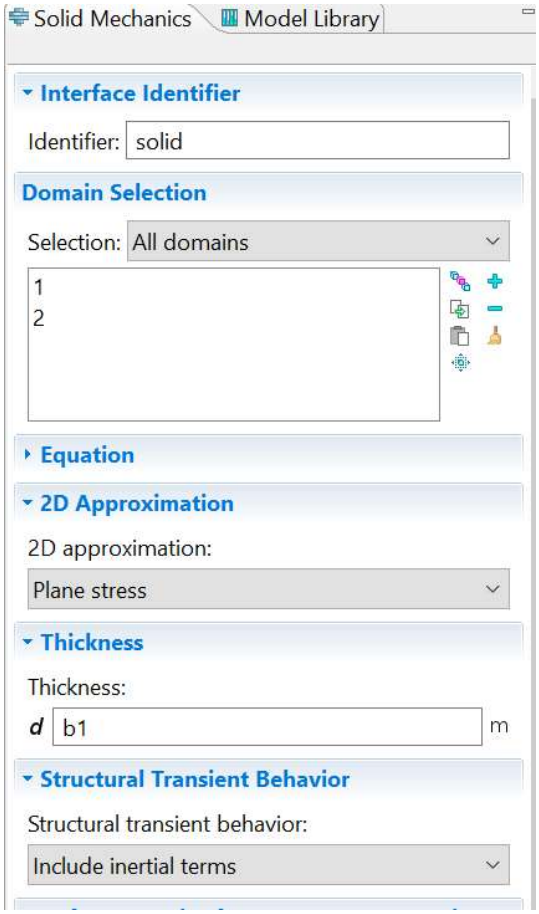


Figure 101: 2D Approximation Solid Mechanics Set-up

Next we apply the boundary conditions to the beam ends to create a simply supported model by using prescribed displacements in the appropriate directions

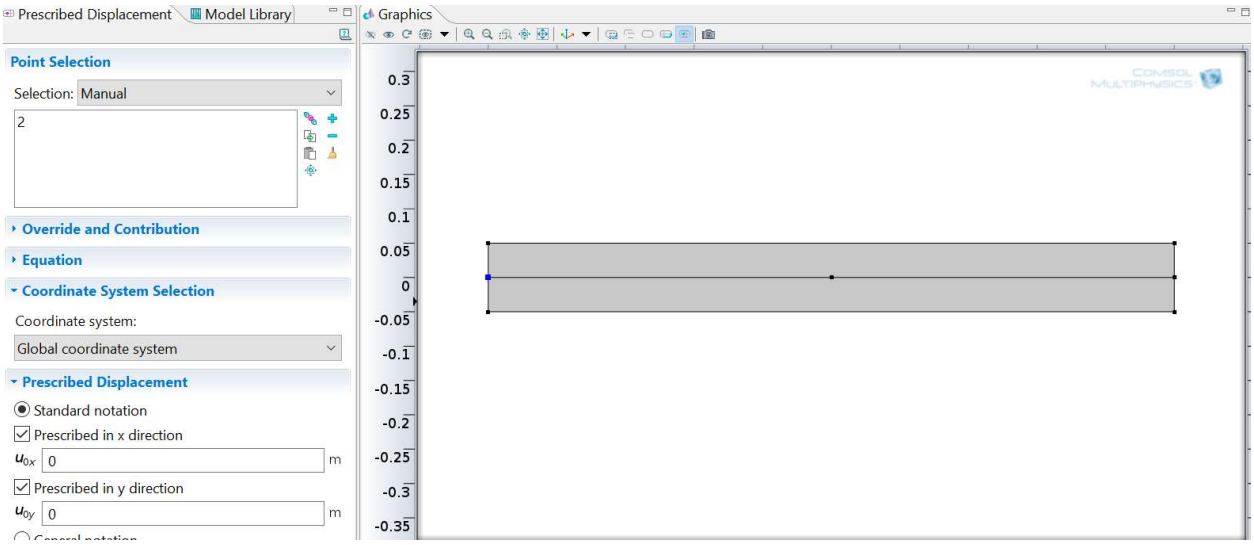


Figure 102: 2D Simply Supported Beam Left End Constraint

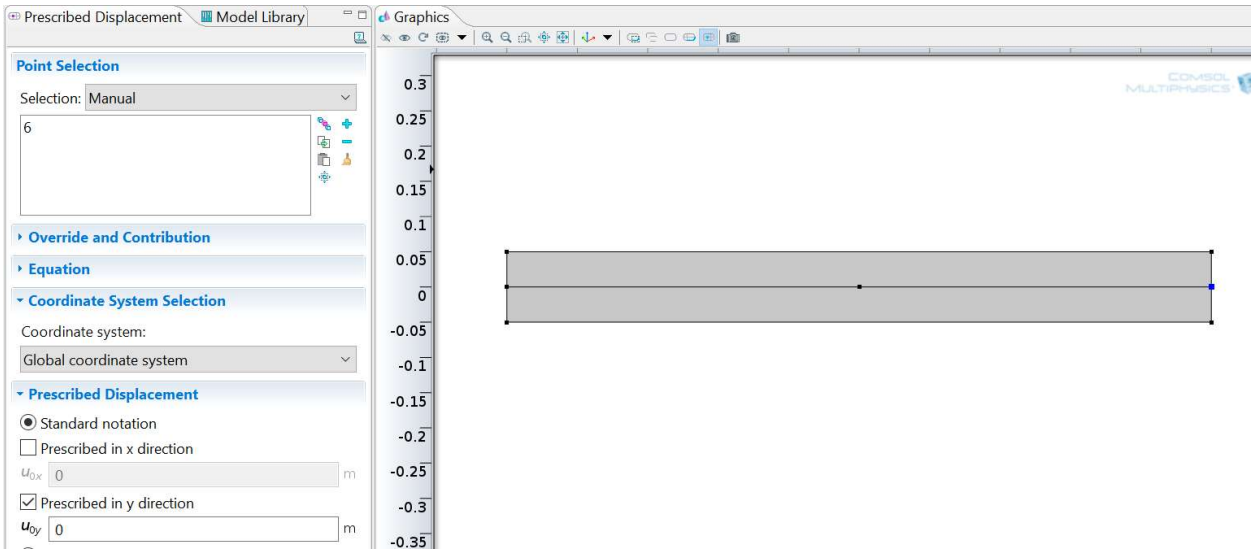


Figure 103: 2D Simply Supported Beam Right End Constraint

The distributed load is then applied

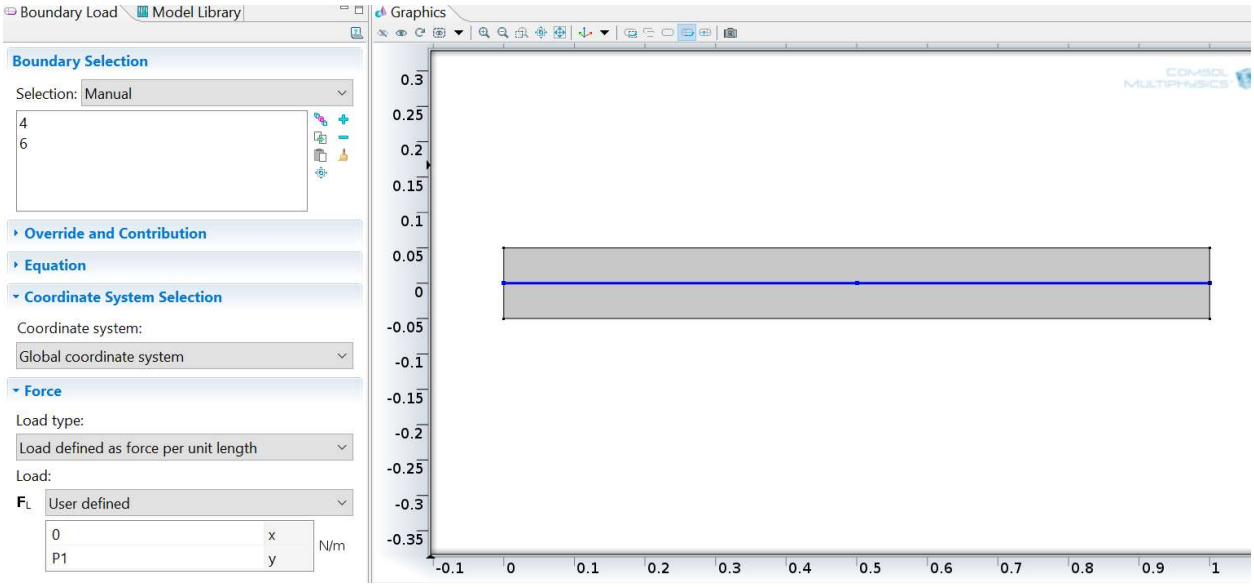


Figure 104: 2D Simply Supported Beam Boundary Load Application

Meshing can be evaluated similar to the canitlever beam by using specified a mesh convergence study using a mapped quad mesh across the length and height and comparing with finite element basis functions of linear and quadratic discretization order. The two element mesh and quality is equivalent to Figures 59 and 60, respectively. The diplacement field plots for the linear and quadratic discretizations of this coarse mesh can be seen below.

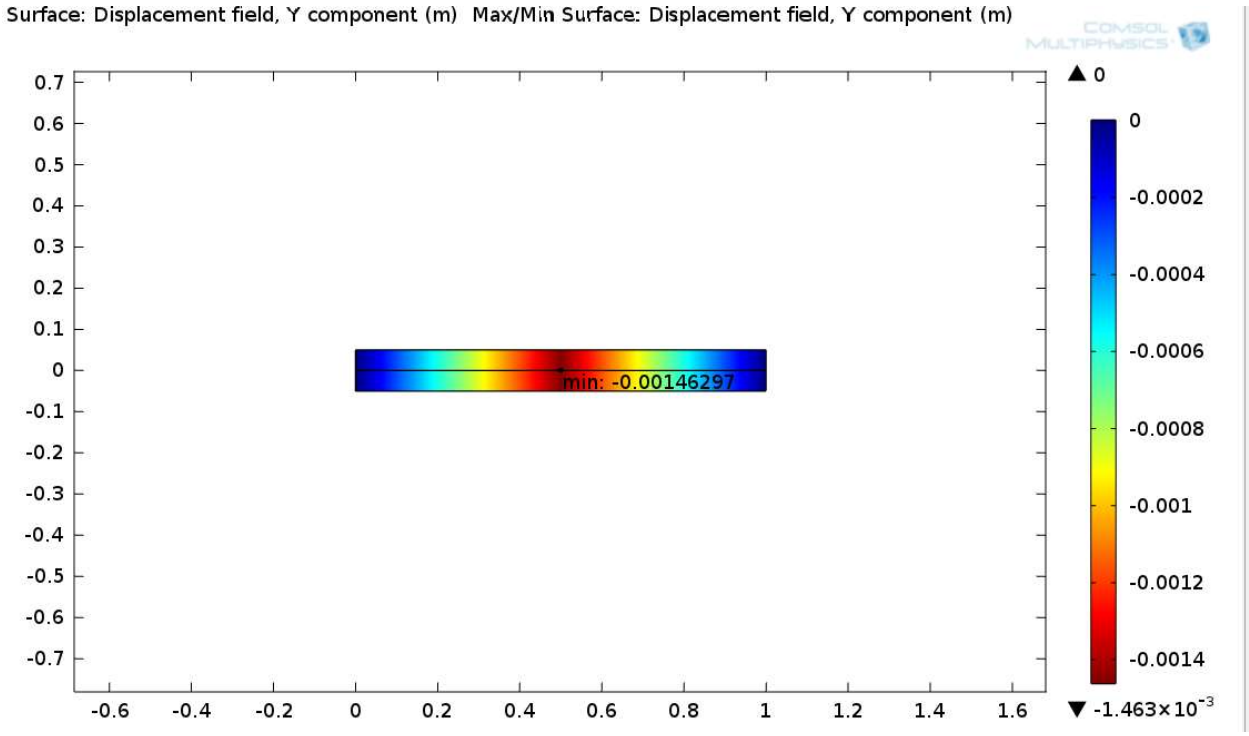


Figure 105: 2D Simply Supported Beam Coarse Mesh Displacement Field Plot (Y) and Point Evaluation – Linear Discretization

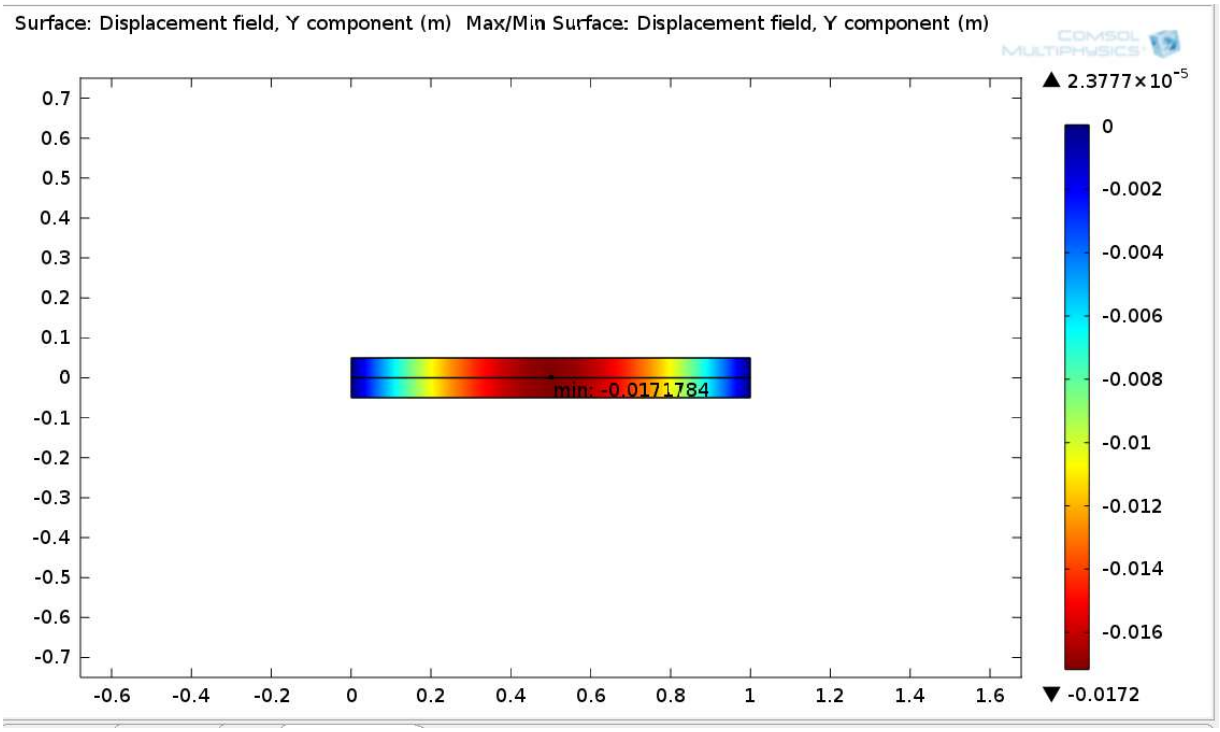


Figure 105: 2D Simply Supported Beam Coarse Mesh Displacement Field Plot (Y) and Point Evaluation – Quadratic Discretization

The mesh extension study tables for each finite element basis function can be viewed below for comparison

Mesh Study										
Simply Supproted Beam - 2D										
Discretization - Linear										
Mesh Characteristics		Study		1	2	3	4	5	6	7	8
		Number of Divisions	Height	2	2	2	6	10	50	100	200
			Length	2	10	20	50	100	500	1000	2000
		DOF		34	146	286	1914	6222	151102	602202	2404402
Results		Solve Time(s)		1	1	1	1	1	4	11	52
		Displacement (m)		-0.00146	-0.01381	-0.01937	-0.01966	-0.01975	-0.01982	-0.01985	-0.01987
Maple Solution	-0.01962	Percent Error		92.56%	29.61%	1.27%	0.20%	0.66%	1.02%	1.17%	1.27%

Figure 106: 2D Simply Supported Beam Mesh Study Table (Linear Discretization)

Mesh Study										
Simply supported Beam - 2D										
Discretization - Quadratic										
Mesh Characteristics		Study		1	2	3	4	5	6	7	8
		Number of Divisions	Height	2	2	2	6	10	50	100	200
			Length	2	10	20	50	100	500	1000	2000
		DOF		86	390	770	5326	17442	427202	1704402	6808802
Results		Solve Time(s)		1	1	1	1	2	11	43	211
		Displacement (m)		-0.01718	-0.0197	-0.01973	-0.01977	-0.0198	-0.01986	-0.01989	-0.01992
Maple Solution	-0.01962	Percent Error		12.44%	0.41%	0.56%	0.76%	0.92%	1.22%	1.38%	1.53%

Figure 107: 2D Simply Supported Beam Mesh Study Table (Quadratic Discretization)

The mesh plot and quality is equivalent to Figure 84. The displacement plots and point evaluations of the finest mesh can be seen below

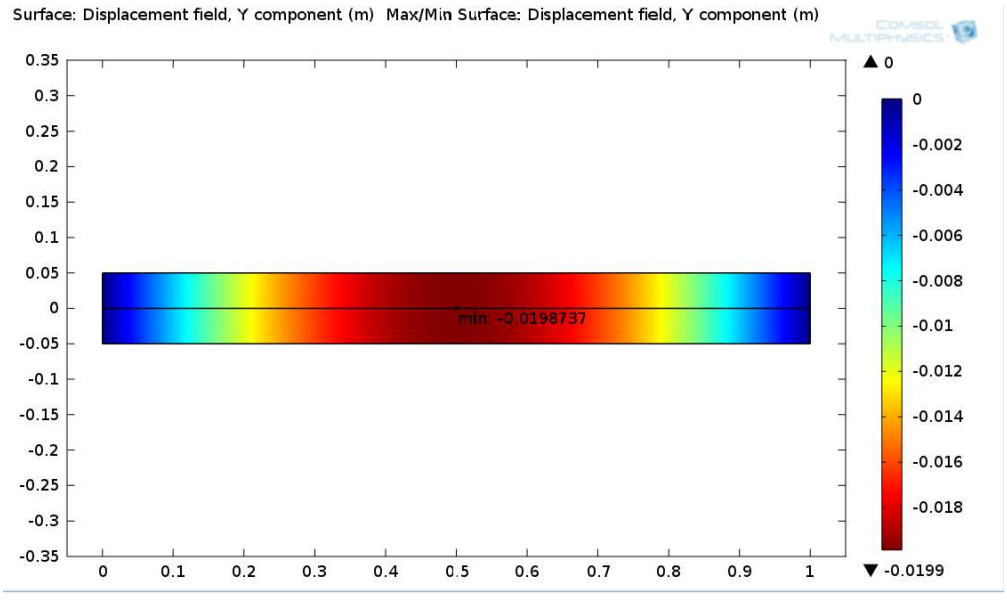


Figure 108: 2D Simply Supported Beam Fine Mesh Displacement Field Plot (Y) and Point Evaluation – Linear Discretization

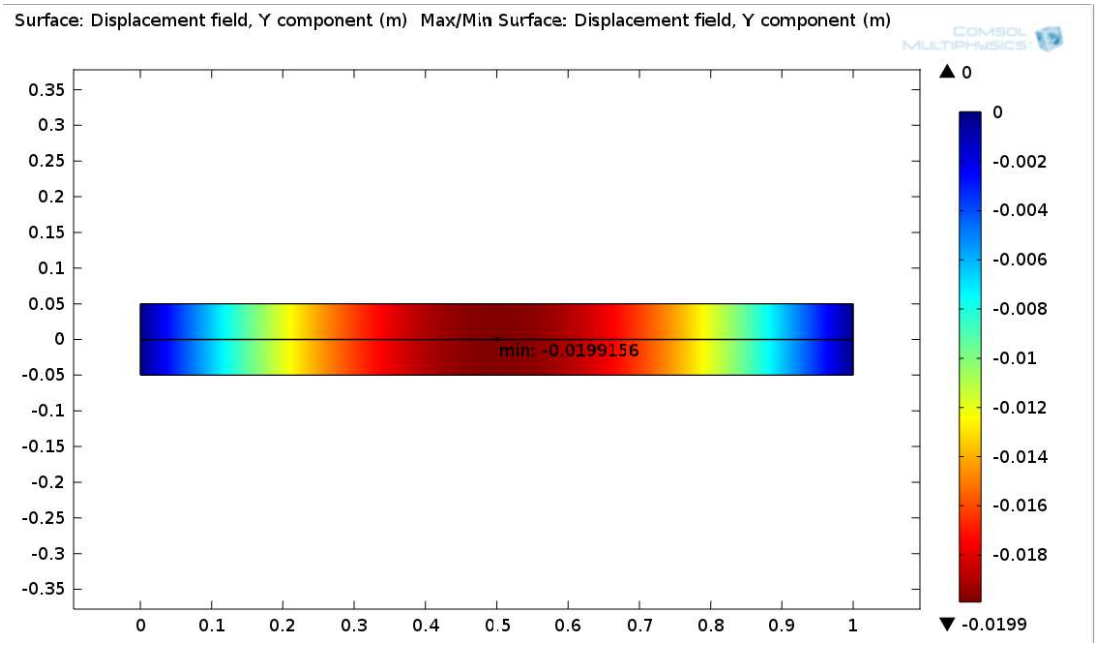


Figure 109: 2D Simply Supported Beam Fine Mesh Displacement Field Plot (Y) and Point Evaluation – Quadratic Discretization

To complete the mesh study, the mesh convergence plot is developed using a line plot

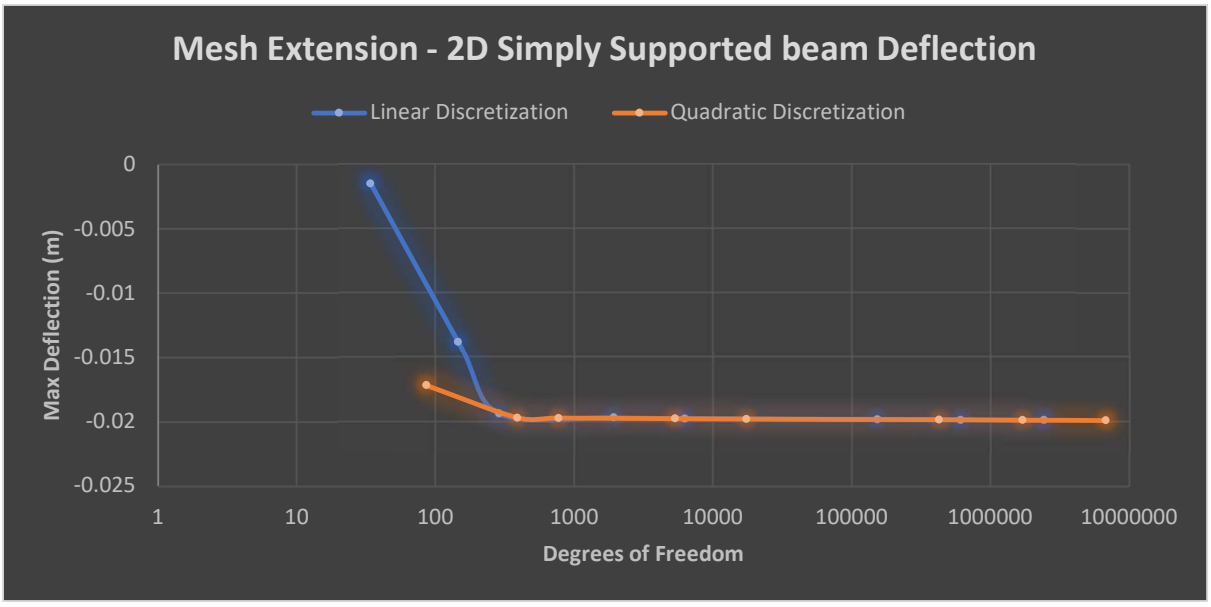


Figure 110: 2D Simply Supported Beam Fine Mesh Convergence Plot

The von Mises plot can be shown after the study is run and we can see the high stress area at the top and bottom of the beam and at the locations of the constraints

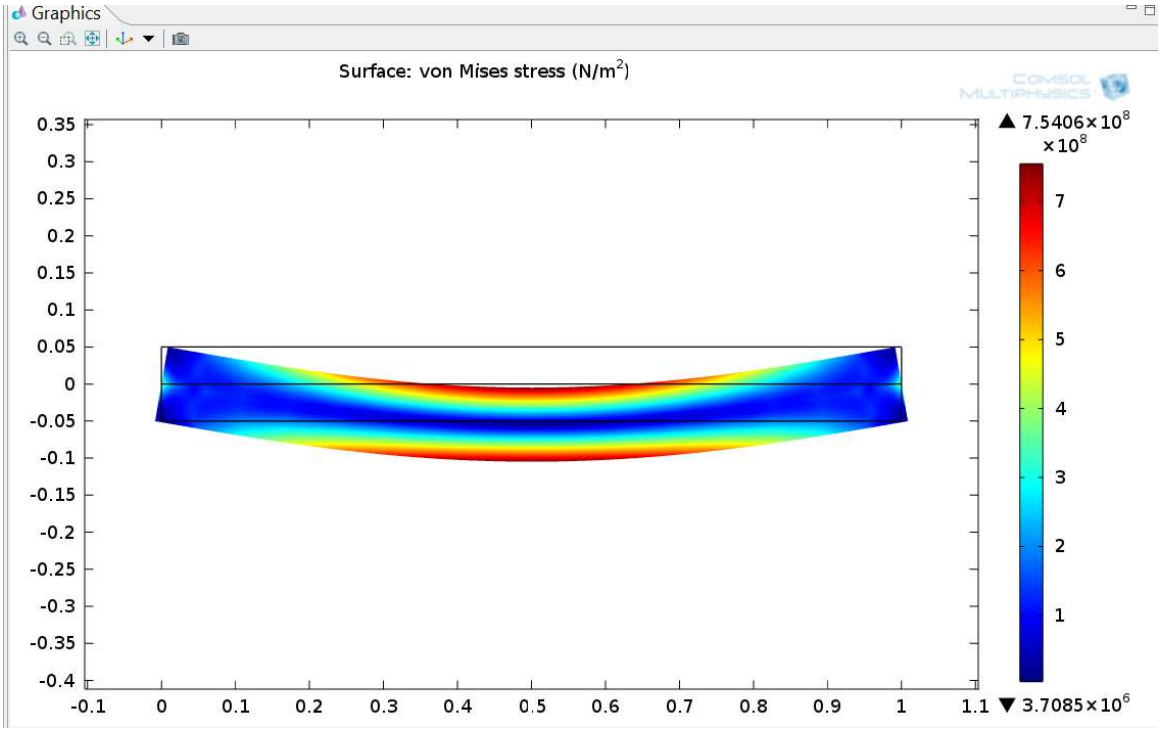


Figure 111: 2D Simply Supported Beam von Mises Stress Plot

Finally we can view a 1D displacement plot in the y direction to compare to the 1D plot in the Maple programming

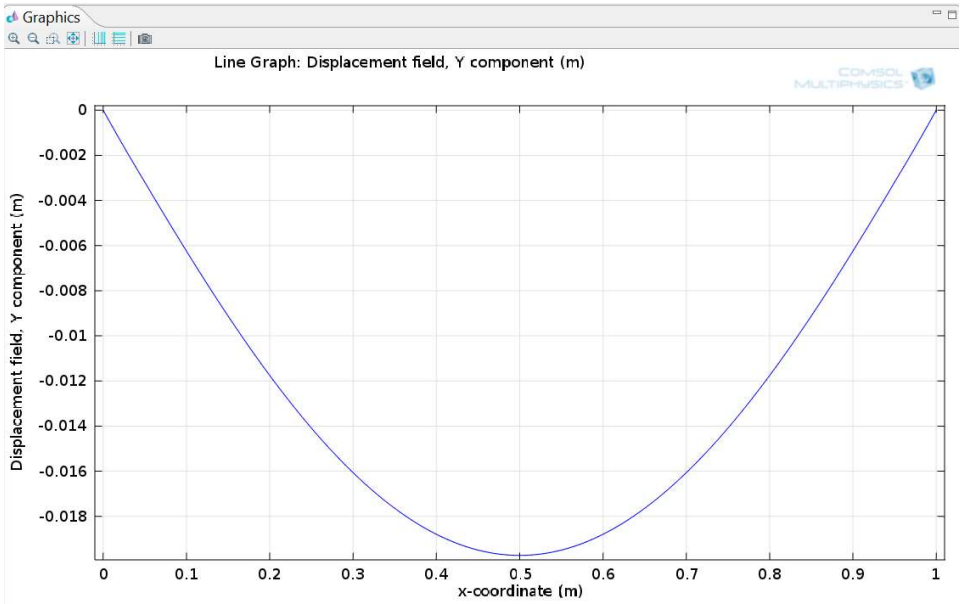


Figure 112: 2D Simply Supported Beam 1D Displacement (y) Plot

Moving to the 3D simply supported beam evaluation in COMSOL, we start by selecting a 3D workspace and set the same parameters. We will build the model in the same way, but the thickness will now be considered in the model as shown below

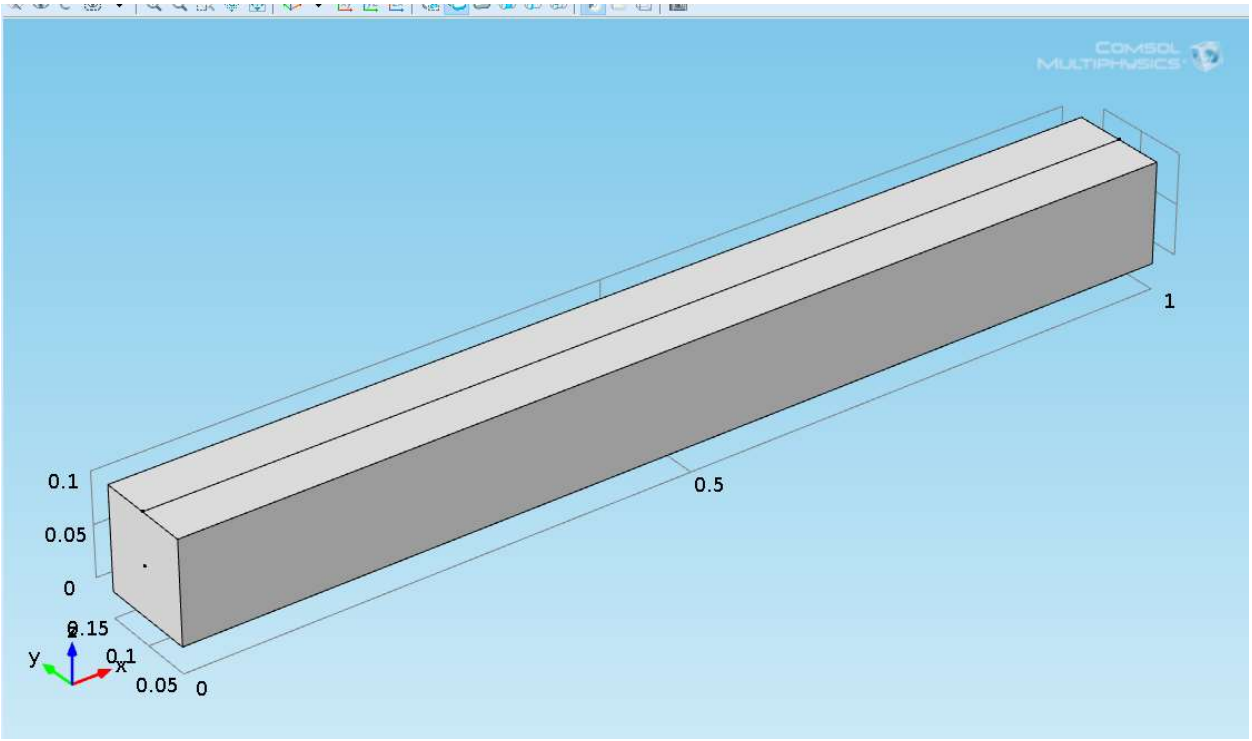


Figure 113: 3D Simply Supported Beam Model

In the same way as before, we apply the material conditions to the model

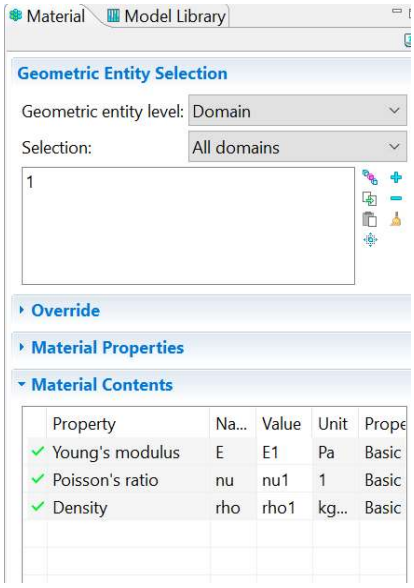


Figure 114: 3D Simply Supported Beam Material Application

Application of the constrains and load

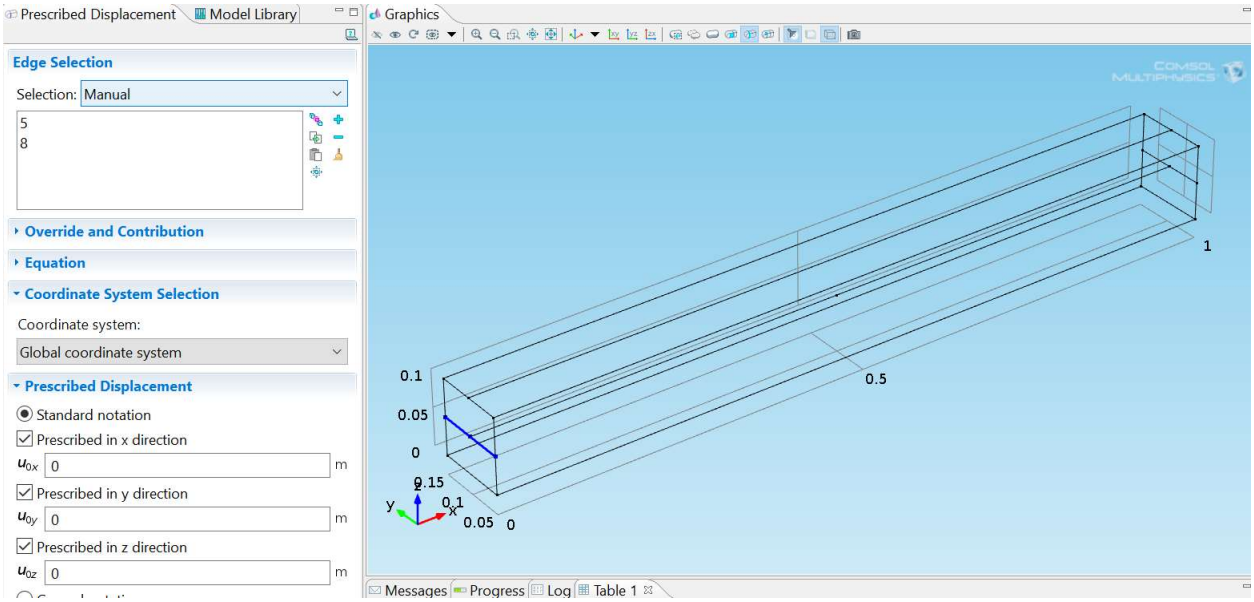


Figure 115: 3D Simply Supported Beam Model (Left End Constraint)

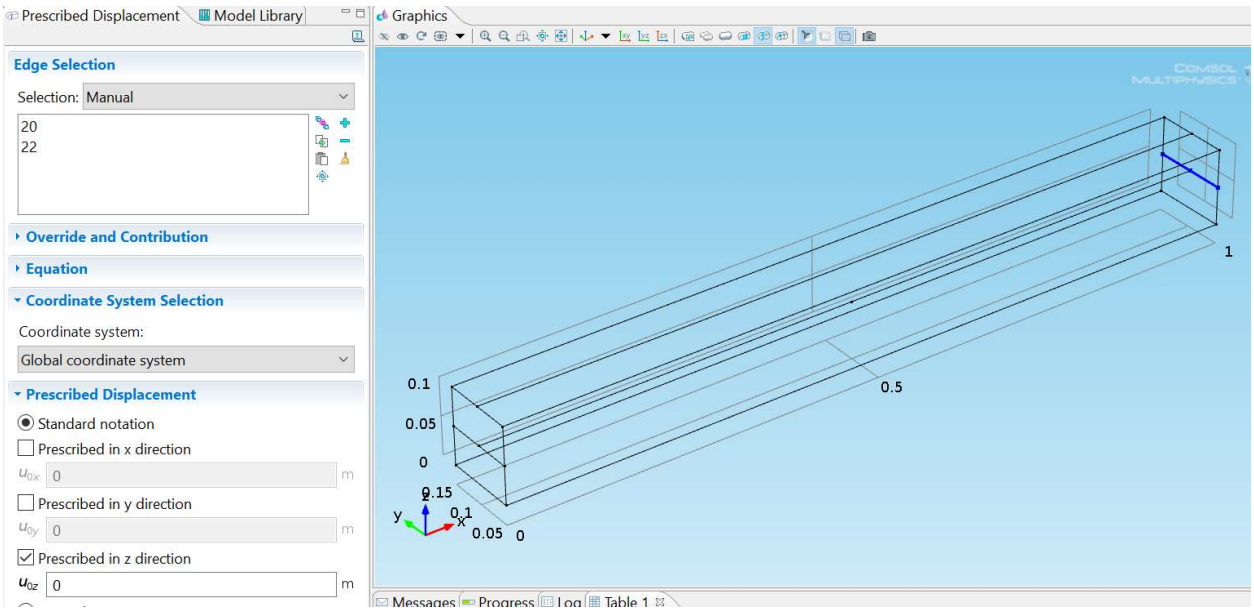


Figure 116: 3D Simply Supported Beam Model (Right End Constraint)

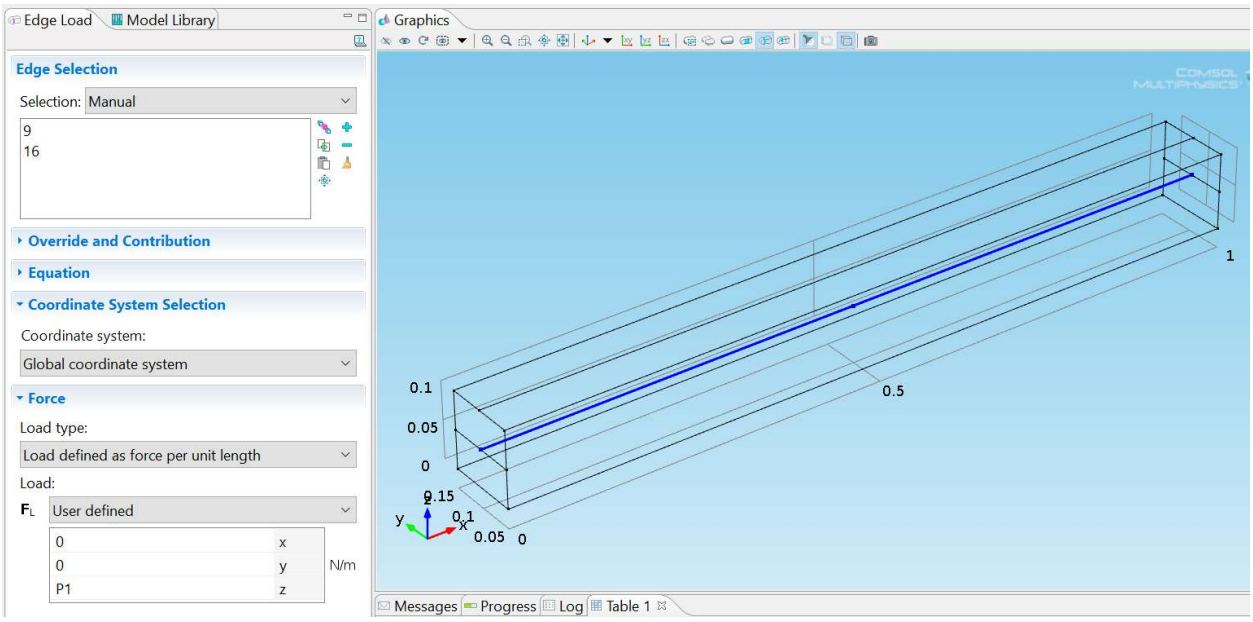


Figure 117: 3D Simply Supported Beam Load Application

The last step before computation is meshing the model. We will observe the mesh using the mesh quality tool that evaluate the quality of each element on a scale of 1 being the best, and 0 being the worst. We will complete the mesh extension study as we did in the 3D cantilever problem starting with tetraheadral physics controlled meshing at a “extra coarse setting”. For this problem we will consider the quadratic and cubic discretization since we observe that the computation is faster and more accurate than using the linear discretization. The extra coarse mesh and quality is shown below

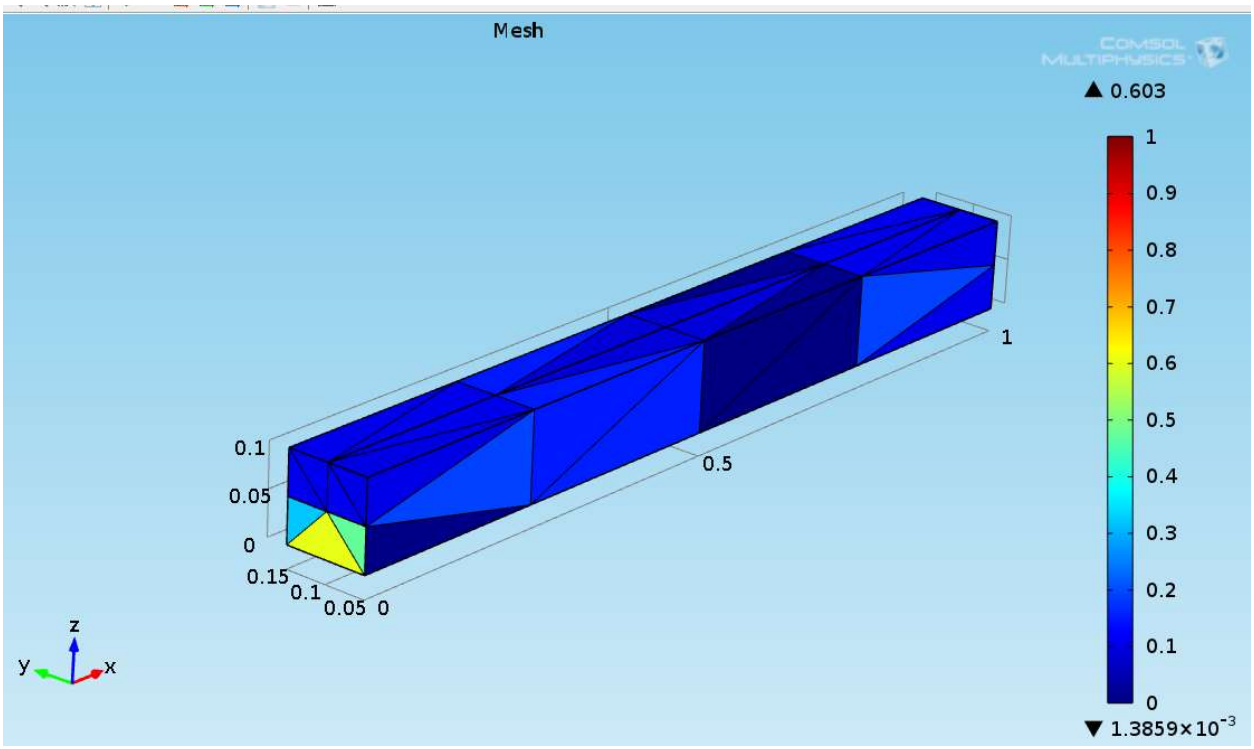


Figure 118: 3D Simply Supported Beam Extra Coarse Mesh Quality

The displacement plots and point evaluations for the extra coarse mesh

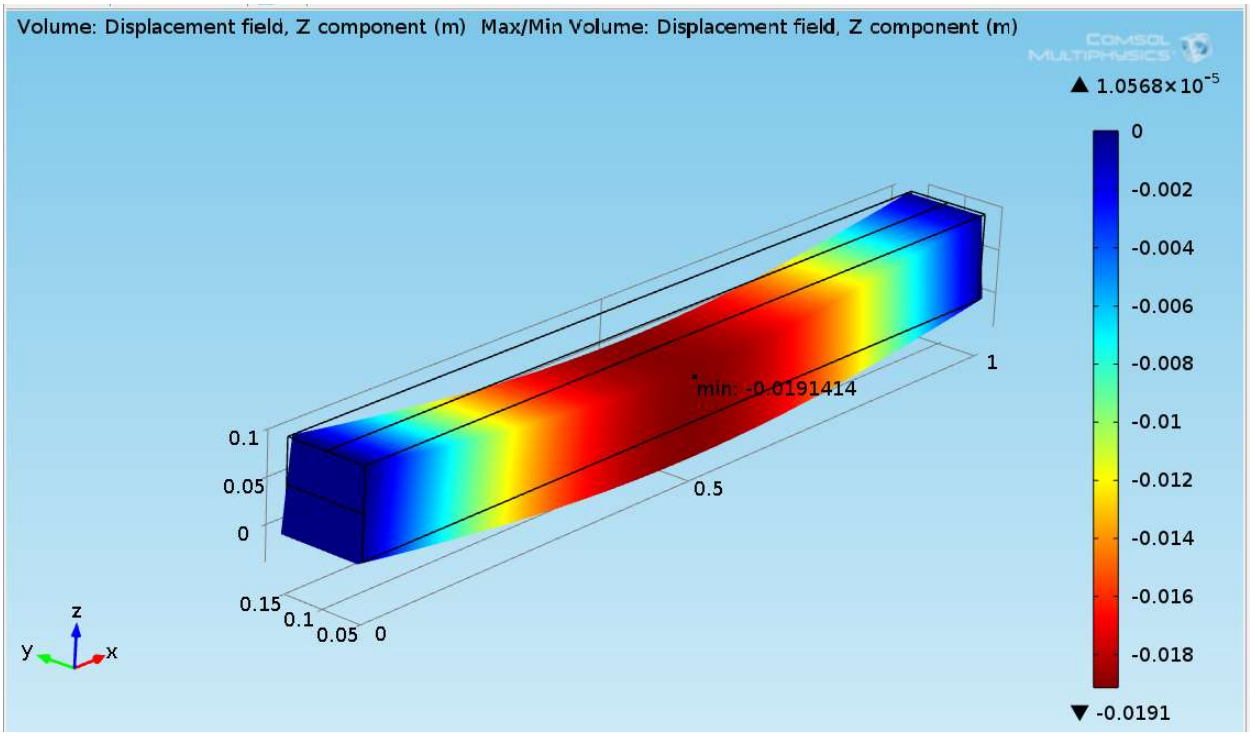


Figure 119: 3D Simply Supported Beam Coarse Mesh Displacement Field Plot (Y) and Point Evaluation – Quadratic Discretization

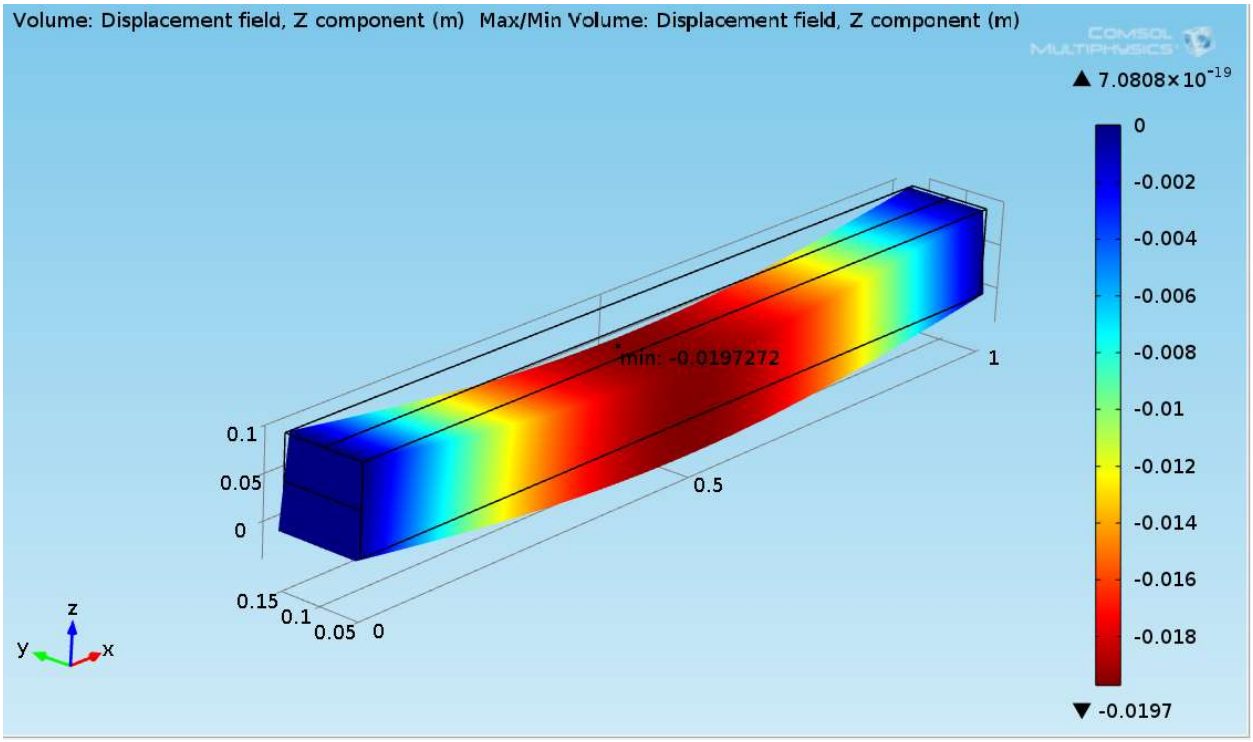


Figure 119: 3D Simply Supported Beam Coarse Mesh Displacement Field Plot (Y) and Point Evaluation – Cubic Discretization

The mesh study tables can be seen here

Mesh Study								
Cantilever Beam (Point Load) - 2D								
Discretization - Quadratic								
Mesh Characteristics		Study	1	2	3	4	5	6
		Physcis Controlled Mesh Tetrahedral	Extra Coarse	Coarse	Normal	Fine	Extra Fine	Extremely Fine
		DOF	567	936	1560	2274	17520	88050
Results		Solve Time(s)	1	1	1	1	3	12
		Displacement (m)	-0.01914	-0.01964	-0.01969	-0.0197	-0.01977	-0.01979
Maple Solution	-0.01962	Percent Error	2.45%	0.10%	0.36%	0.41%	0.76%	0.87%

Figure 120: 3D Simply Supported Beam Coarse Mesh Study Table (Quadratic Discretization)

Mesh Study								
Cantilever Beam (Point Load) - 2D								
Discretization - Cubic								
Mesh Characteristics		Study	1	2	3	4	5	6
		Physcis Controlled Mesh Tetrahedral	Extra Coarse	Coarse	Normal	Fine	Extra Fine	Extremely Fine
		DOF	1617	2697	4551	6627	54978	287079
Results		Solve Time(s)	1	1	2	2	7	62
		Displacement (m)	-0.01973	-0.01974	-0.01975	-0.01973	-0.0198	-0.0198
Maple Solution	-0.01962	Percent Error	0.56%	0.61%	0.66%	0.56%	0.92%	0.92%

Figure 120: 3D Simply Supported Beam Coarse Mesh Study Table (Cubic Discretization)

The extremely fine mesh and quality can be seen to improve dramatically. To show a transition, we show the normal mesh and quality

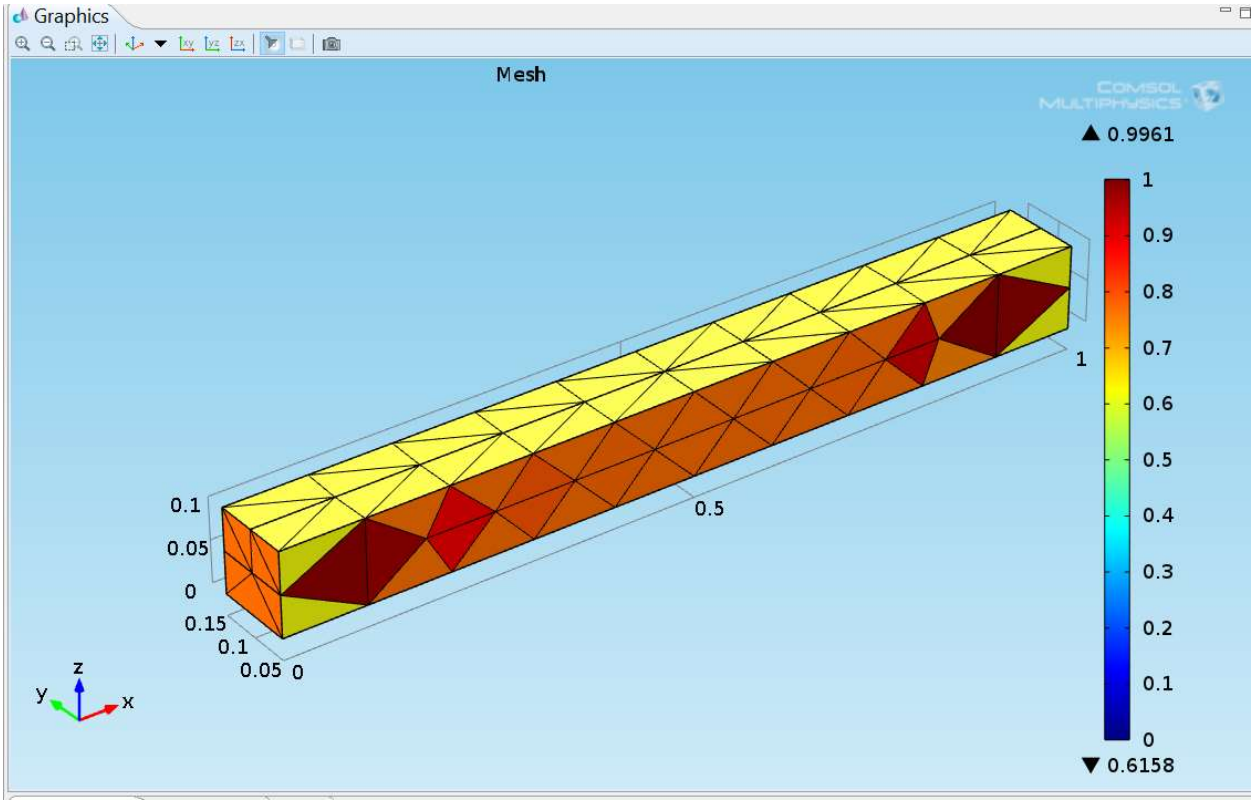


Figure 121: 3D Simply Supported Beam Model Normal Mesh Quality

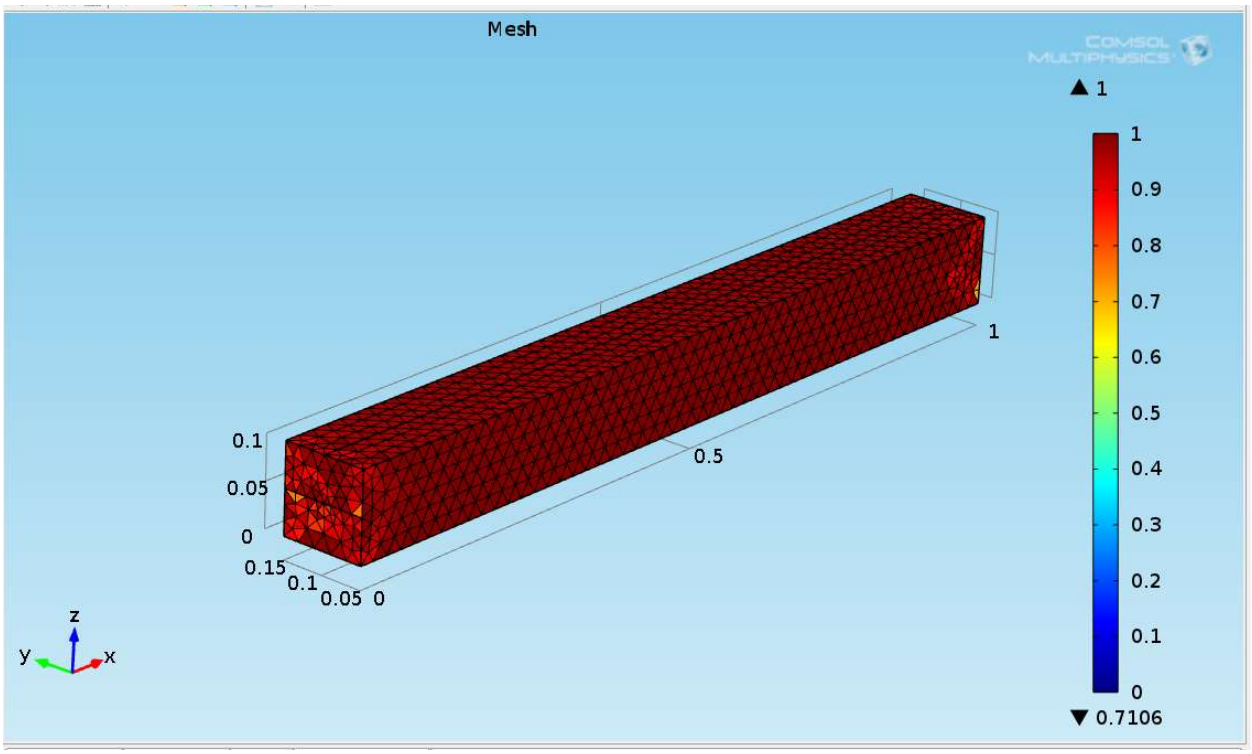


Figure 122: 3D Simply Supported Beam Model Extremely Fine Mesh Quality

The mesh quality looks to imporve along the beam and through the thickness. We can see the displacement and point evaluation of this mesh

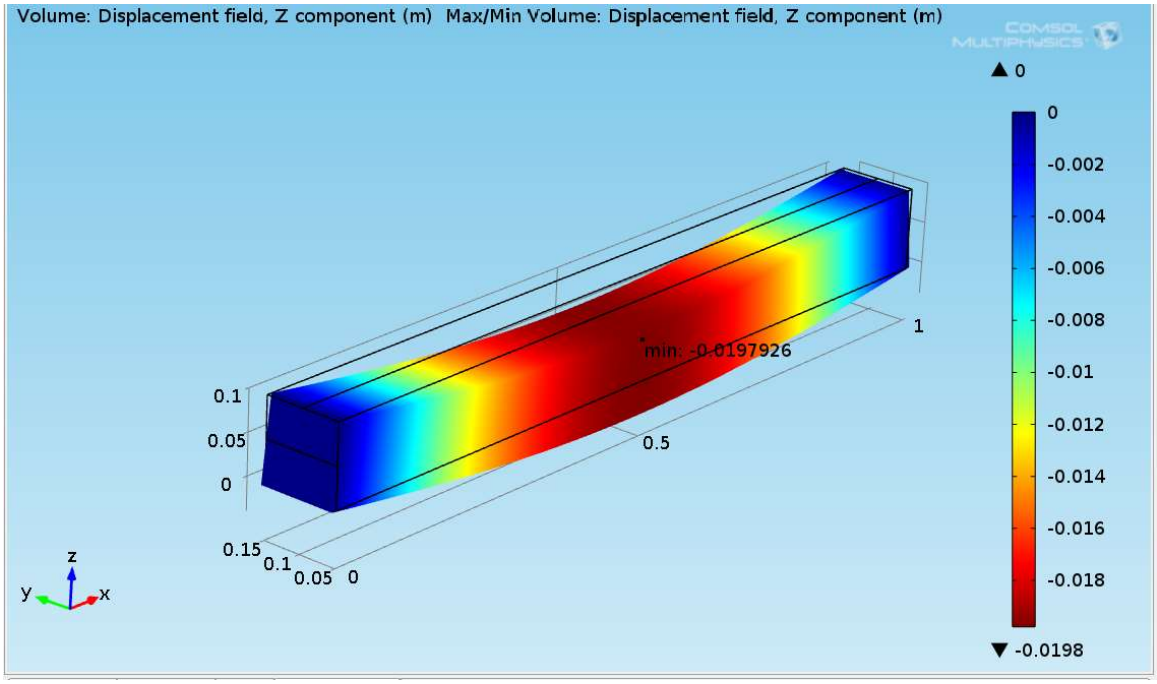


Figure 123: 3D Simply Supported Beam Extremely Fine Mesh Displacement Field Plot (Y) and Point Evaluation – Quadratic Discretization

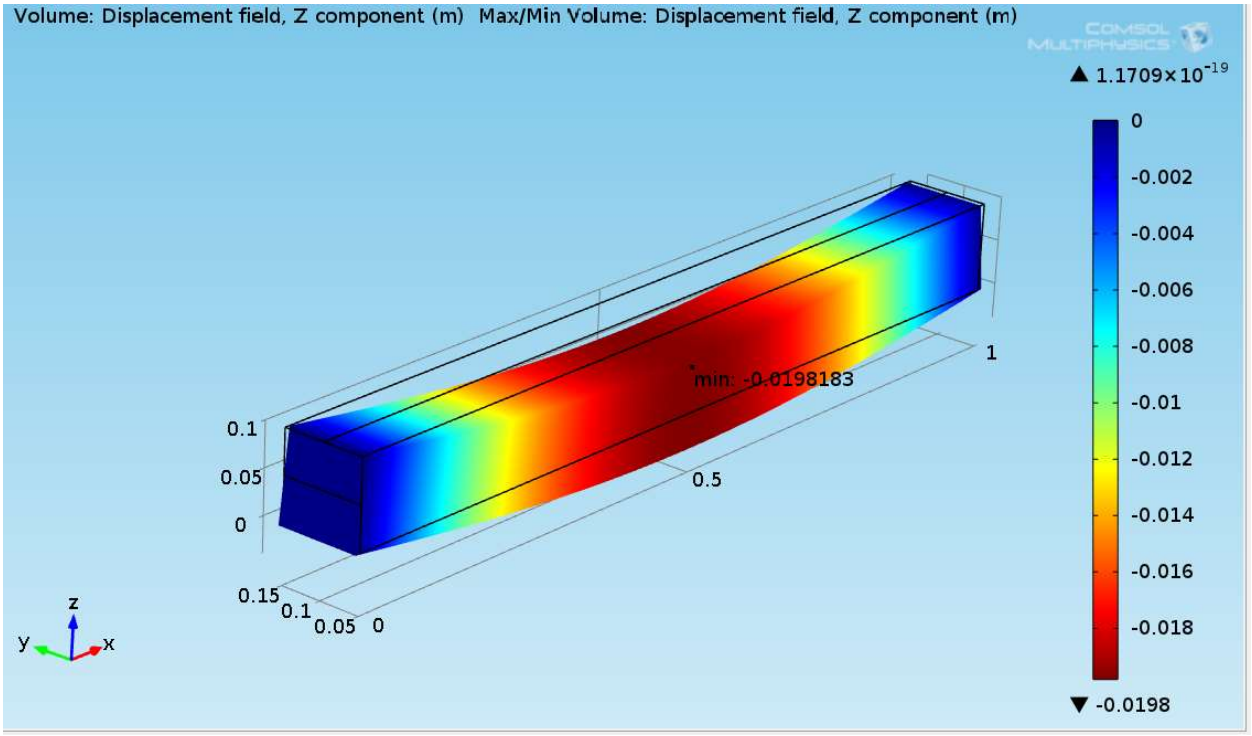


Figure 124: 3D Simply Supported Beam Extremely Fine Mesh Displacement Field Plot (Y) and Point Evaluation – Cubic Discretization

Completion of the mesh extension is the mesh convergence plot for the simply supported beam

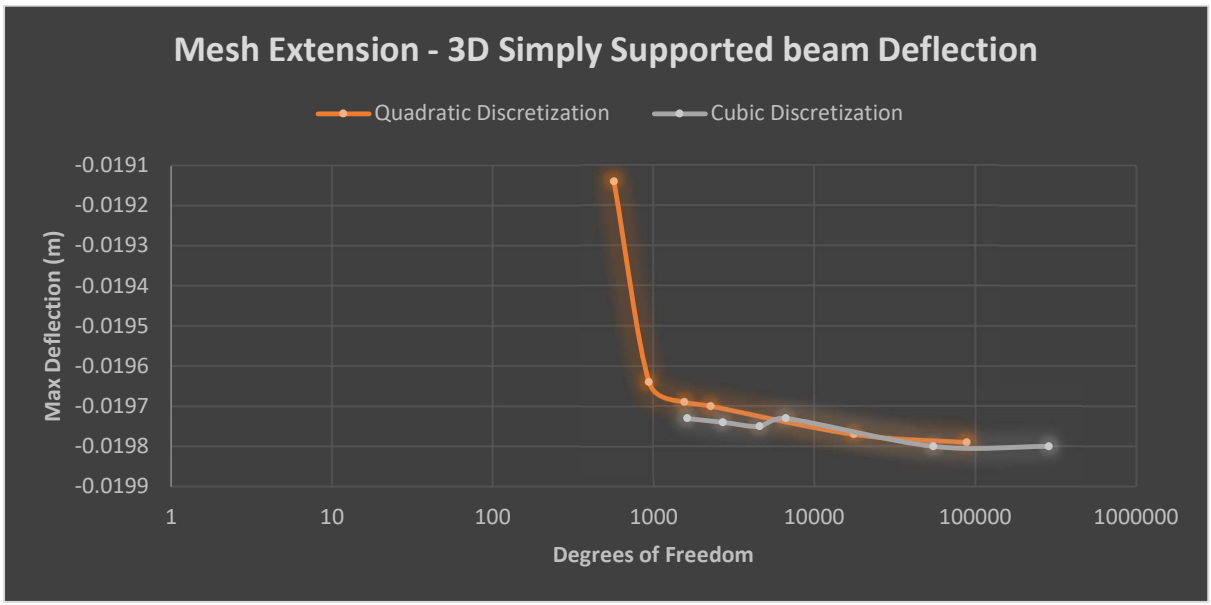


Figure 125: 3D Simply Supported Beam Mesh Convergence Plot

After computation, we can immediately view the von Mises stress in the beam

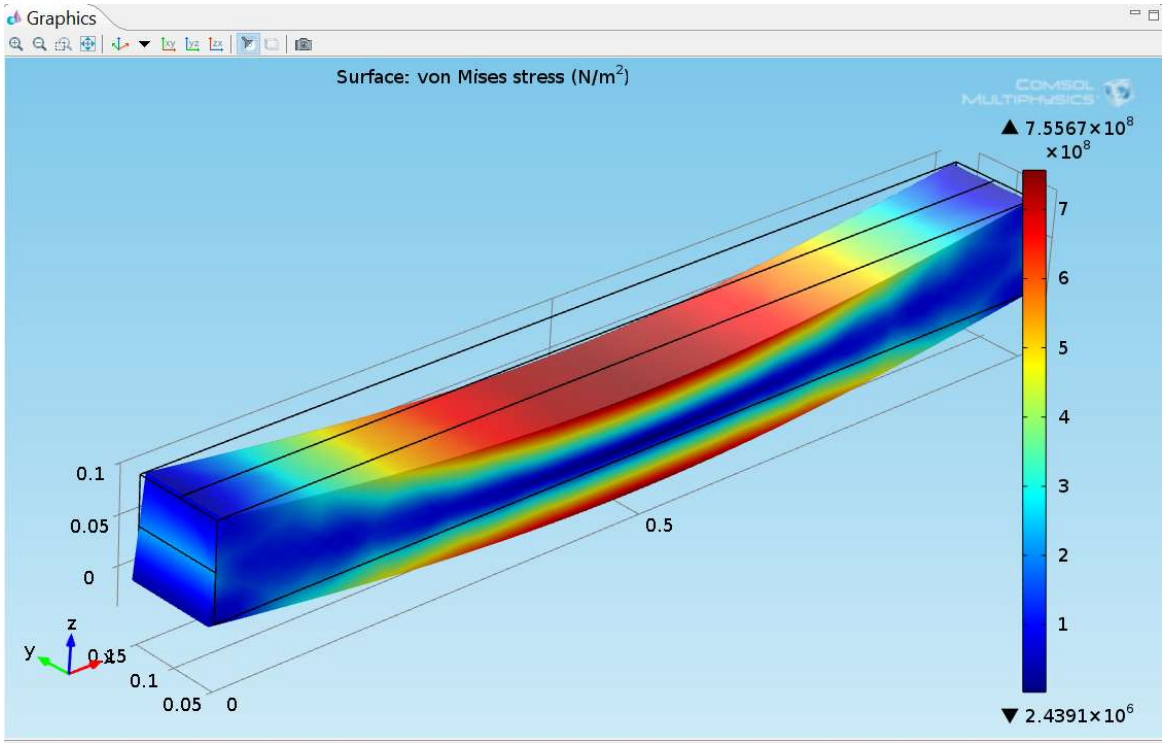


Figure 126: 3D Simply Supported Beam Model von Mises Suface Plot

From a one-dimensional standpoint and for direct comparison to the Maple displacement plot in the z direciton

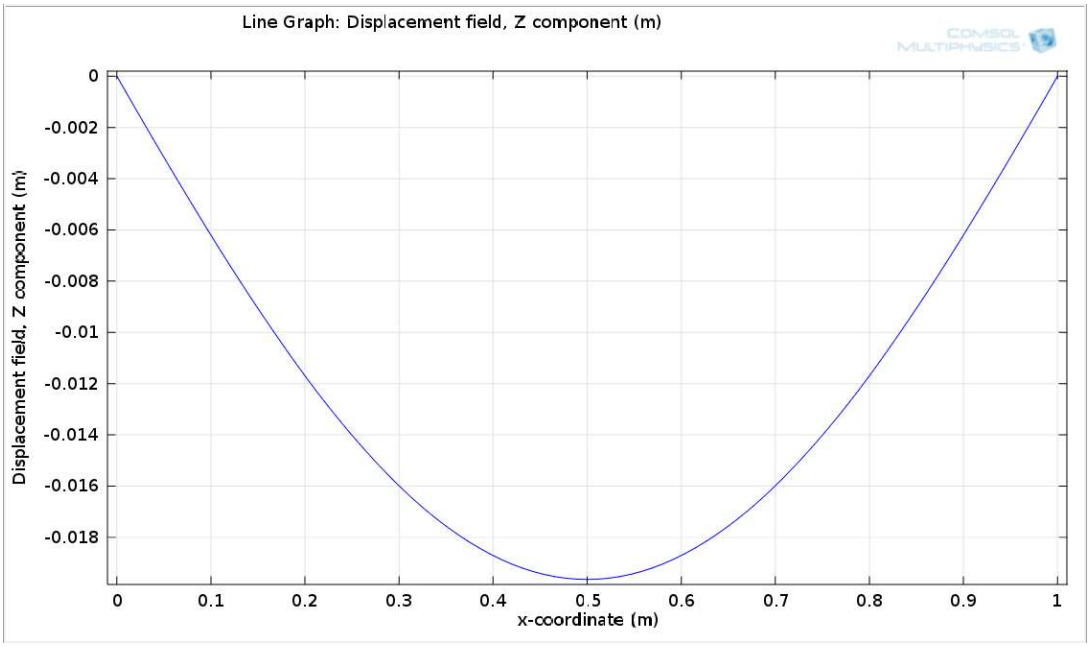


Figure 127: 3D Simply Supported Beam Model Mesh (Physics Controlled "Extremely Fine")

Finite Element Approximation - Inconel 625 Annealed Plate (Simply Supported)

Now that the beams have been studied in depth in 1D and 2D, it is time to move to plates which will more realistically model the systems of interest. This will require the same material, Inconel 625, that we have been evaluating in the cantilever and simply supported beam but introduce dimensions and loads that are more realistic to rupture discs made in the micro-hydraulic industry. A square plate 1cm x 1cm with a thickness of 0.01 cm will be used to simulate the rupture disc.

First, we must understand how to find the solution using Navier Solution for Simply Supported Plates, which is described in Timoshenko Plate and Shell Theory after describing methods toa solution using a sinusoidal loading condition, but Navier uses a Fourier series for load and deflection that is powerful enough to capture any loading conditions in one equation over the whole surface of the plate. We will look at a simply supported, square plate for this approach [14] [15].

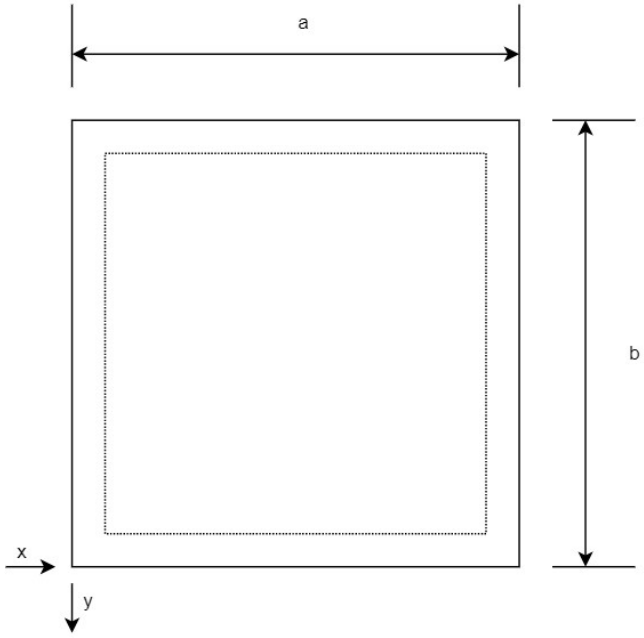


Figure 128: Simply Supported Plate

The general plate is subjected to a distributed load P(x,y). We again start with the governing equation

$$\nabla^4 w = \frac{p}{D} \tag{1.82}$$

$$\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} = \frac{p}{D} \tag{1.84}$$

The deflection of the plate must meet the governing equation and boundary conditions of

$$w = 0 = \frac{\partial^2 w}{\partial x^2} \ @ \ (x = 0, x = a) \tag{1.85}$$

$$w = 0 = \frac{\partial^2 w}{\partial y^2} \ @ \ (y = 0, y = b) \tag{1.86}$$

At large, the load and deflection can be described as

$$p(x,y) = \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} p_{mn} * \sin\left(\frac{m\pi x}{a}\right)\sin\left(\frac{n\pi y}{b}\right) \tag{1.87}$$

$$w(x,y) = \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} a_{mn} * \sin\left(\frac{m\pi x}{a}\right)\sin\left(\frac{n\pi y}{b}\right) \tag{1.88}$$

[15] To better understand equation 1.88 and the determination of a_{mn}, we can imagine sinusoidal curve as below over the length and width of the plate with infinite possible formations in the form of m and n. a_{mn} can then be interpretted as the maximum central coordinates of the curves and m and n are the amount of half sine curves in both directions. When we increase the number of m and n, then our accuracy will improve, but will be convergent in nature [15].

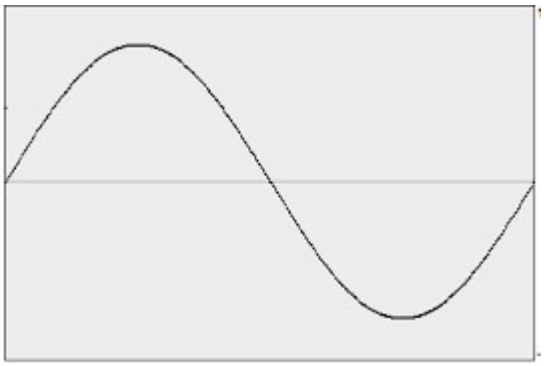


Figure 129: Navier Plate Theory Visualization (Sinusoidal Curve)

A method to finding p_{mn} would be to multiply both sides of equation 1.87 by $\sin\left(\frac{m'\pi x}{a}\right)\sin\left(\frac{n'\pi y}{b}\right)dxdy$ and integrate between the limits of the plate

$$\sum_{m=1}^{\infty}\sum_{n=1}^{\infty} p_{mn}\int_0^b\int_0^a \sin\left(\frac{m\pi x}{a}\right)\sin\left(\frac{n\pi y}{b}\right)\sin\left(\frac{m'\pi x}{a}\right)\sin\left(\frac{n'\pi y}{b}\right)dxdy \tag{1.89}$$

$$\int_0^b \sin\left(\frac{n\pi y}{b}\right)\sin\left(\frac{n'\pi y}{b}\right)dy = \begin{cases} 0 & (n \neq n') \\ b/2 & (n = n') \end{cases} \tag{1.90}$$

$$\int_0^a \sin\left(\frac{m\pi x}{a}\right)\sin\left(\frac{m'\pi x}{a}\right)dx = \begin{cases} 0 & (m \neq m') \\ a/2 & (m = m) \end{cases} \tag{1.91}$$

Treating the fourier series yields

$$p_m(y) = \frac{2}{a}\int_0^a p(x,y)\sin\left(\frac{m\pi x}{a}\right)dx, \ p_n(x) = \frac{2}{b}\int_0^b p(x,y)\sin\left(\frac{n\pi x}{b}\right)dy \tag{1.92}$$

Combining

$$p_{mn} = \frac{4}{ab}\int_0^a\int_0^b p(x,y)\sin\left(\frac{m\pi x}{a}\right)\sin\left(\frac{n\pi y}{b}\right)dxdy \tag{1.93}$$

To solve for a_{mn}, we can plug eq 1.88 into 1.84,

$$\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\left\{a_{mn}\left[\left(\frac{\mathrm{m\pi}}{\mathrm{a}}\right)^4+2\left(\frac{\mathrm{m\pi}}{\mathrm{a}}\right)^2\left(\frac{\mathrm{n\pi}}{\mathrm{b}}\right)^2+\left(\frac{\mathrm{n\pi}}{\mathrm{b}}\right)^4\right]-\frac{p_{mn}}{D}\right\}\sin\left(\frac{m\pi x}{a}\right)\sin\left(\frac{n\pi y}{b}\right)=0 \qquad (1.94)$$

In order for this to apply to all positions of x and y,

$$a_{mn}=\frac{1}{\pi^4 D}*\frac{p_{mn}}{\left[\left(\frac{m}{a}\right)^2+\left(\frac{n}{b}\right)^2\right]^2} \qquad (1.95)$$

And plugging back into the deflection equation 1.88, the deflection for a multitude of loads can be described by

$$w=\frac{1}{\pi^4 D}\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{p_{mn}}{\left[\left(\frac{m}{a}\right)^2+\left(\frac{n}{b}\right)^2\right]^2}\sin\left(\frac{m\pi x}{a}\right)\sin\left(\frac{n\pi y}{b}\right) \qquad (1.96)$$

When considering our simply supported plate subjected to a uniformly distributed load (q), simplifications can be made to the above approach. Integrating eq 1.93,

$$p_{mn}=\frac{4q}{\pi^2 mn}(1-\cos(m\pi))(1-\cos(n\pi))=$$

$$p_{mn}=\frac{16q}{\pi^2 mn}\ when\ m,n=odd\ integers \qquad (1.97)$$

We specify odd integers for m and n because with even numbers, $p_{mn}=0$. Plugging p_{mn} into the deflection equation, we obtain

$$w=\frac{16}{\pi^6 D}\sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\frac{\sin\left(\frac{m\pi x}{a}\right)\sin\left(\frac{n\pi y}{b}\right)}{\left[\left(\frac{m}{a}\right)^2+\left(\frac{n}{b}\right)^2\right]^2},\quad (m,n=1,3,5,9\ldots) \qquad (1.98)$$

We can utilize Maple software to evaluate the defleciton of the simply supported plate as shown below

```
> restart;
```

Navin_Ian_Milestone 3 - Plate Theory, Navier Solution for Thin Plates

Inconel 625 Annealed Plate at RT

Flexural Rigidity

```
> De := E·h^3 / (12·(1 − nu^2));
```

$$De := \frac{E\,h^3}{-12\,\nu^2 + 12}$$

```
> P := -1e6 : a := .01 : E := 8.14e10 : b := .01 : I1 := b·h^3/12 : h := 0.0001 : nu := .278 : rho := 8440 : alpha := b/a : M := 7 : N := 7 :
>
> w := (16·P/(π^6·De)) · sum( sum( sin(m·Pi·x/a)·sin(n·Pi·y/b) / (m·n·((m^2/a^2)+(n^2/b^2))^2), n = 1..N ), m = 1..M ) :
>
>
>
> subs({y = b/2, x = a/2}, w) :
```

Figure 130: Navier Plate Solution for Thin Plates Displacement

```
> plot(subs({y = b/2}, w), x = 0..a);
```

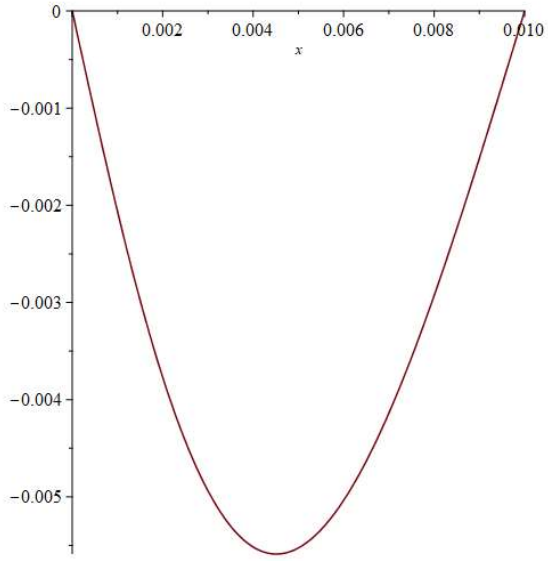


Figure 131: Navier Plate Solution 1D Displacement Plot in Z Component

```
> plot3d(w, x = 0 ..a, y = 0 ..b);
```

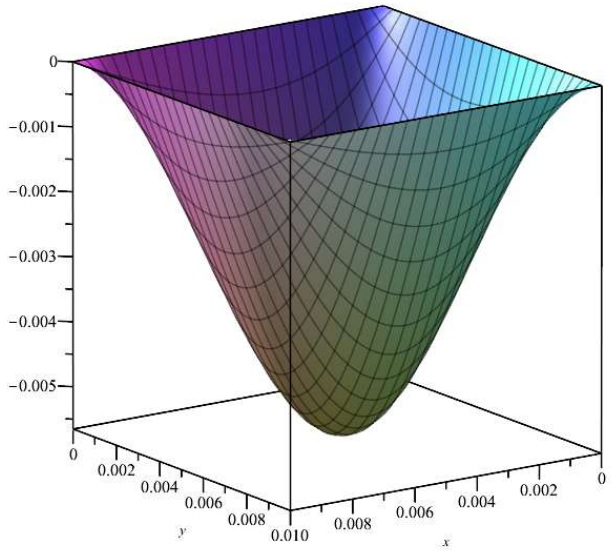


Figure 132: Navier Plate Solution 3D Displacement Plot in Z Component

```
> evalf(subs({x = a/2, y = b/2}, w));
                                   −0.005525451836

> evalf(int(int(w, x = 0 ..a), y = 0 ..b));
                                  −2.315815009 10^−7
```

Figure 133: Navier Plate Solution 3D Displacement Value in Z Component and Global Error

With the Maple evaluation complete, we can compare using COMSOL in a 3D space. Just as in the above evaluations using COMSOL, we define the parameters for the plate geometry and material

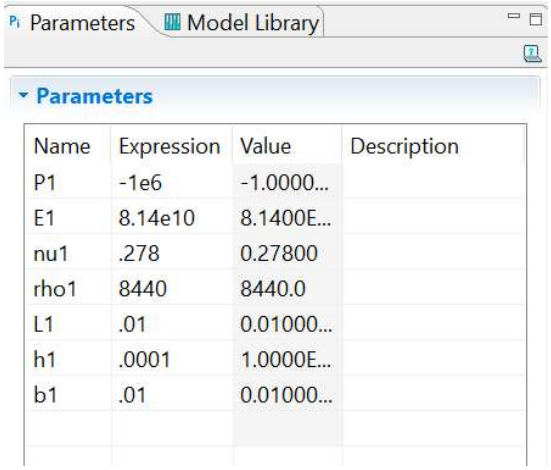


Figure 134: COMSOL Simply Supported Plate Material and Geometry Parameters

Building the model with the specified length, breadth, and thickness yields

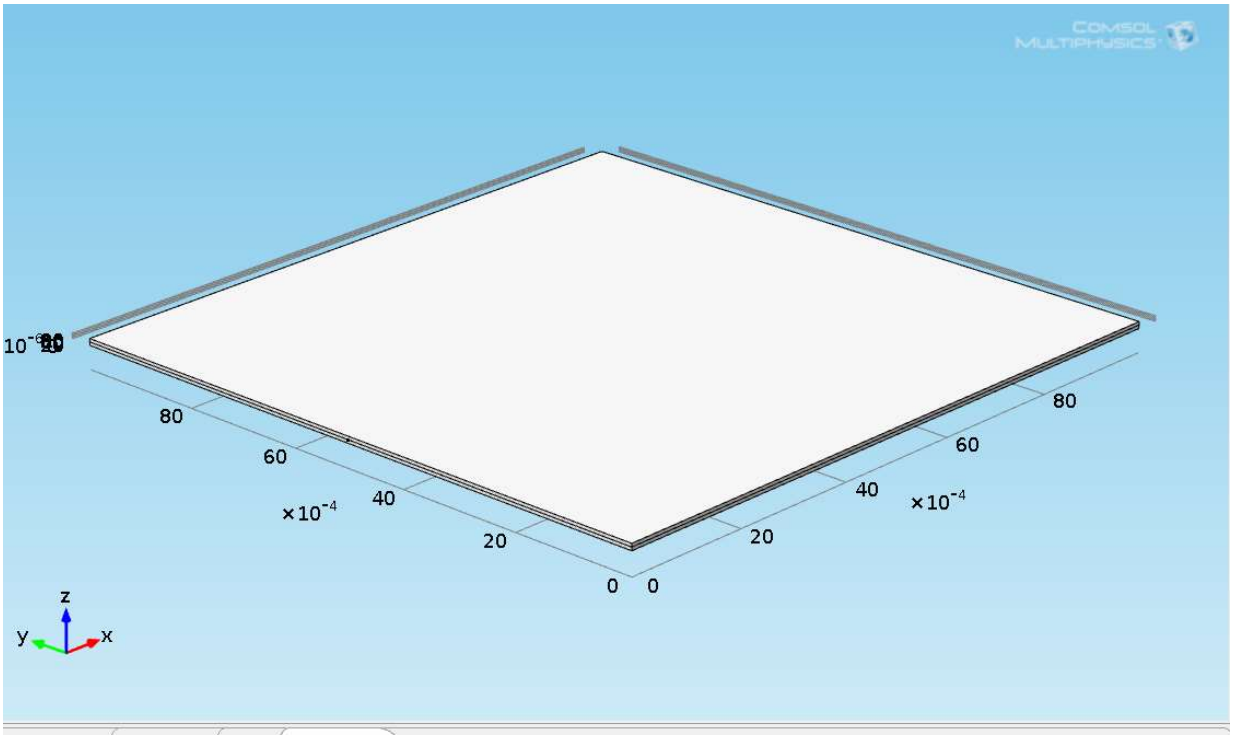


Figure 135: 3D Simply Supported Plate Model

Application of the material properties is completed using the material definition and selecting the domain.

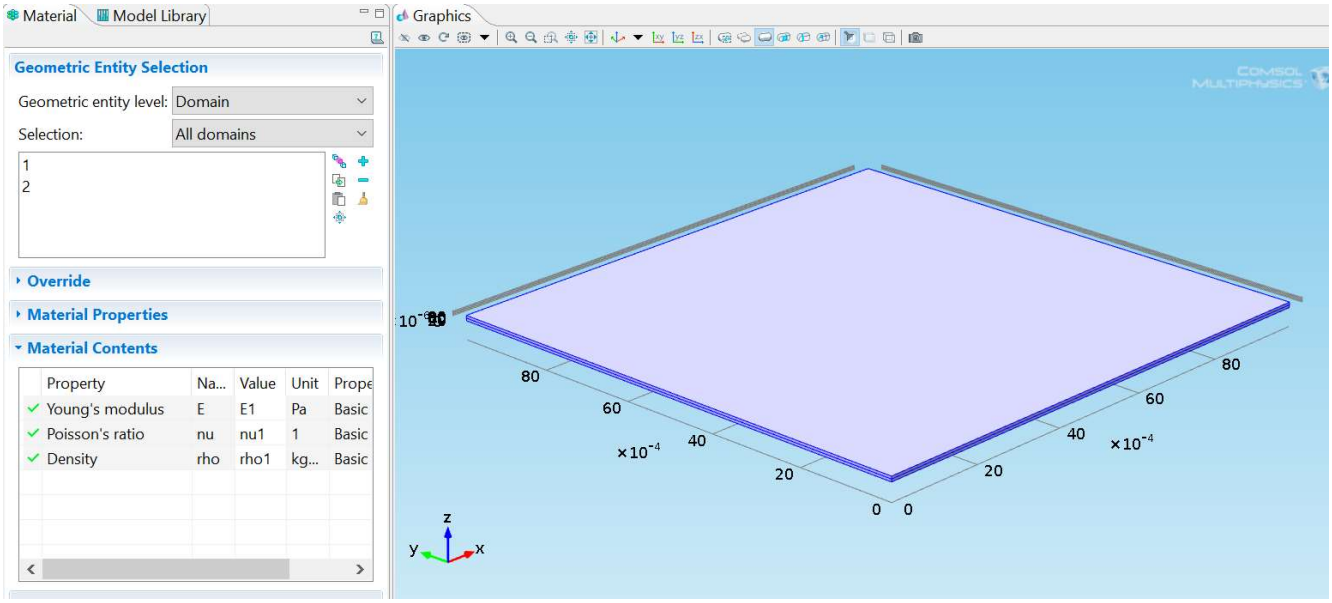


Figure 136: 3D Simply Supported Plate Model Material Application

Next the loading condition and constraints can be applied

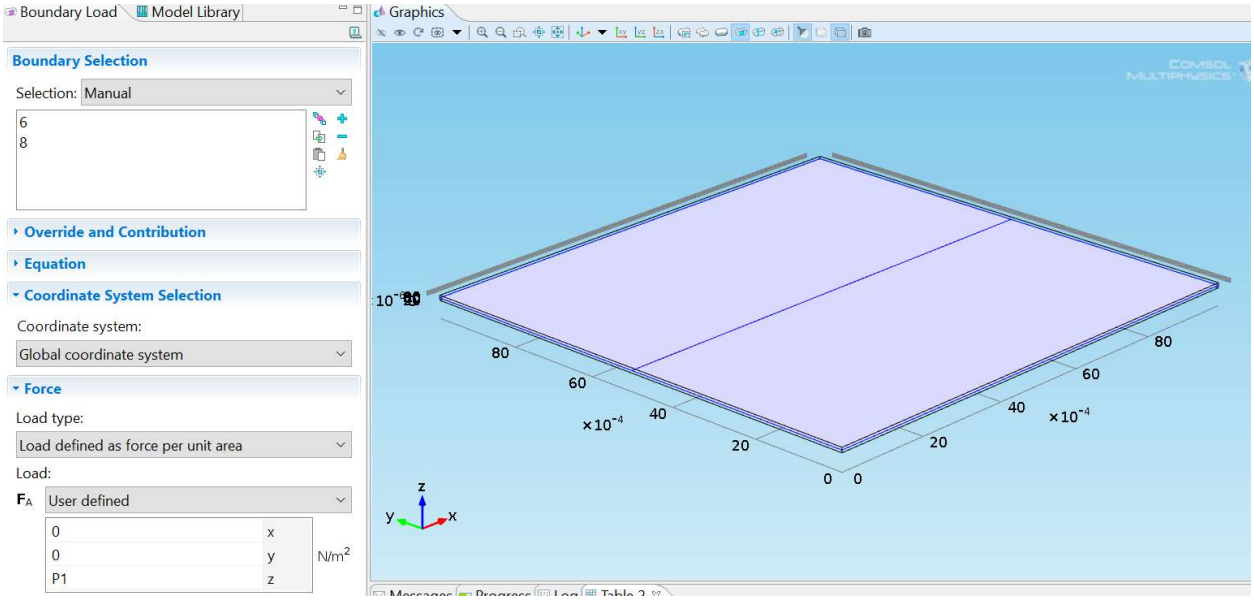


Figure 137: 3D Simply Supported Plate Model Load Application

Adding the simply supported constraints to a 3D plate is somewhat tricky, but is accomplished as shown below

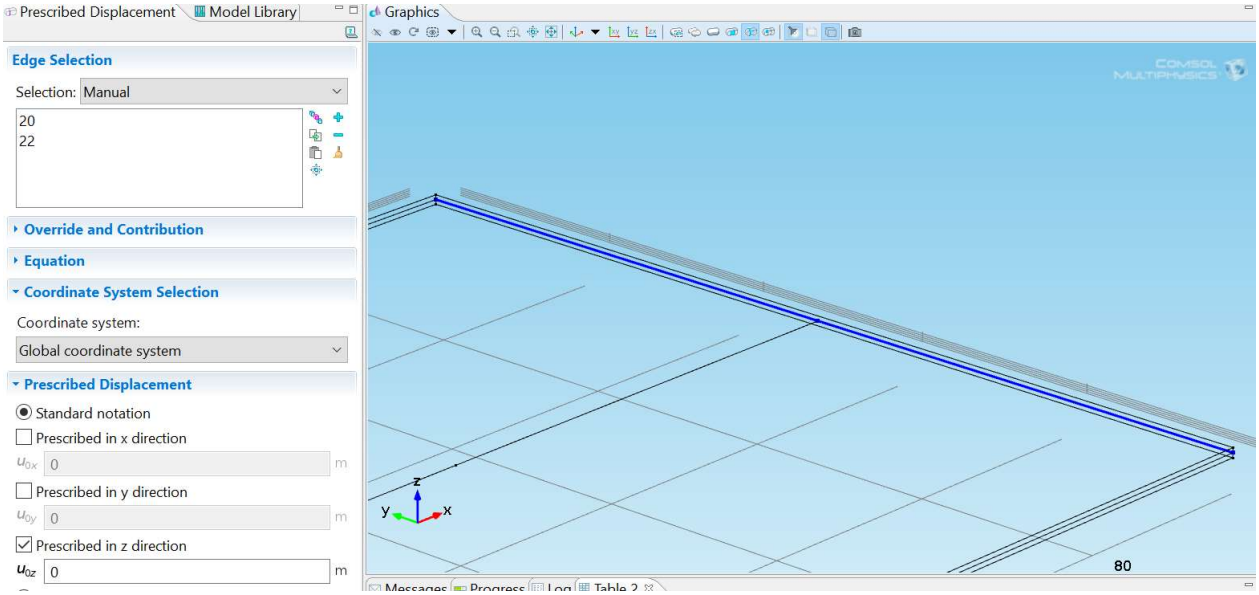


Figure 138: 3D Simply Supported Plate Model Constrain (Fixed in Z Direction – 1)

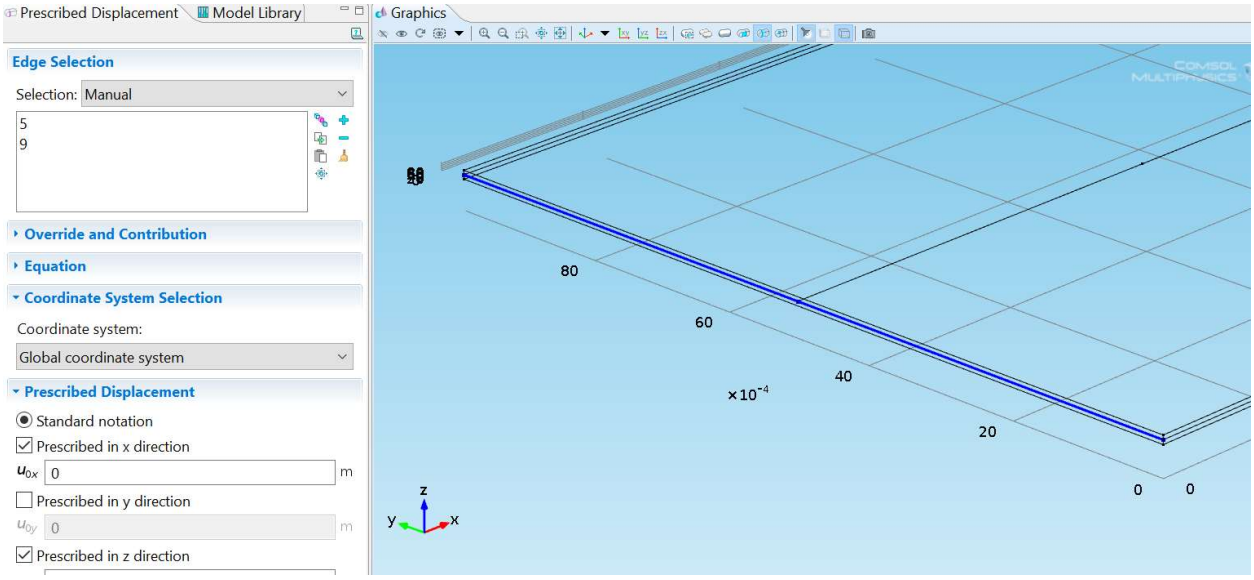


Figure 139: 3D Simply Supported Plate Model Constrain (Fixed in X and Z Direction)

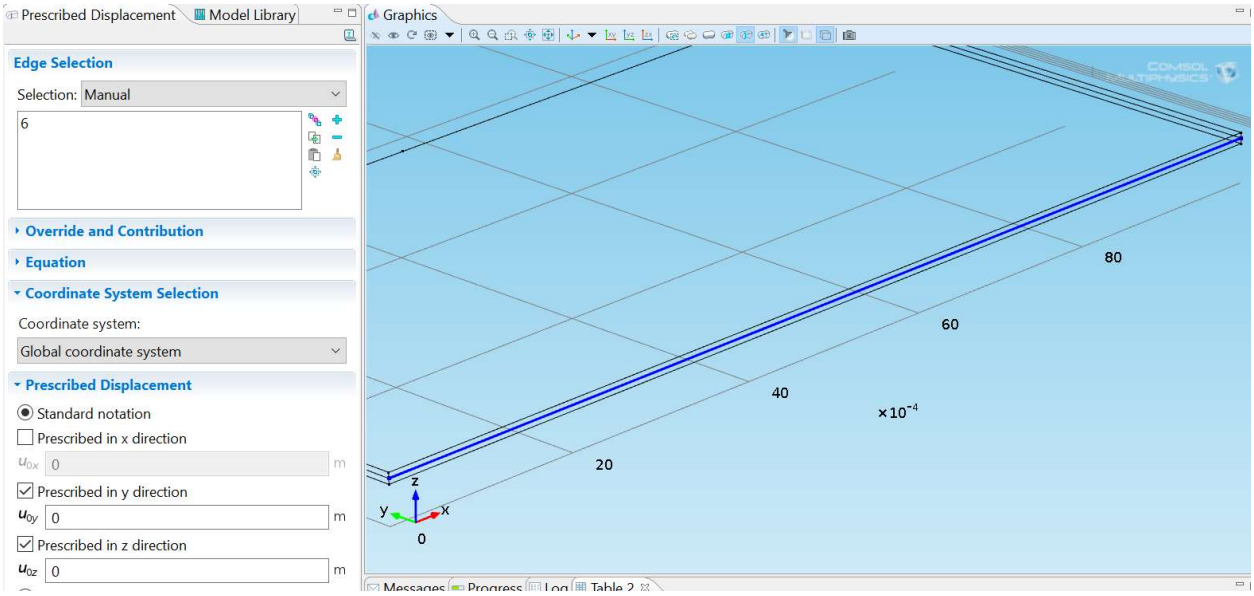


Figure 140: 3D Simply Supported Plate Model Constrain (Fixed in Y and Z Direction)

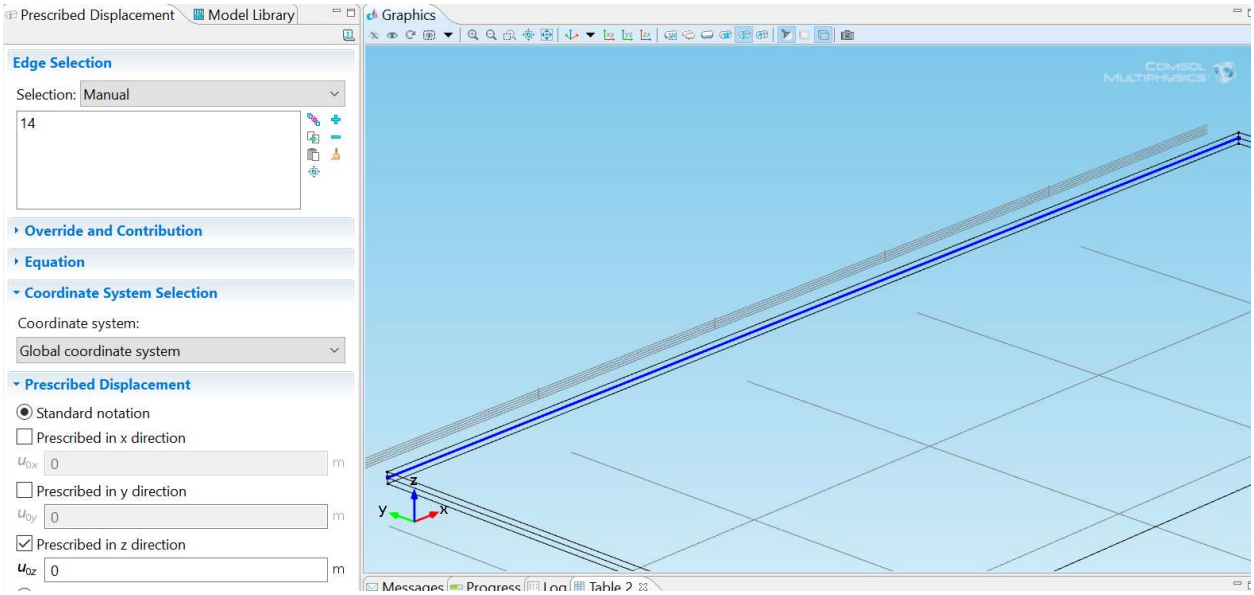


Figure 141: 3D Simply Supported Plate Model Constrain (Fixed in Z Direction – 2)

Completing a mesh study, we use a mapped mesh on the top and bottom faces and sweep through the thickness. The study will show the benefit of increasing elements through the thickness and the length/width. A linear and quadratic finite element basis order is used. We start with a 2x2 mesh with 1 element through the thickness. Warning: the images below for the thickness will appear to have double to elements chosen. This is because there are beizier polynomials draw through the center of the thickess for application of the boundary conditions.

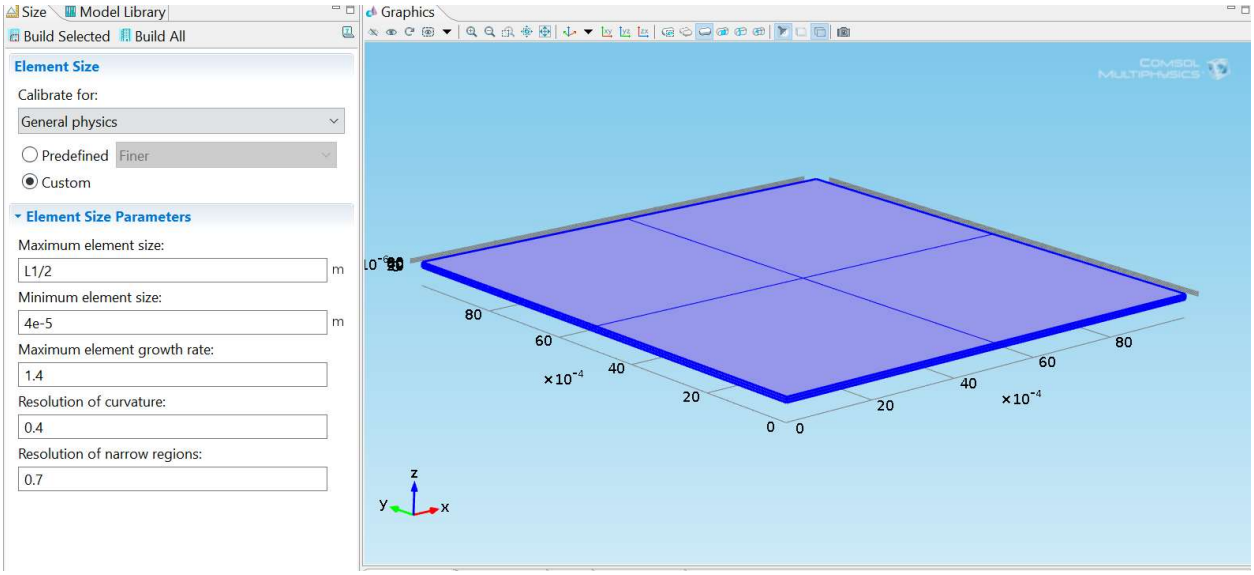


Figure 142: 3D Simply Supported Plate Model Mapped and Swept Mesh (Coarse)

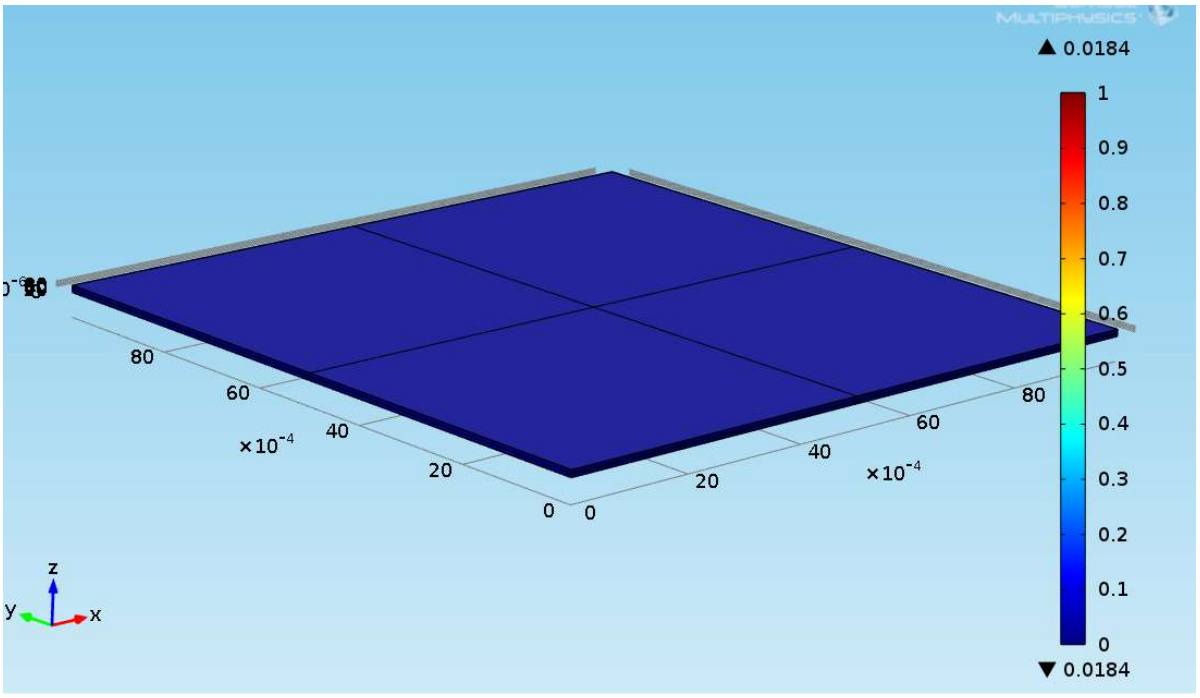


Figure 142: 3D Simply Supported Plate Model Mapped and Swept Mesh Quality (Coarse)

After computing the study, we can plot the 3D volume deformation and evaluate the displacement in the z direction at the center of the plate for comparison to the Navier Solution for the linear and quadradic discretization methods of the coarse mesh.

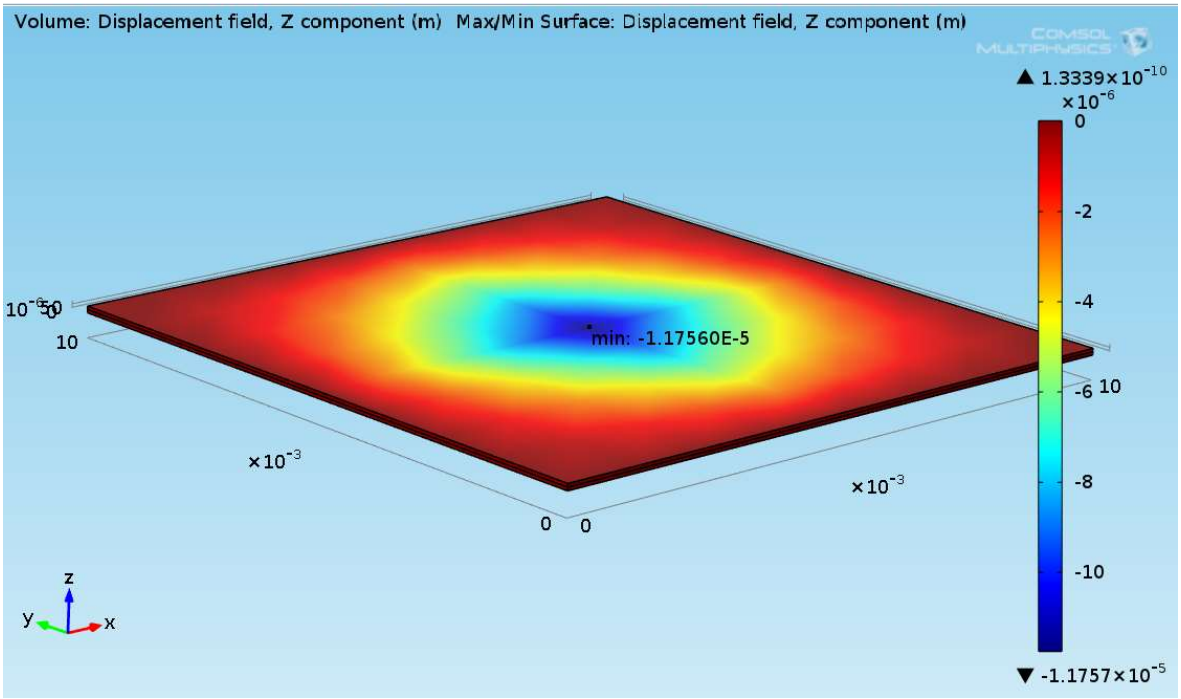


Figure 143: 3D Simply Supported Plate Model Coarse Mesh Displacement Plot (Z) and Point Evaluation – Linear Discretization

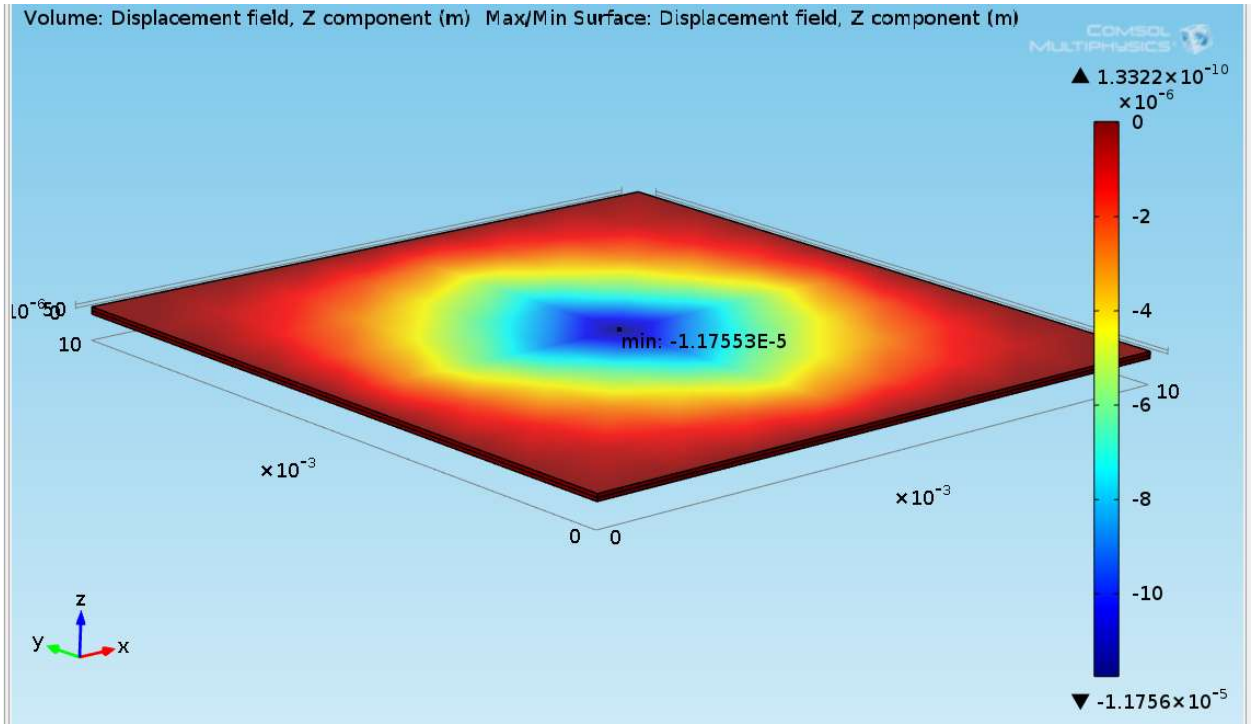


Figure 144: 3D Simply Supported Plate Model Coarse Mesh Displacement Plot (Z) and Point Evaluation – Quadratic Discretization

The study can be viewed in the tables below

Mesh Study											
Simply suppoted Plate											
Discretization - Linear											
Mesh Characteristics		Study		1	2	3	4	5	6	7	8	9
		Number of Divisions	Width	2	2	50	100	100	50	200	200	500
			Length	2	2	50	100	100	50	200	200	500
			Thickness	1	20	1	1	4	20	1	4	1
		DOF		81	567	23409	91809	153015	163863	363609	606015	2259009
Results		Solve Time(s)		1	1	3	7	19	35	30	66	297
		Displacement (m)		-1.18E-05	-1.18E-05	-0.00315	-0.00455	-0.00467	-0.00323	-0.00511	-0.00528	-0.0053
Maple Solution	-0.00553	Percent Error		99.79%	99.79%	42.99%	17.65%	15.48%	41.54%	7.51%	4.43%	4.05%

Figure 145: 3D Simply Supported Plate Model Mesh Study (Linear Discretization)

Mesh Study								
Simply supported plate								
Discretization - Quadratic								
Mesh Characteristics		Study		1	2	3	4	5	6
		Number of Divisions	Width	2	2	50	50	100	100
			Length	2	2	50	50	100	100
			Thickness	1	20	1	4	1	4
		DOF		375	3075	153015	275427	606015	1090827
Results		Solve Time(s)		1	1	22	55	102	341
		Displacement (m)		-4.84E-03	-4.84E-03	-0.00557	-0.00557	-0.00558	-0.00558
Maple Solution	-0.00553	Percent Error		12.40%	12.40%	0.81%	0.81%	0.94%	0.90%

Figure 146: 3D Simply Supported Plate Model Mesh Study (Quadratic Discretization)

The mesh for the finer mesh can be seen of being a much high quality on the face and through the thickness

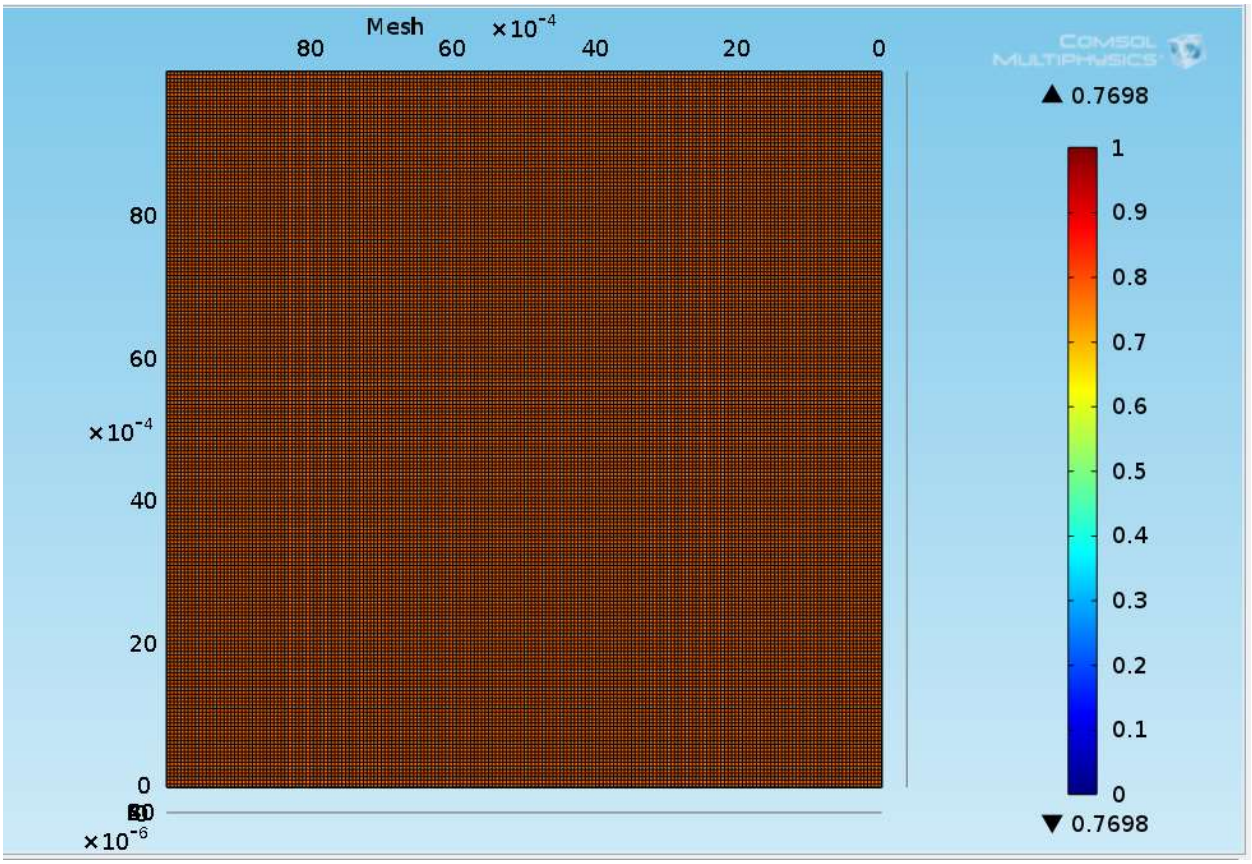


Figure 147: 3D Simply Supported Plate Model Fine Mesh Quality (Face)

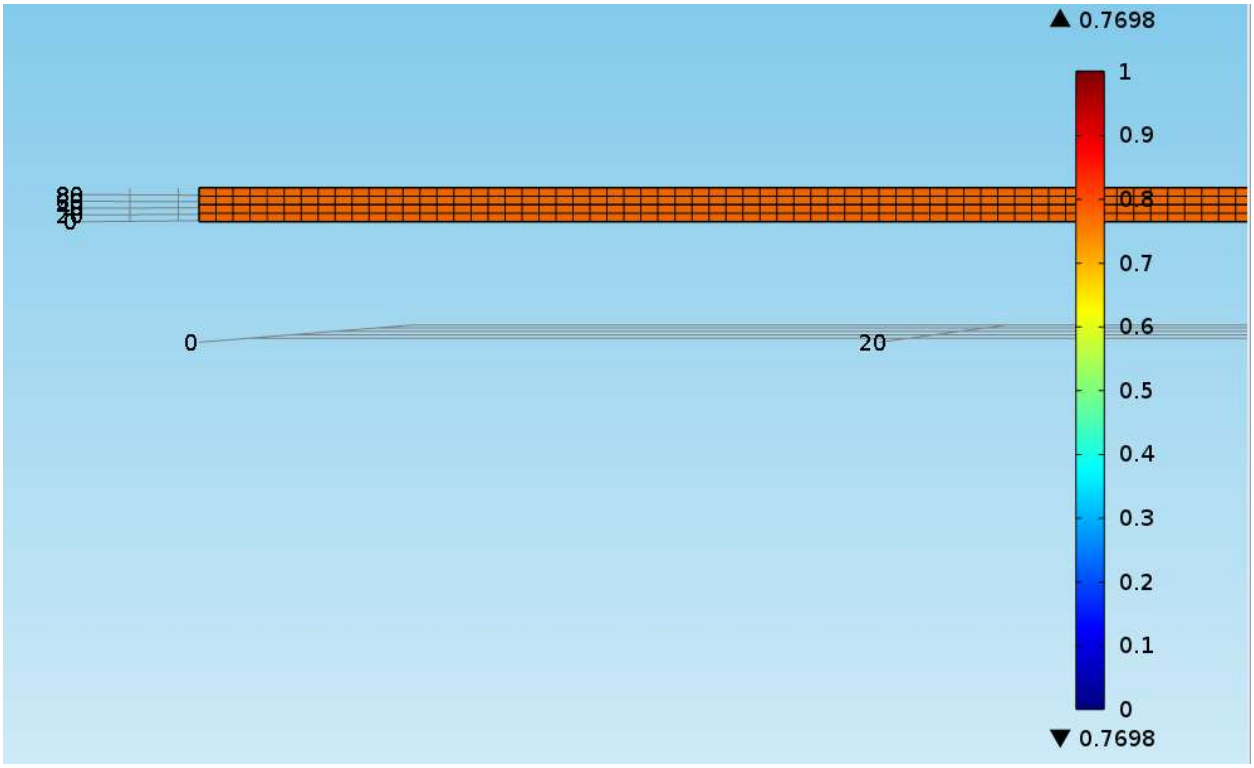


Figure 148: 3D Simply Supported Plate Model Fine Mesh Quality (Thickness)

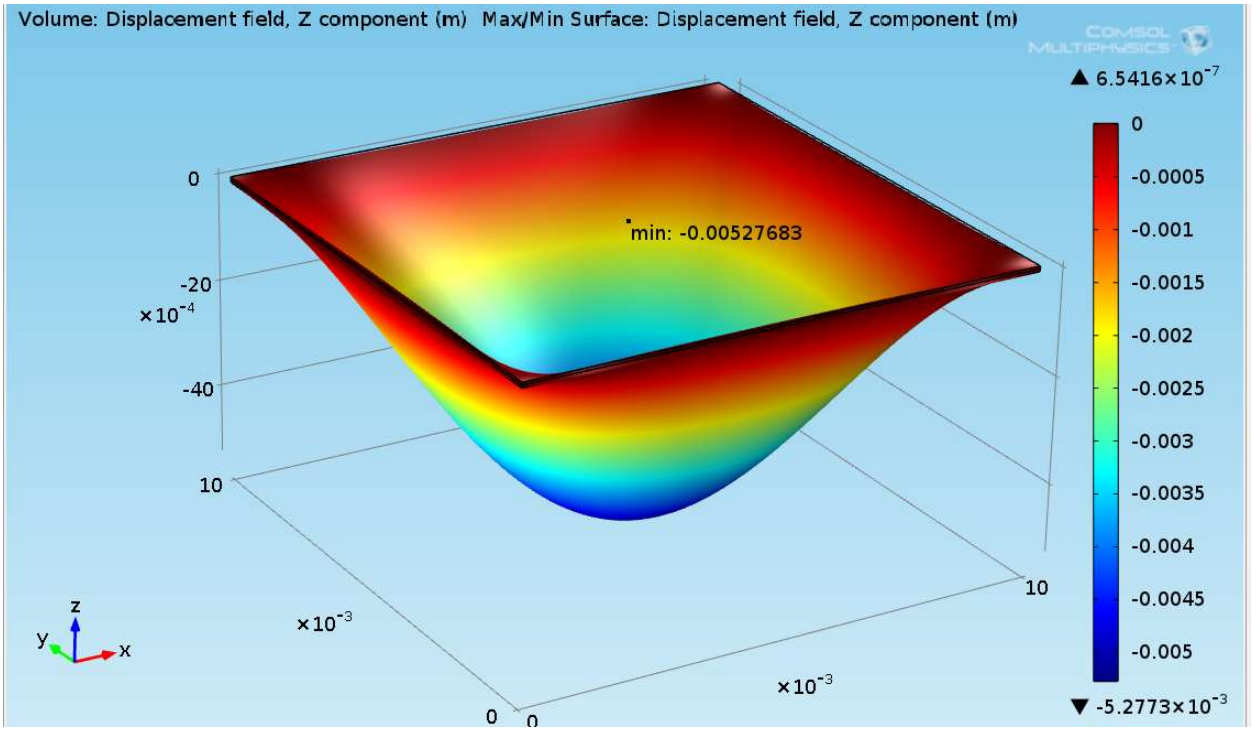


Figure 149: 3D Simply Supported Plate Model Finer Mesh Displacement Plot (Z) and Point Evaluation – Linear Discretization

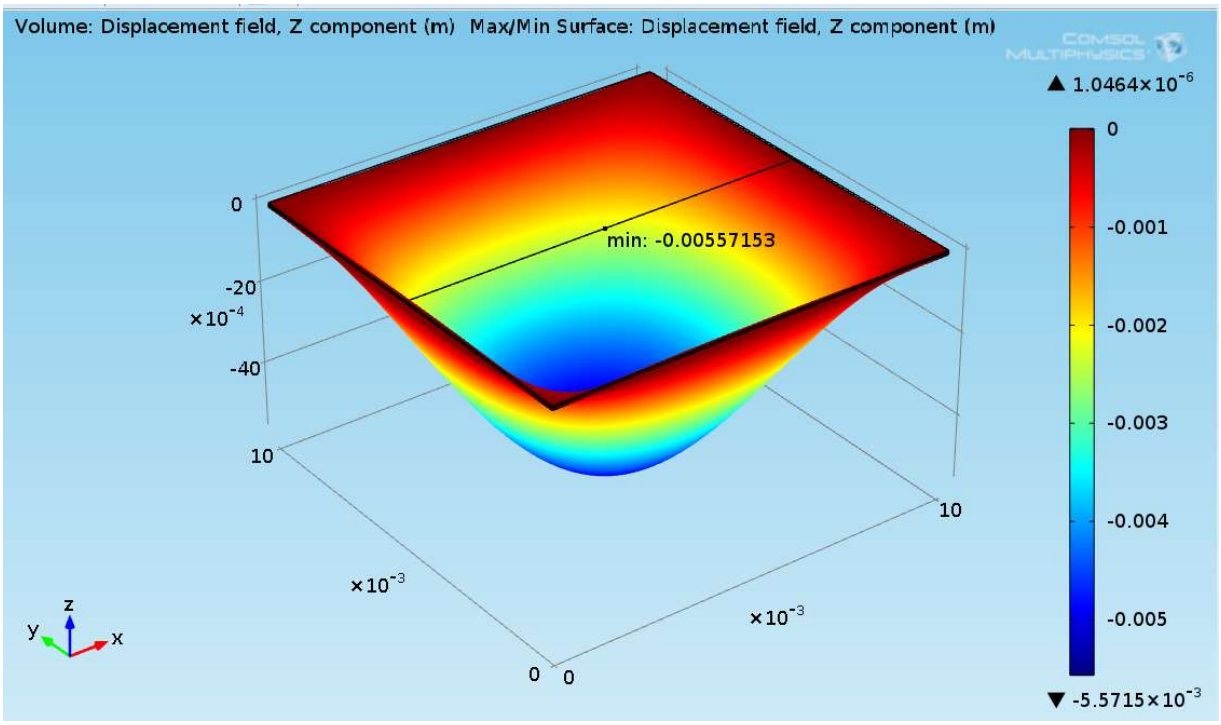


Figure 150: 3D Simply Supported Plate Model Finer Mesh Displacement Plot (Z) and Point Evaluation – Quadratic Discretization

The plots for the mesh convergence study are shaped from the data

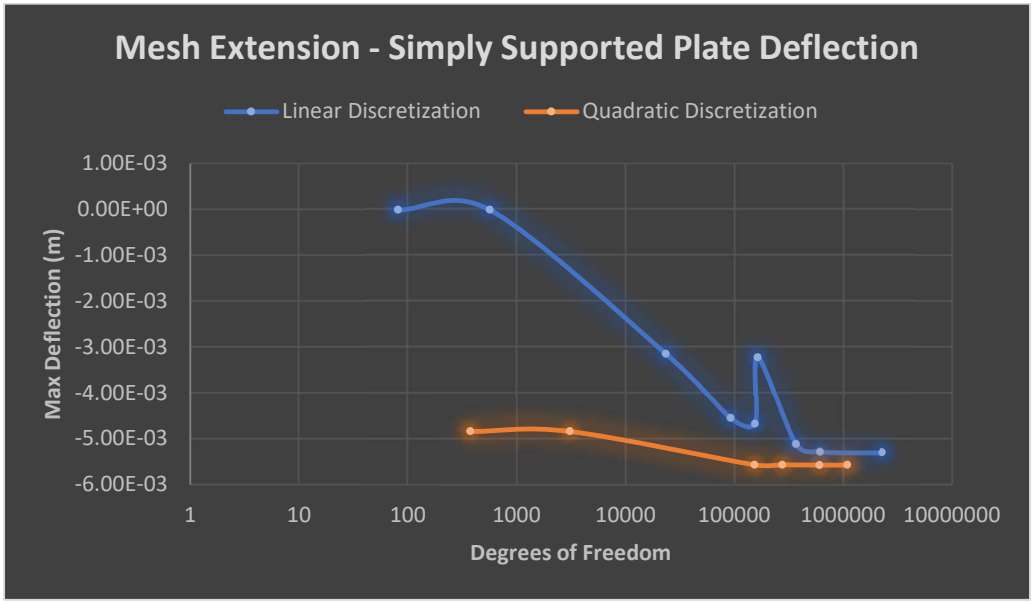


Figure 151: 3D Simply Supported Plate Model Mesh Convergence Plot

The von mises stress plot can be seen by plotting a surface plot like below

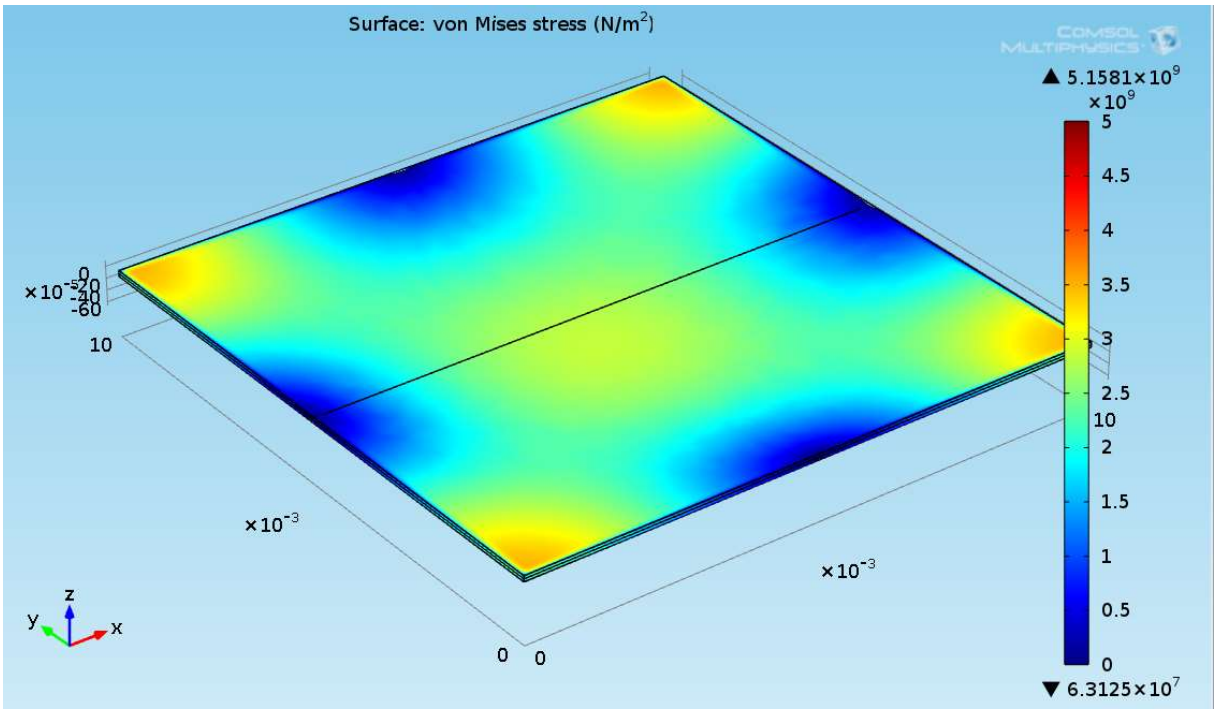


Figure 152: 3D Simply Supported Plate Model von Mises Stress Plot

Finally we can see the 1D displacement plot that can be directly compared to the Maple program.

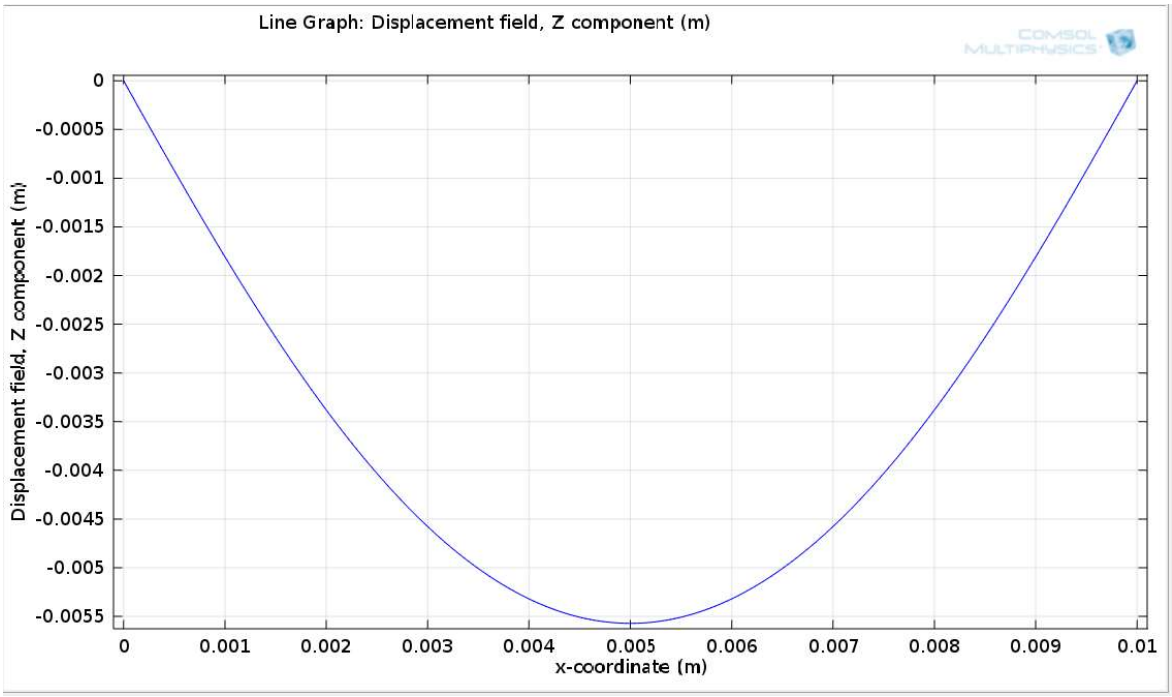


Figure 153: 3D Simply Supported Plate Model 1D Z Displacement Plot

We are able to evaluate the global error in Maple and we can also do this in COMSOL by doing a surface integration on the displacement field

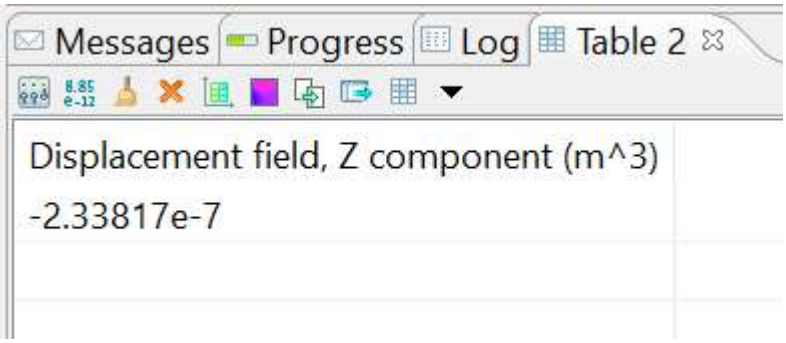


Figure 154: 3D Simply Supported Plate Global Error

Finite Element Approximation - Inconel 625 Annealed Plate (Scored and Unscored)

To accurately evaluate a rupture disc with a score, we must first evaluate the disc unscored, which is currently used in design. This will differ from the previous solid plate example since the boundary conditions differ. The plate in the previous example was simply supported, but with a rutpure disc, it will be clamped on all edges. We would like to see that implementing a score will conentrate stress and strain to the scored area and allow us to better control where the disc opens ,“flowers”, when pressurized to the point of material failure, though we are remaining in the eleastic region for our analysis which mimics the disc holding against possible pressures during operation and before burst.

We apply all of the same parameters for dimensions and materials as the simply supported disc. To constrain the plate in a clamped manner, predescribed boundary displacements of zero in the x, y, and z direction are set on all four side of the plate.

Meshing can be completed in a few different ways; mapped with a sweep and physics controlled tetrahedral with distributions through the thickness and to the boundaries. We will use both, but we will consider the quadratic discretization for computation time saving and higher accuracy. From the many previous examples, we have proven that the degrees of freedom increase substantially from linear to quadratic to cubic discretization. The quadratic does a sufficient job for our modeling, but we will still do a convergence study to let us know when our answer is reliable.

Mesh Study									
Simply supported plate									
Discretization - Quadratic									
Mesh Characteristics	Study		1	2	3	4	5	6	tetrahedral
	Number of Divisions	Width	2	2	50	50	100	100	
		Length	2	2	50	50	100	100	
		Thickness	1	4	1	4	1	4	
	DOF		375	675	153015	275427	606015	1090827	1375296
Results	Solve Time(s)		1	1	22	53	119	299	157
	Displacement (m)		-1.67753E-05	-1.67753E-05	-0.001719	-0.001719	-0.001722	-0.001722	-0.00172279

Figure 155: Clamped Plate – No Score Mesh Study Table

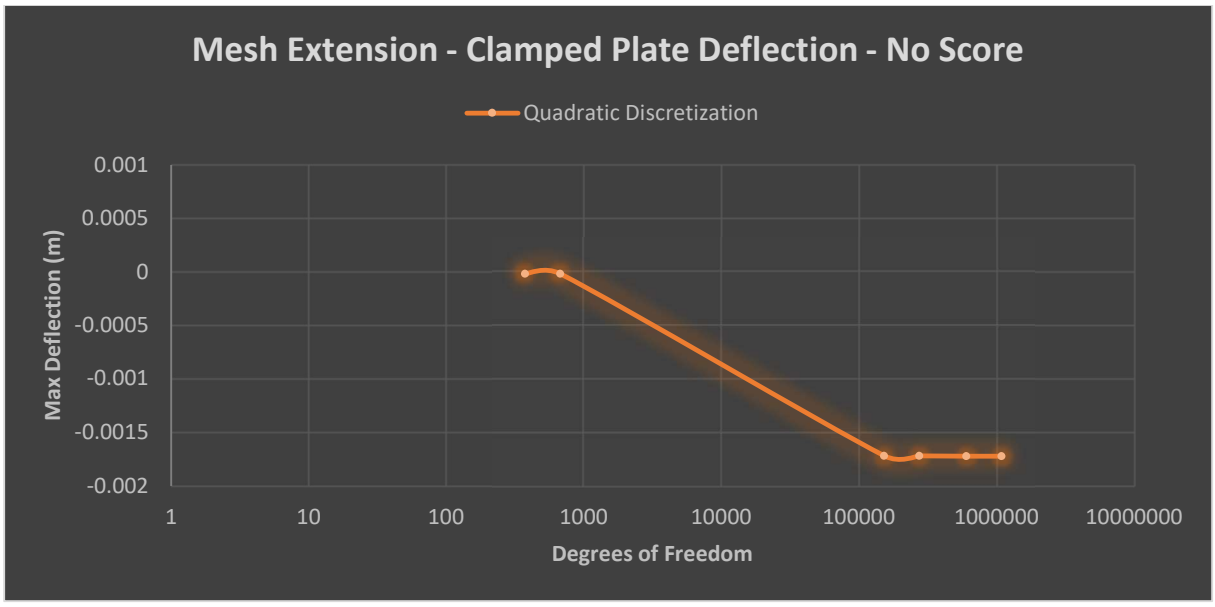


Figure 156: Clamped Plate – No Score Mesh Convergence Plot

Shown below is the finest mesh quality used in the convergence study and its displacement plot

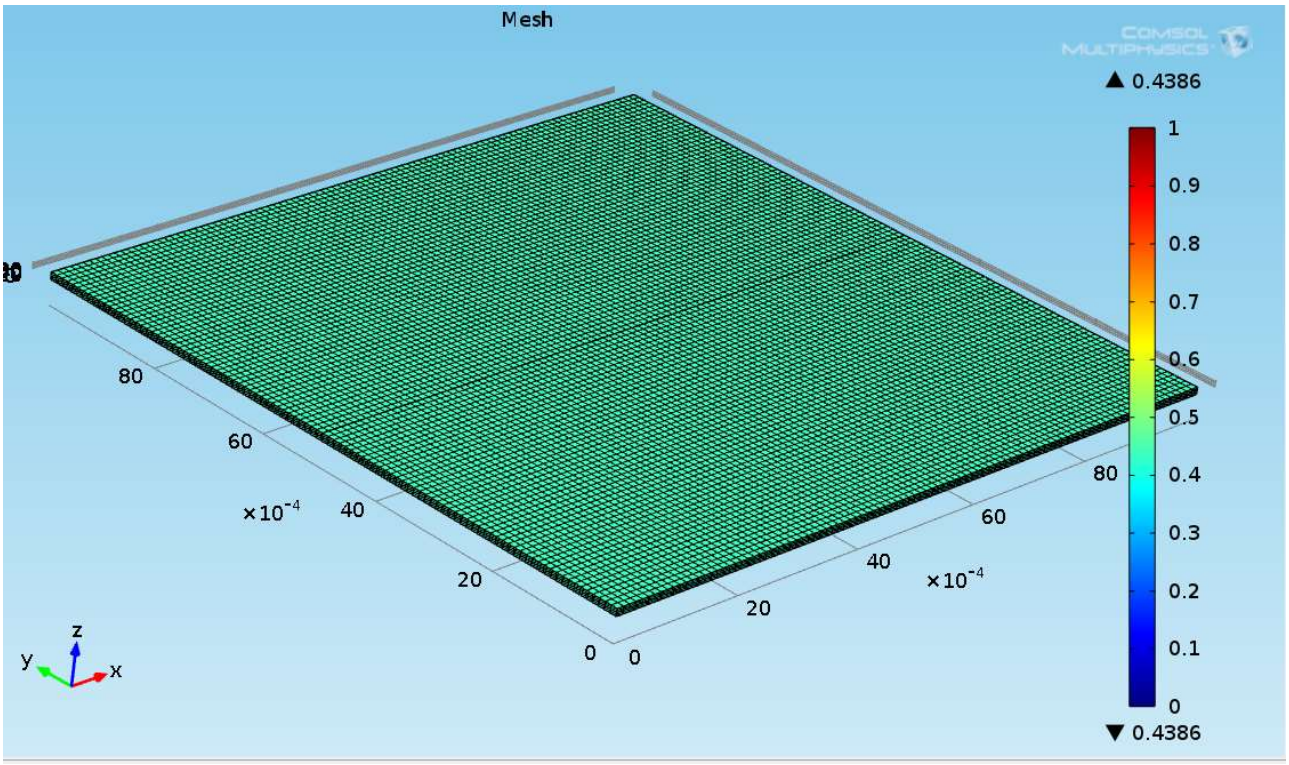


Figure 157: Clamped Plate – No Score Fine Convergence Mesh Quality (Mapped)

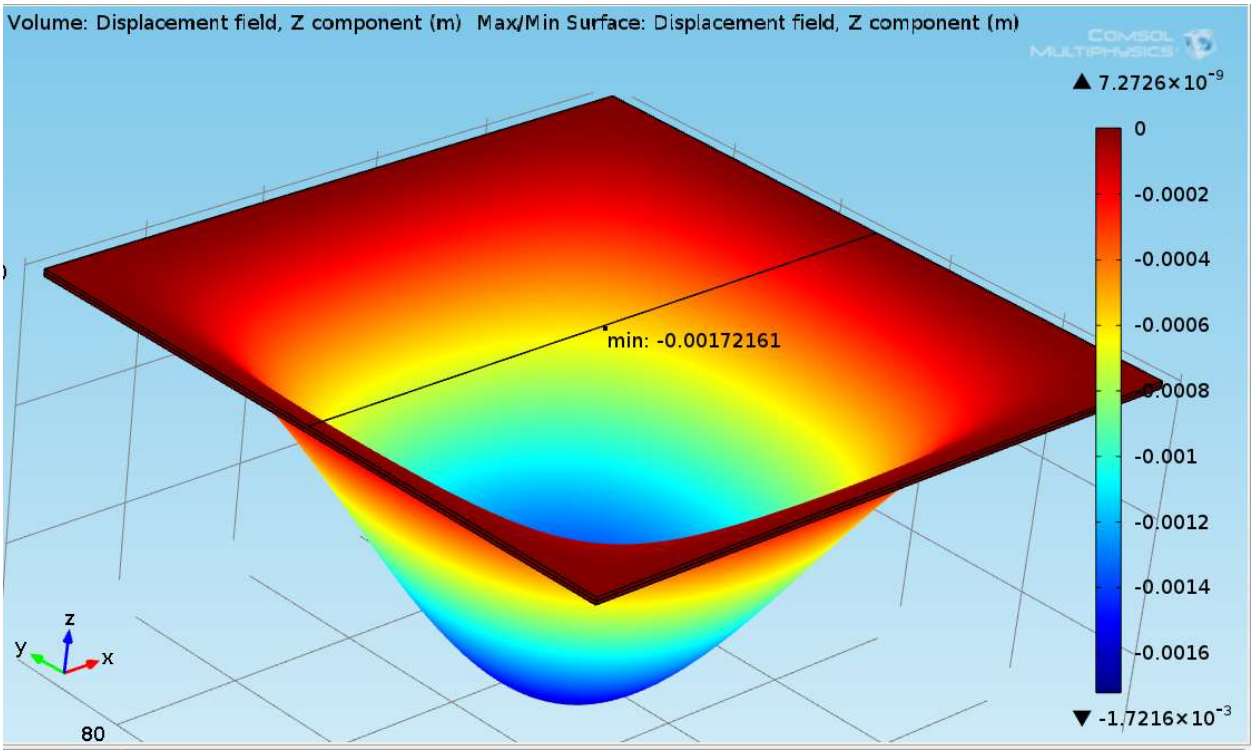


Figure 158: 3D Clamped Plate Model Finer Mesh Displacement Plot (Z) and Point Evaluation – Quadratic Discretization

Using the tetrahedral physics controlled mesh with refinement through the thickness and out to the boundaries yields the following study

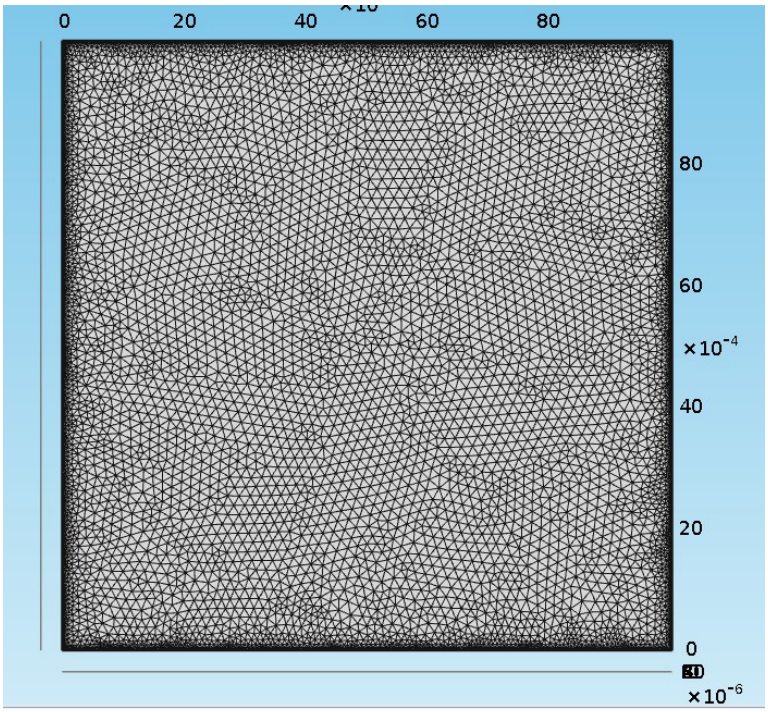


Figure 159: 3D Clamped Plate Model Tetrahedral Distributed Mesh

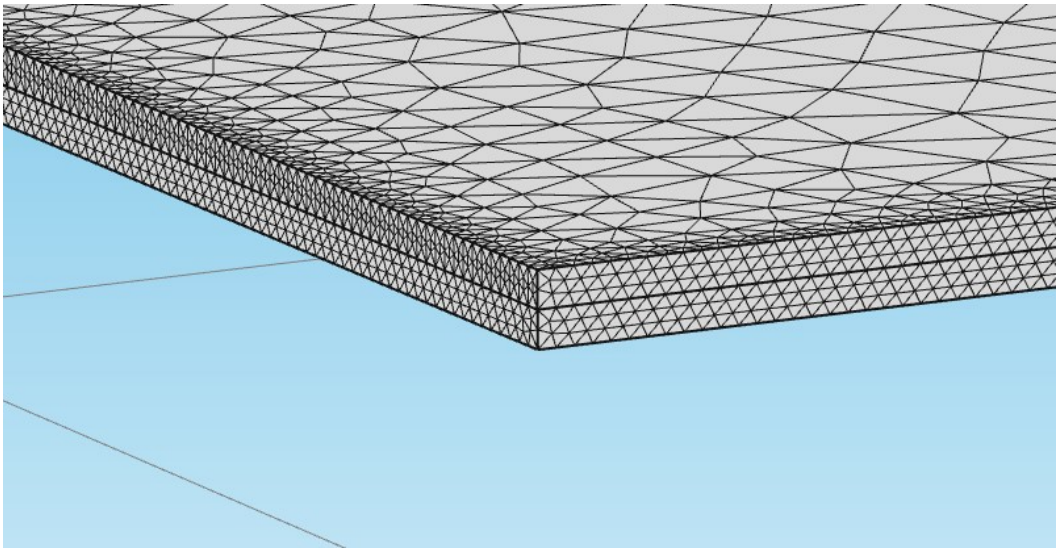


Figure 160: 3D Clamped Plate Model Tetrahedral Distributed Mesh (Thickness)

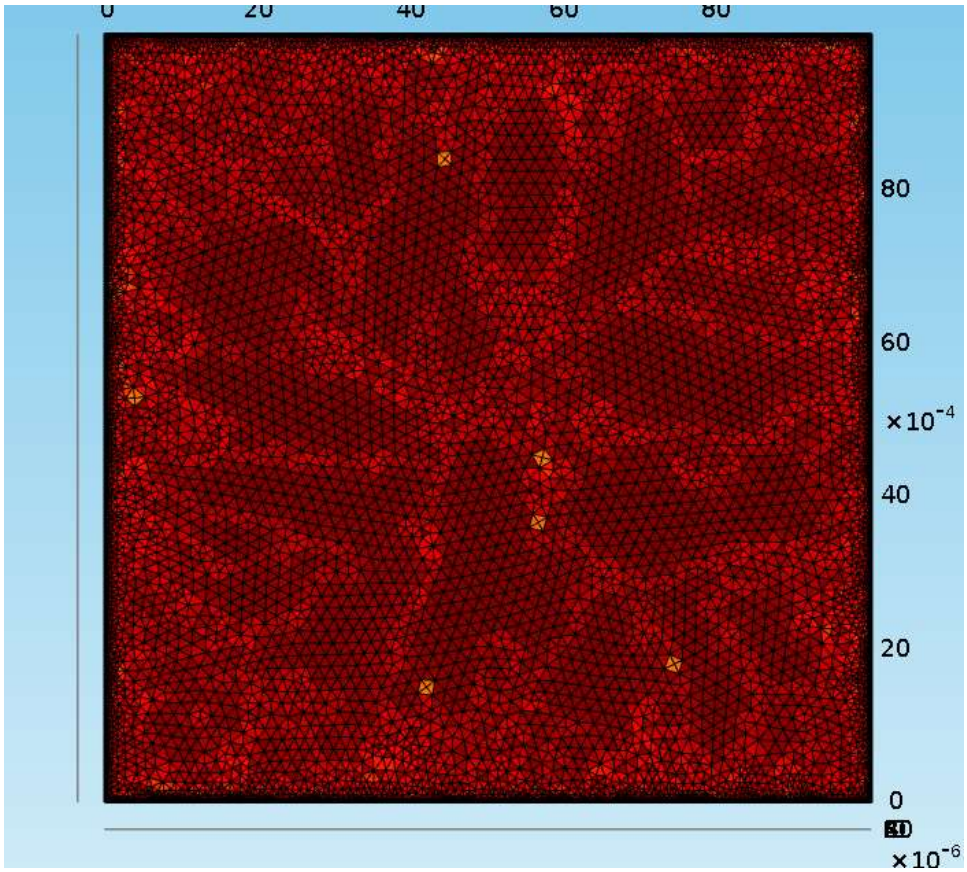


Figure 161: 3D Clamped Plate Model Tetrahedral Distributed Mesh Quality (XY)

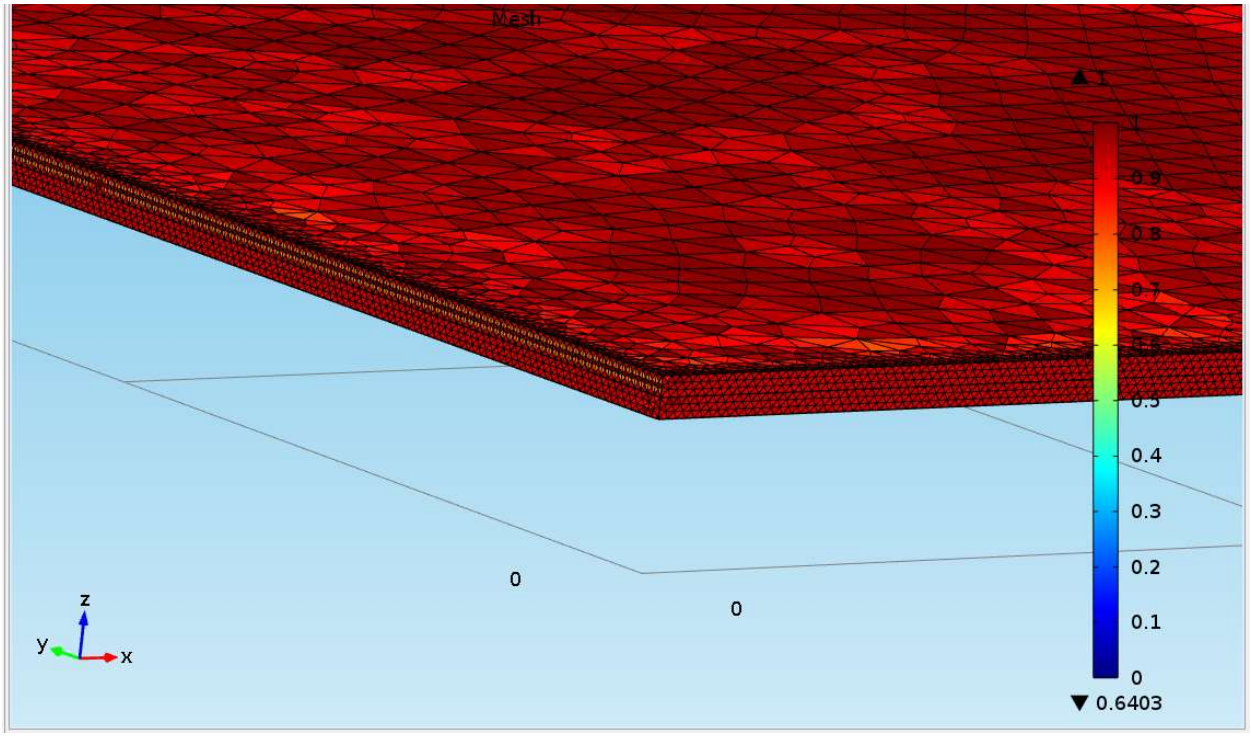


Figure 162: 3D Clamped Plate Model Tetrahedral Distributed Mesh Quality (Thickness)

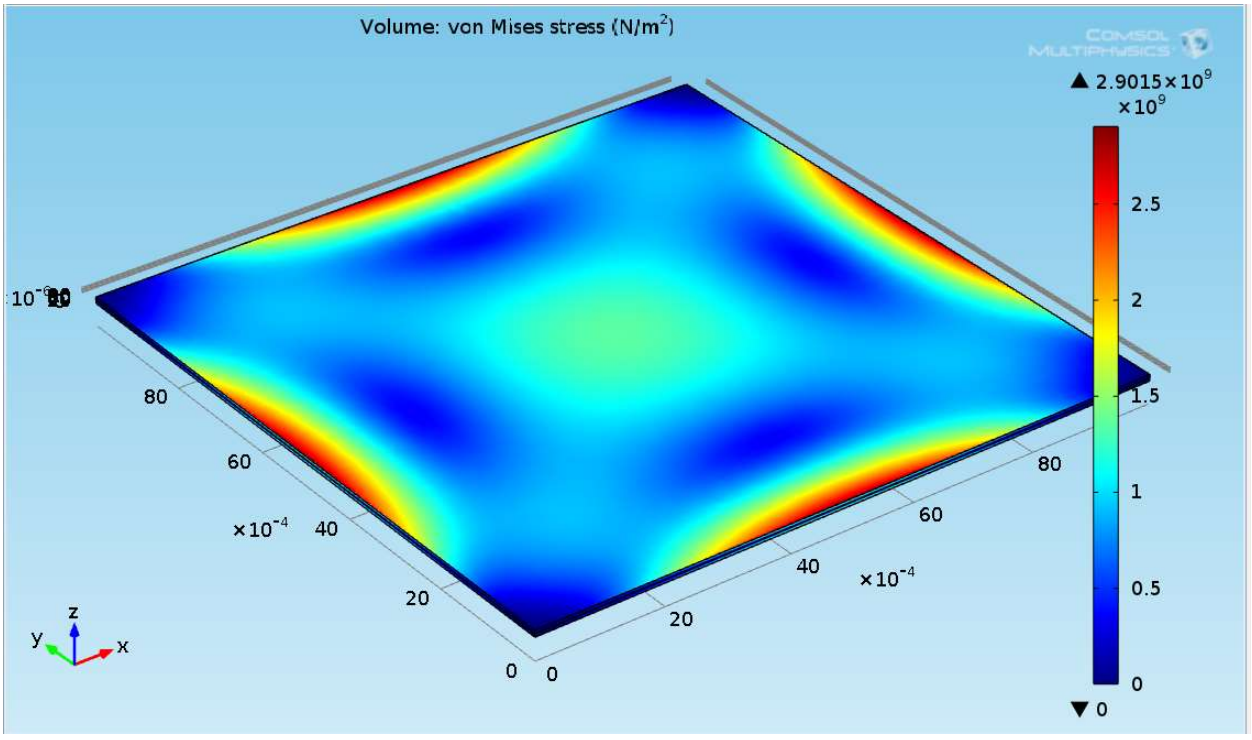


Figure 163: 3D Clamped Plate Model Tetrahedral Distributed Mesh – von Mises Stress 3D Plot

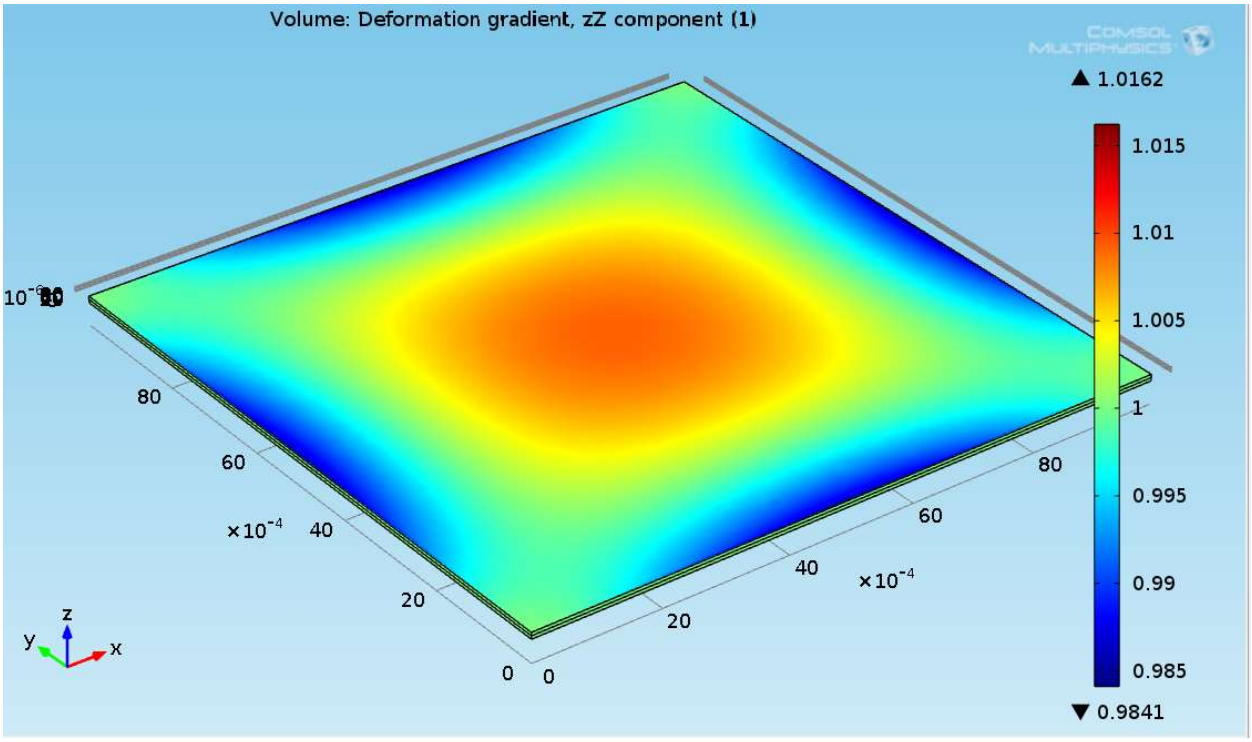


Figure 164: 3D Clamped Plate Model Tetrahedral Distributed Mesh – Strain zZ-Deformation 3D Plot

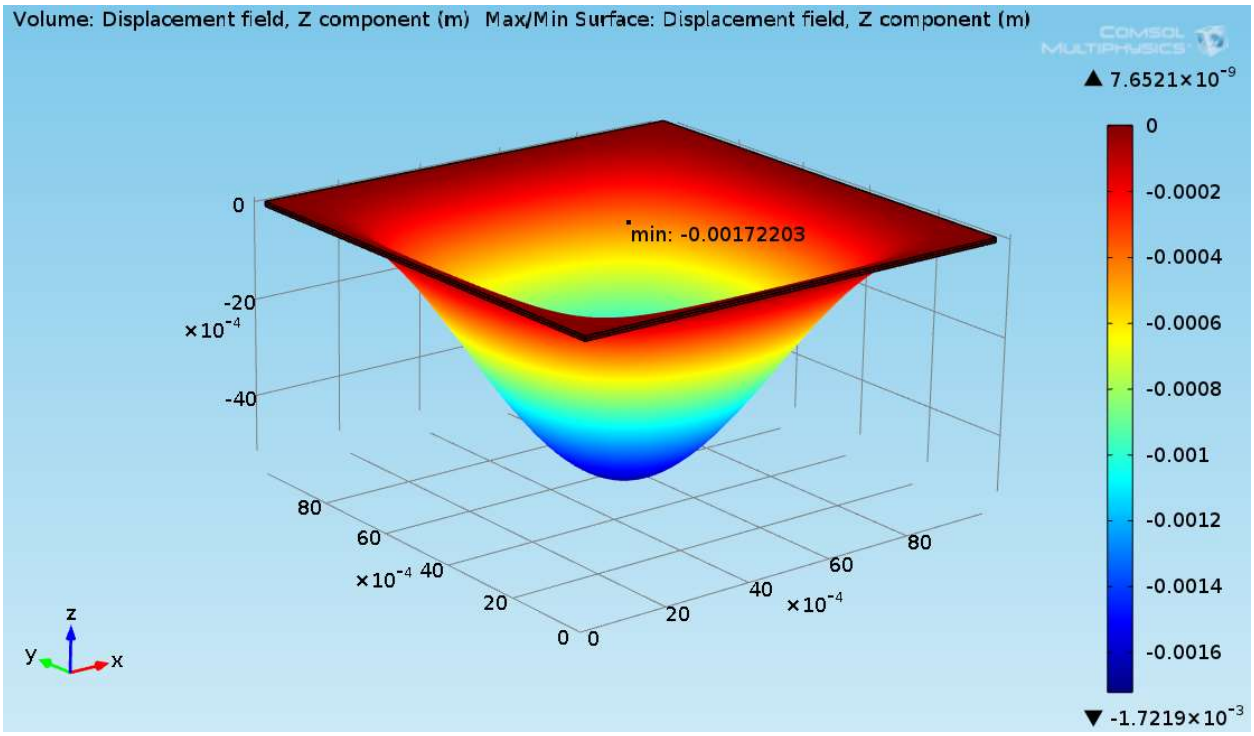


Figure 165: 3D Clamped Plate Model Tetrahedral Distributed Mesh – Deflection Plot and Point Evaluation 3D Plot

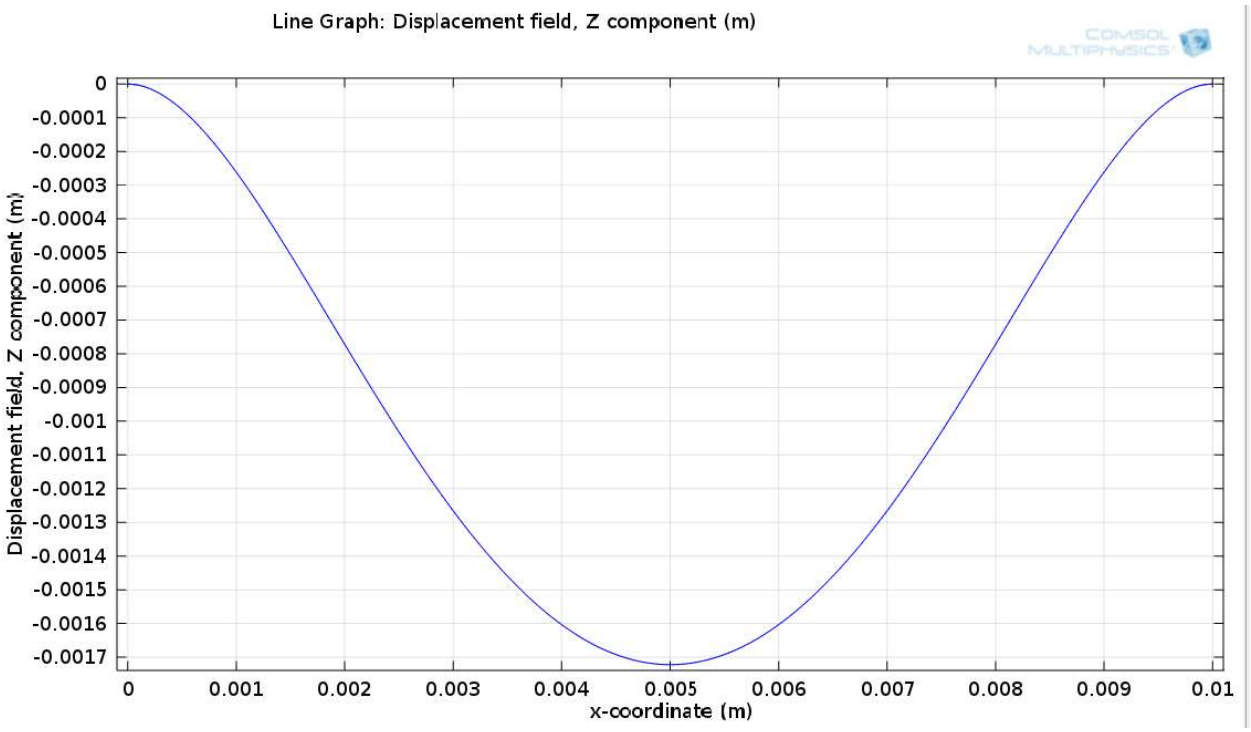


Figure 166: 3D Clamped Plate Model Tetrahedral Distributed Mesh – Z-Displacement 1D Plot

We can see, as expected, the stress and strain is most prevalent at the boundaries of the disc, but also at the center. This is where we would like to apply the score for symmetry and also for centered flow of the fluid after burst.

We can also see that the tetrahedral meshing matches the deflection solution for the finest mapped mesh, so it can be seen as a reliable method of meshing.

Moving onto the scored plate, the parameters are the same, except we introduce a new parameter t1 and h2 for the score parameters.

Parameters

Name	Expression	Value	Description
P1	-1e6	-1.0000...	
E1	8.14e10	8.1400E...	
nu1	.278	0.27800	
rho1	8440	8440.0	
L1	.01	0.01000...	
h1	.0001	1.0000E...	
b1	.01	0.01000...	
t1	.002	0.00200...	

Figure 167: 3D Clamped Scored Plate Model Parameters

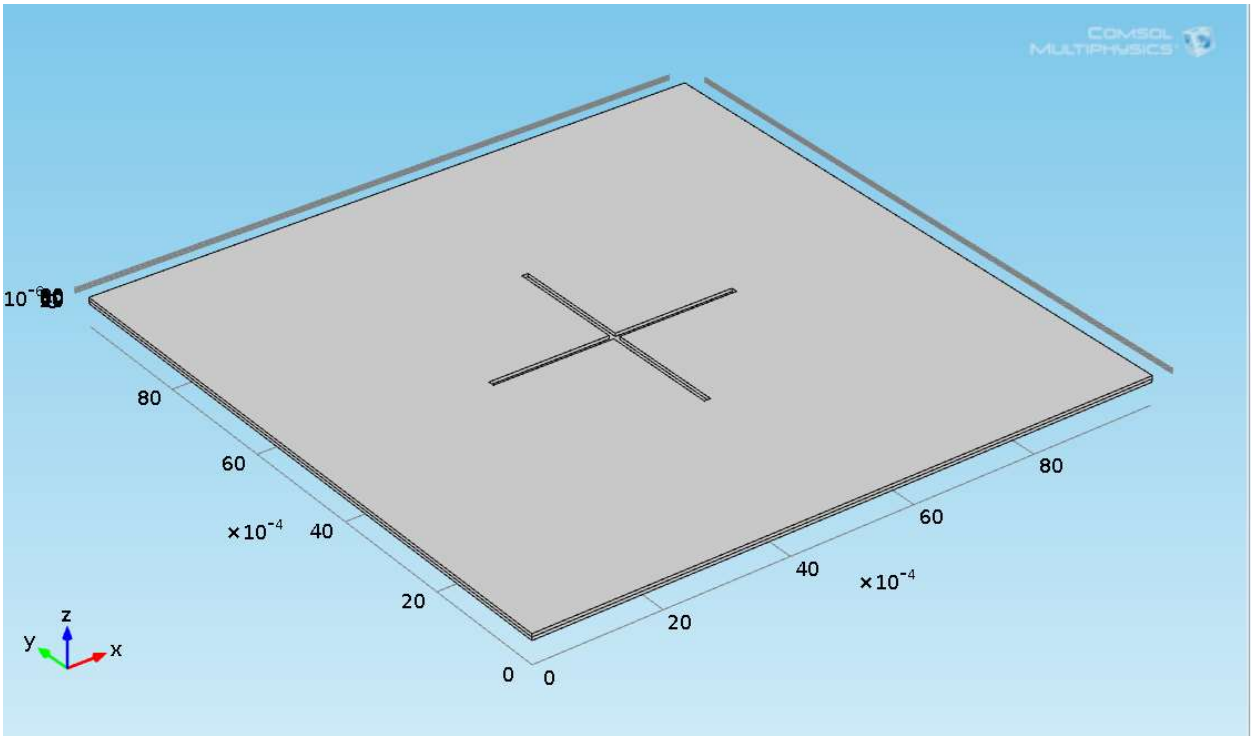


Figure 168: 3D Clamped Scored Plate Model

Applying the material properties, boundary constraints and surface load as shown below

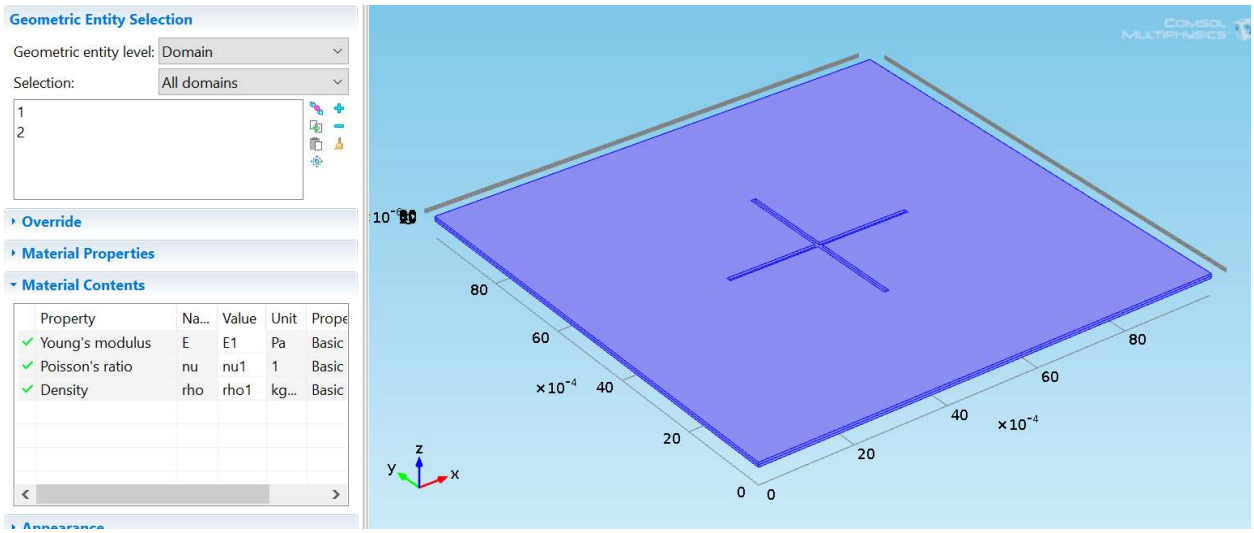


Figure 169: 3D Clamped Scored Plate Model Material Application

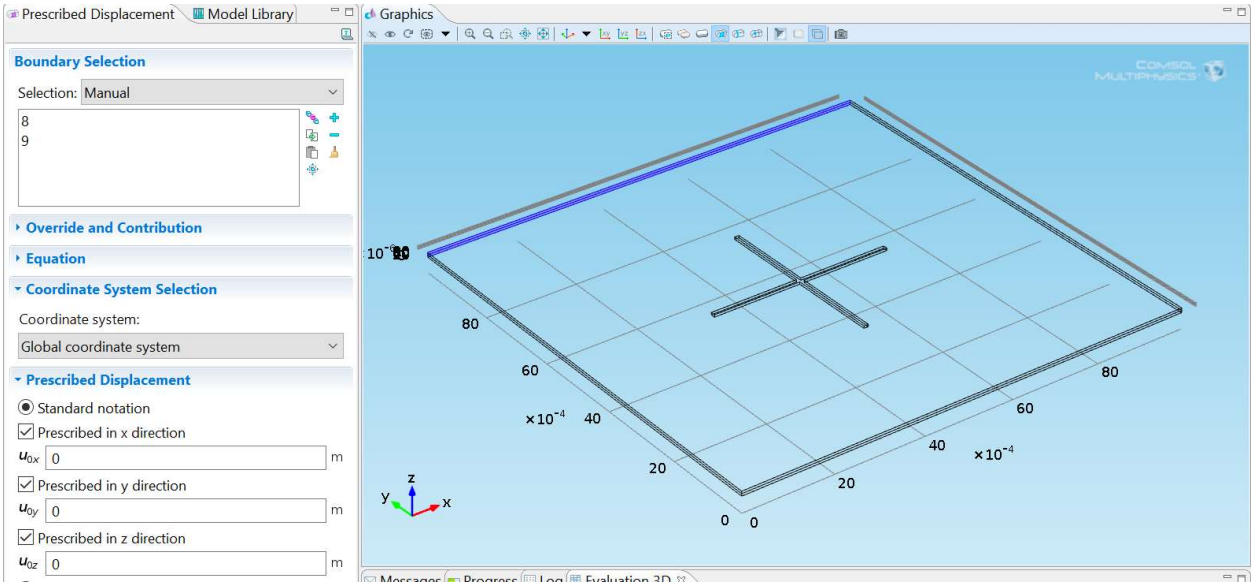


Figure 170: 3D Clamped Scored Plate Model – Clamped Constraint (1)

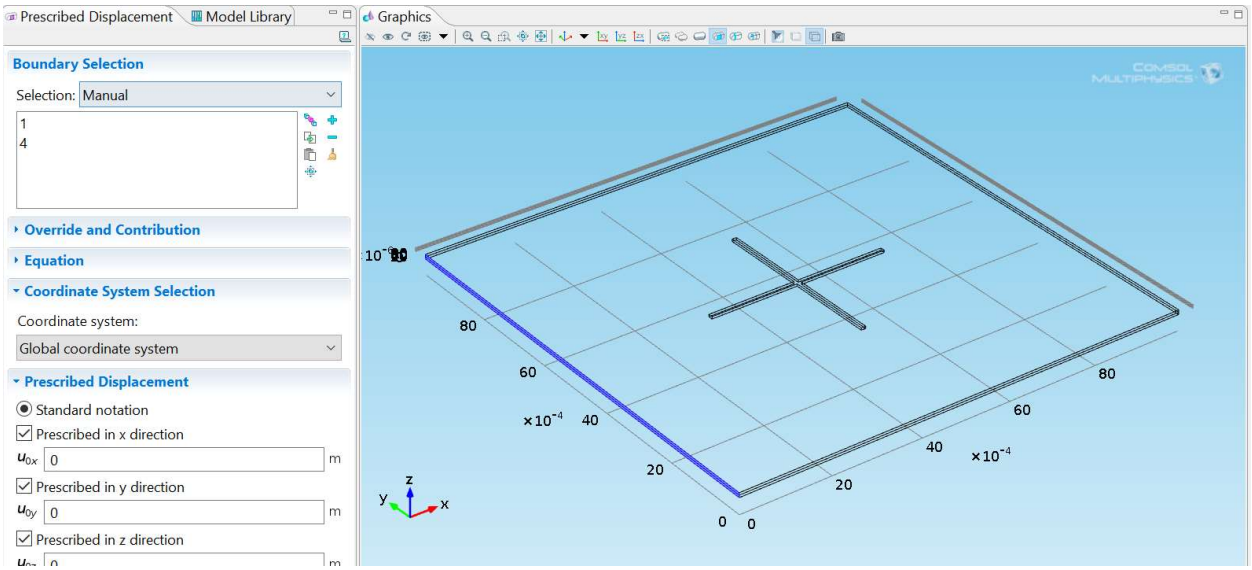


Figure 171: 3D Clamped Scored Plate Model – Clamped Constraint (2)

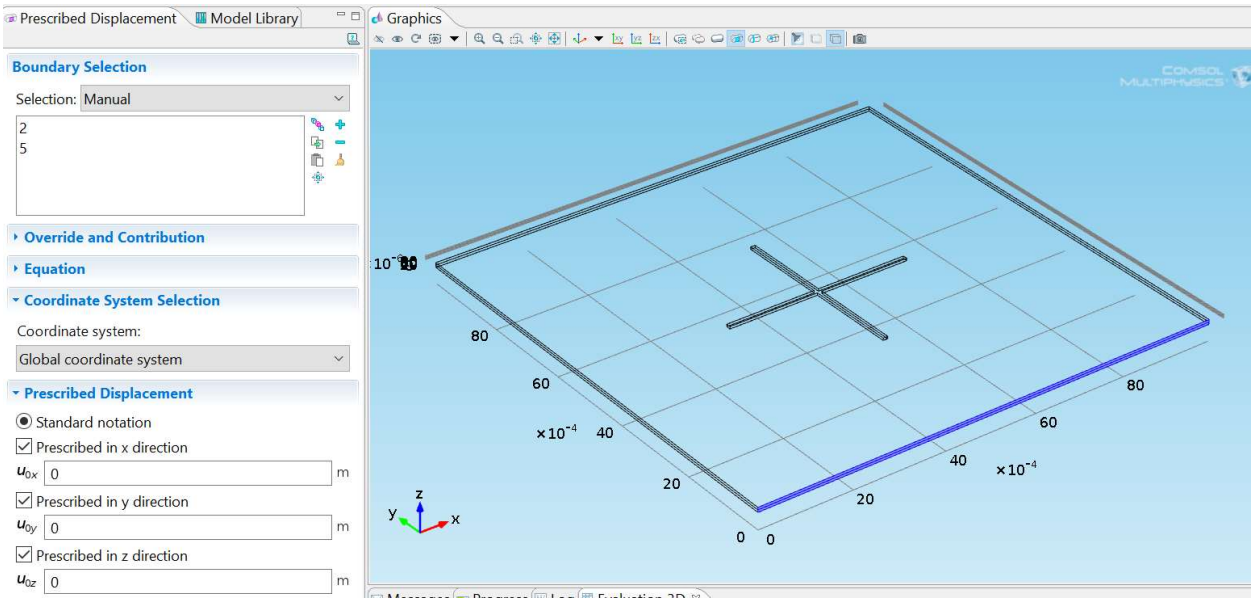


Figure 172: 3D Clamped Scored Plate Model – Clamped Constraint (3)

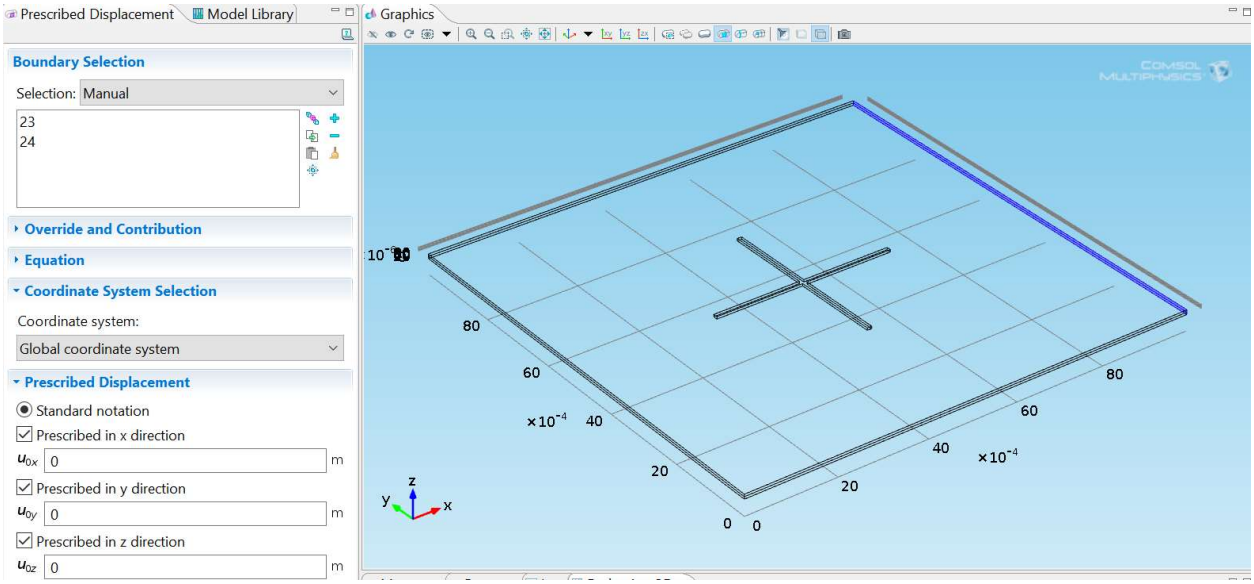


Figure 173: 3D Clamped Scored Plate Model – Clamped Constraint (4)

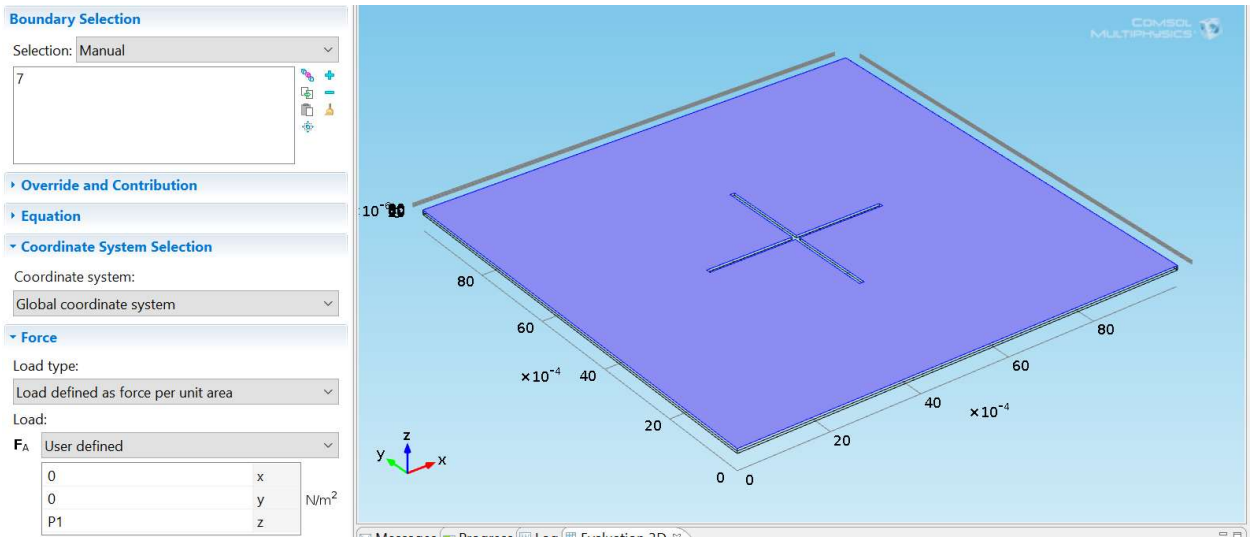


Figure 174: 3D Clamped Scored Plate Model – Load Application

The mesh study will be conducted with physics controlled tetrahedral meshing with the addition of refinement through the thickness, to the outer edges, and around the score of the disc all completed with quadratic discretization. The physics controlled mesh, quadratic discretization, and tetrahedral elements have proven in the previous examples to provide quality mesh elements and answers with low error with high degrees of freedom.

The mesh extension study can be seen in the table below.

Mesh Study									
Simply suppoted Plate									
Discretization - Quadratic									
Mesh Characteristics	Study		1	2	3	4	5	6	7	8
	Number of Divisions/Physics Controlled	Face	Extra Coarse	Coarse	Normal	Extra coarse	Coarse	Normal	Finer	Finer
		Score	5	5	5	30	30	30	5	30
		Thickness	1	1	1	4	4	4	1	4
	DOF		99117	154347	278127	552849	664101	823665	1406127	2028323
Results	Solve Time(s)		8	15	27	33	44	96	251	467
	Displacement (m)		-0.0016720	-0.0018330	-0.0018753	-0.0017506	-0.0018867	-0.0018986	-0.001907	-0.0019099

Figure 175: 3D Clamped Scored Plate Model – Mesh Extension Study

The mesh convergence plot is constructed

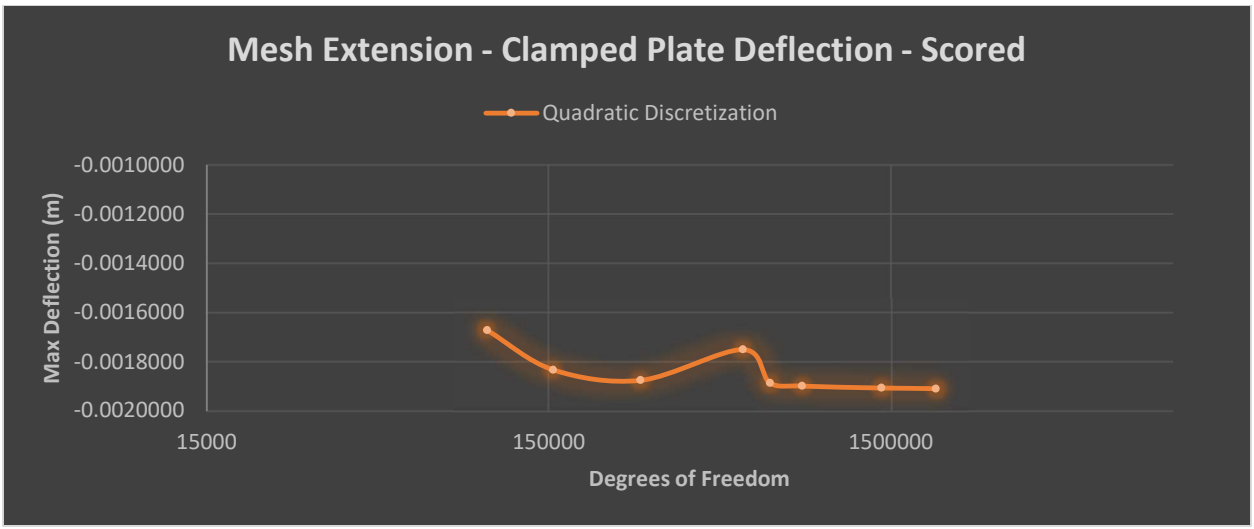


Figure 176: 3D Clamped Scored Plate Model – Mesh Extension Study

Looking at both ends of the spectrum, we can view the results of the extra coarse, 1 element thick, 5 element score edge refinement study and move to the more refined mesh. The mesh and quality of the mesh with the least degrees of freedom is shown here

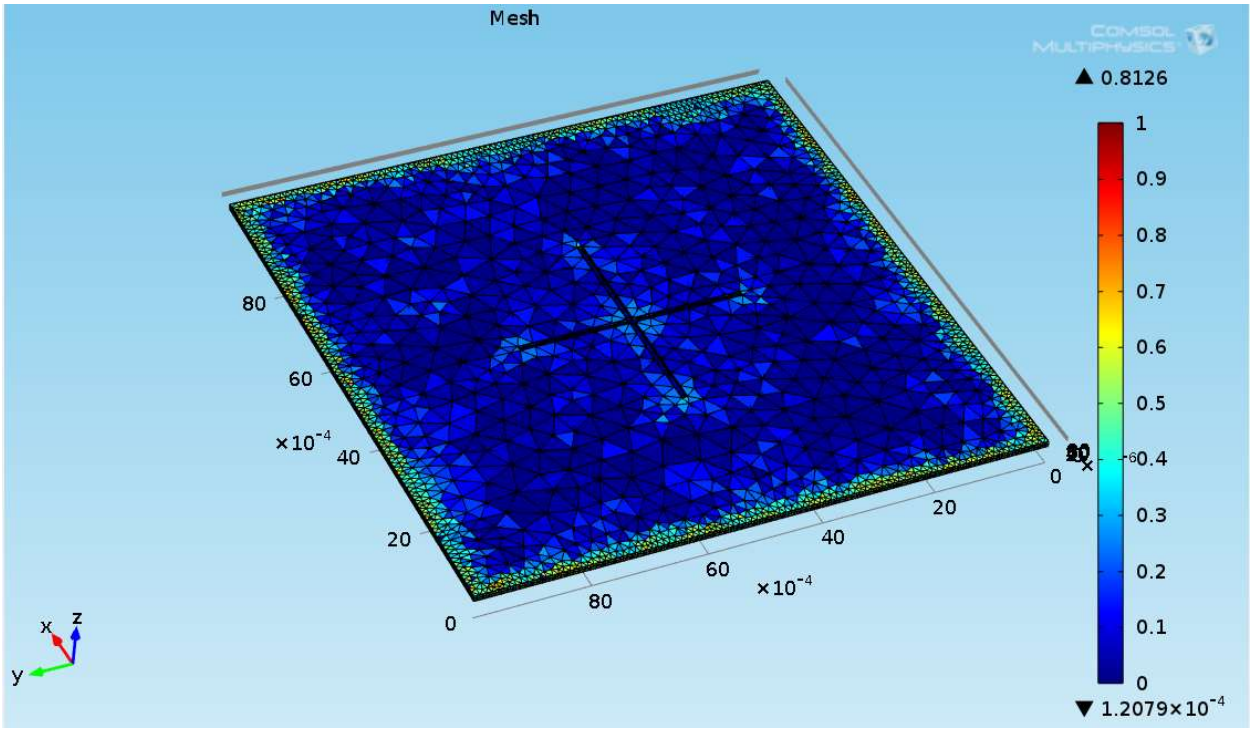


Figure 177: 3D Clamped Scored Plate Model – Extra Coarse Mesh Quality (Isometric View)

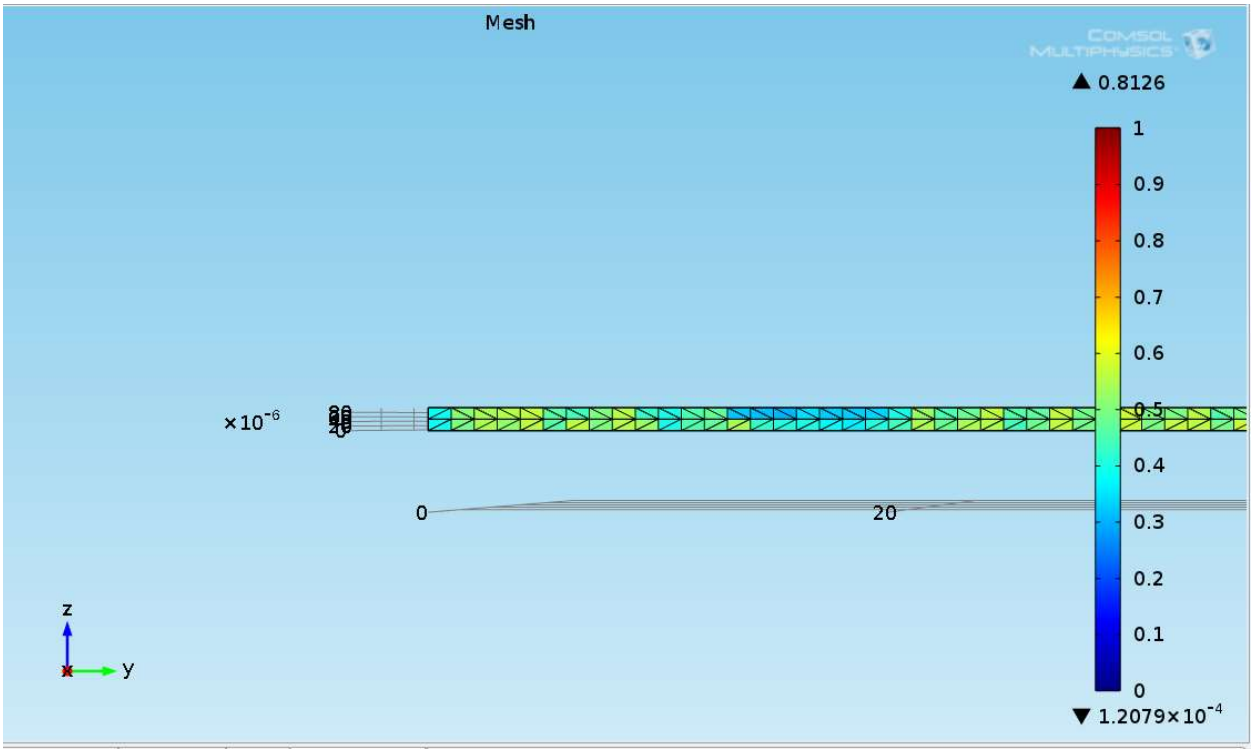


Figure 178: 3D Clamped Scored Plate Model – Extra Coarse Mesh Quality (Thickness View)

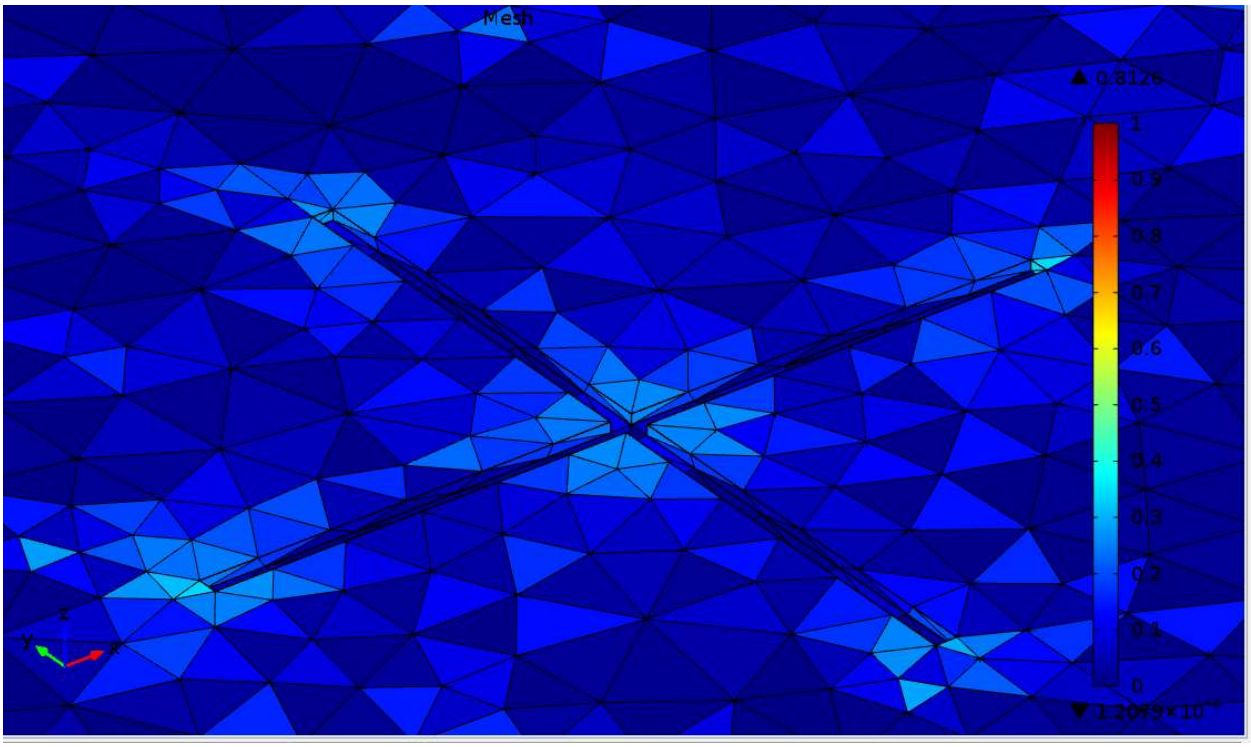


Figure 179: 3D Clamped Scored Plate Model – Extra Coarse Mesh Quality (Score View)

The results of the deflection field in the z component as well as the point evaluation is found to be

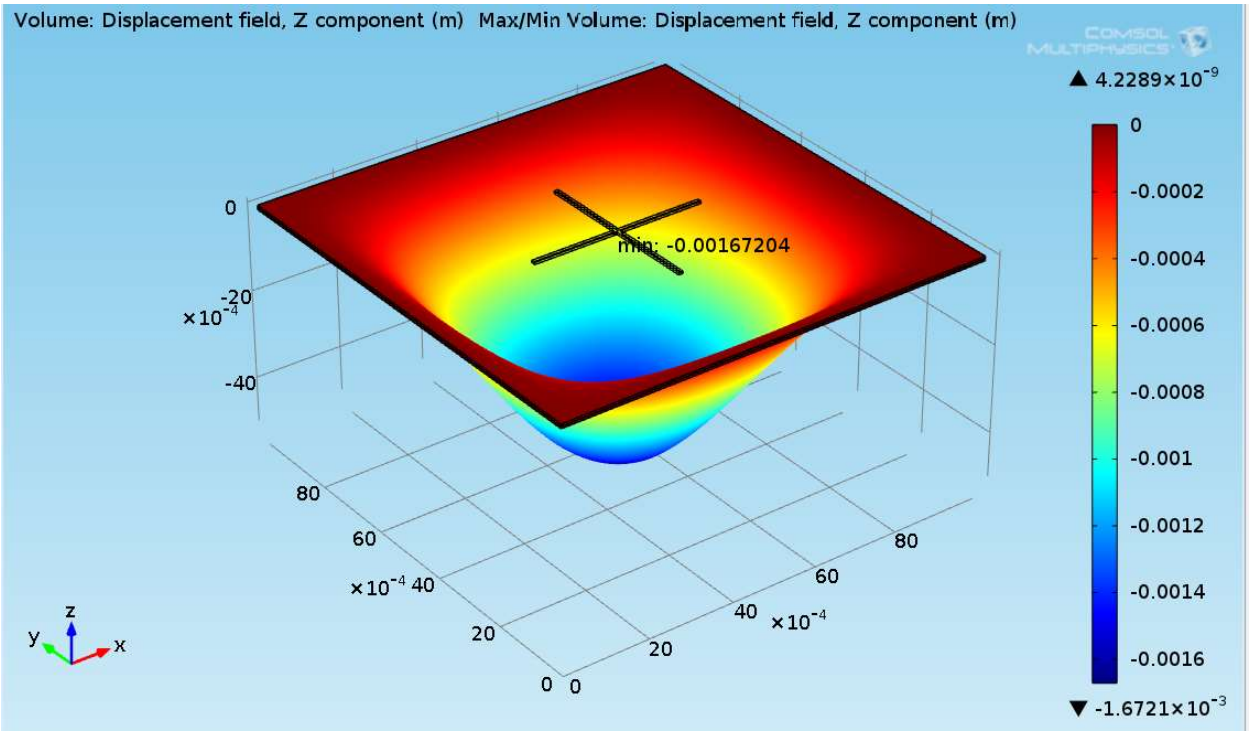


Figure 180: 3D Clamped Scored Plate Model – Extra Coarse Mesh Displacement Field (Z) and Point Evaluation

The goal is to not only view the deflection but also the stress and strain when introducing the score, so the plots below reflect this

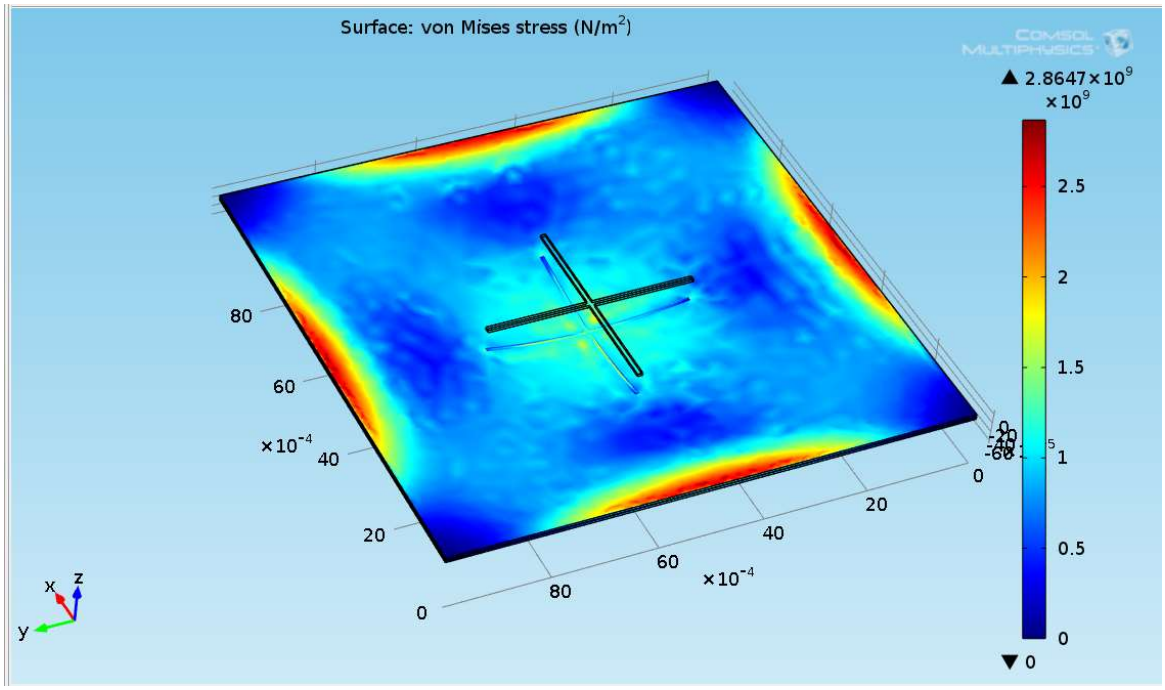


Figure 181: 3D Clamped Scored Plate Model – Extra Coarse Mesh von Mises Stress Plot

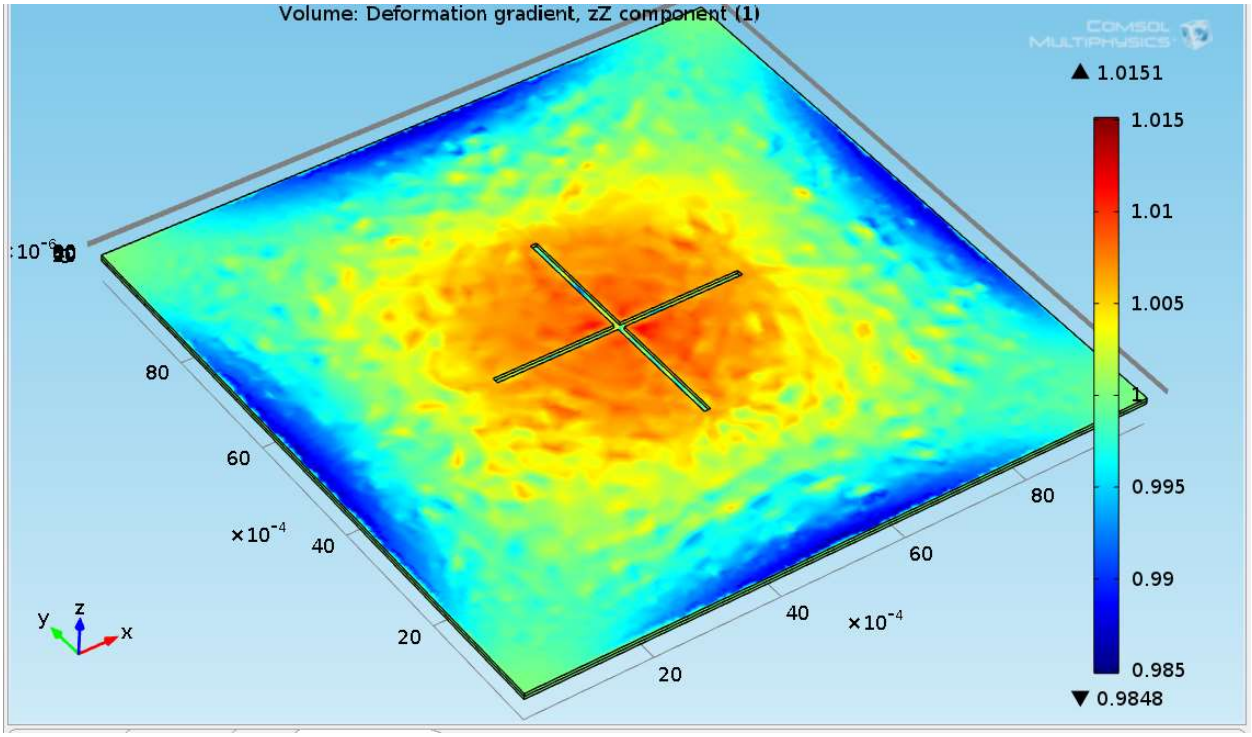


Figure 182: 3D Clamped Scored Plate Model – Extra Coarse Mesh Strain/Deformation (zZ) Plot

Going to the other end of the spectrum of the mesh extension study, we have a mesh quality as shown

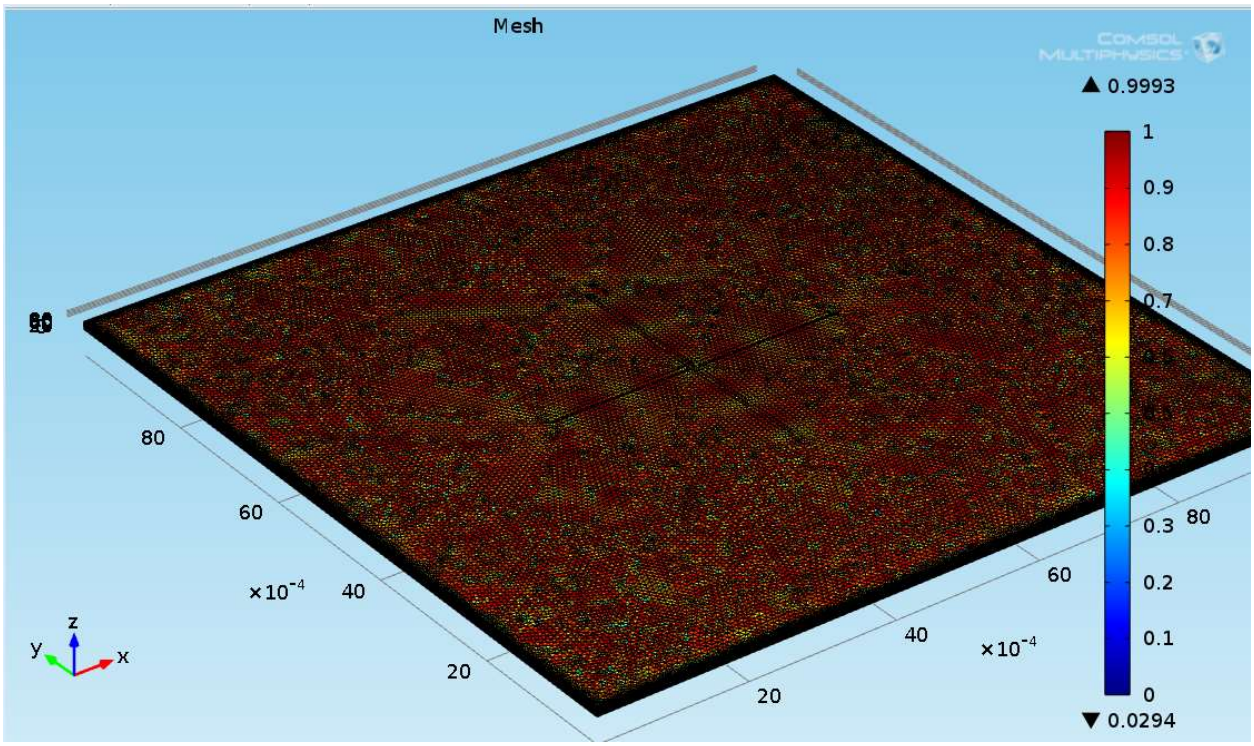


Figure 183: 3D Clamped Scored Plate Model – Finer Mesh Quality (Isometric View)

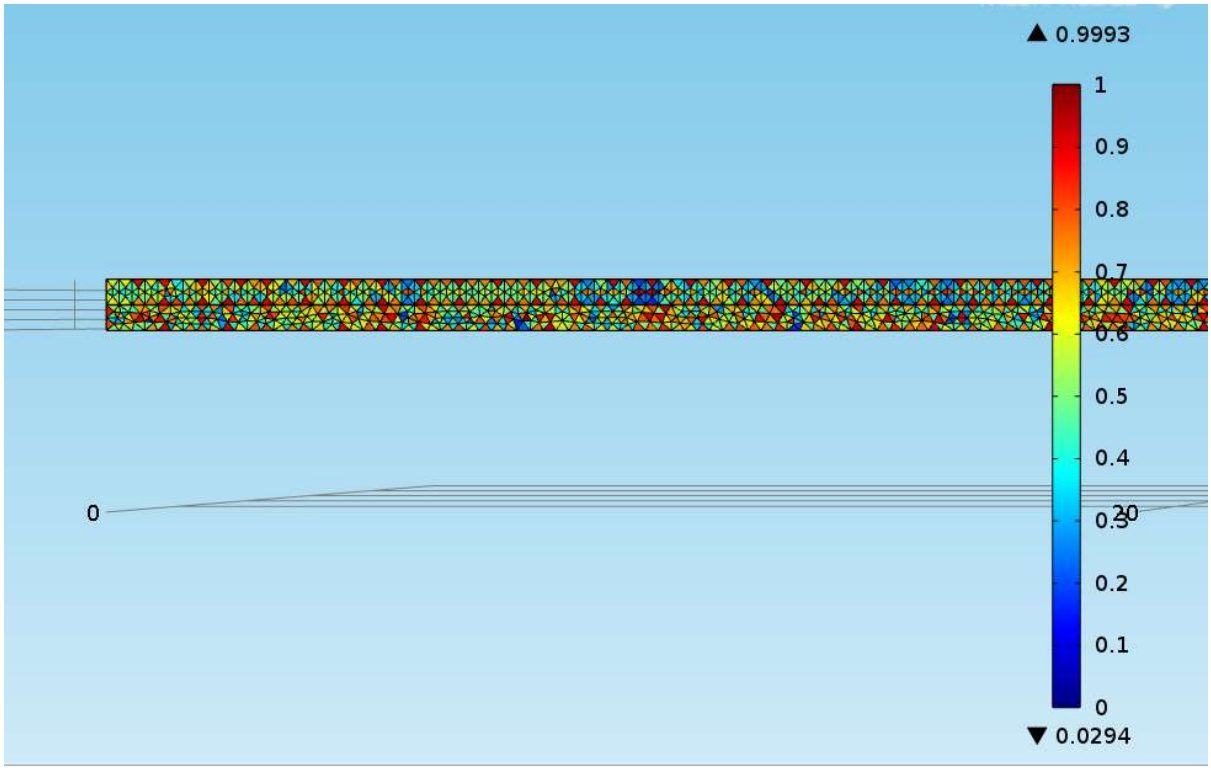


Figure 184: 3D Clamped Scored Plate Model – Finer Mesh Quality (Thickness View)

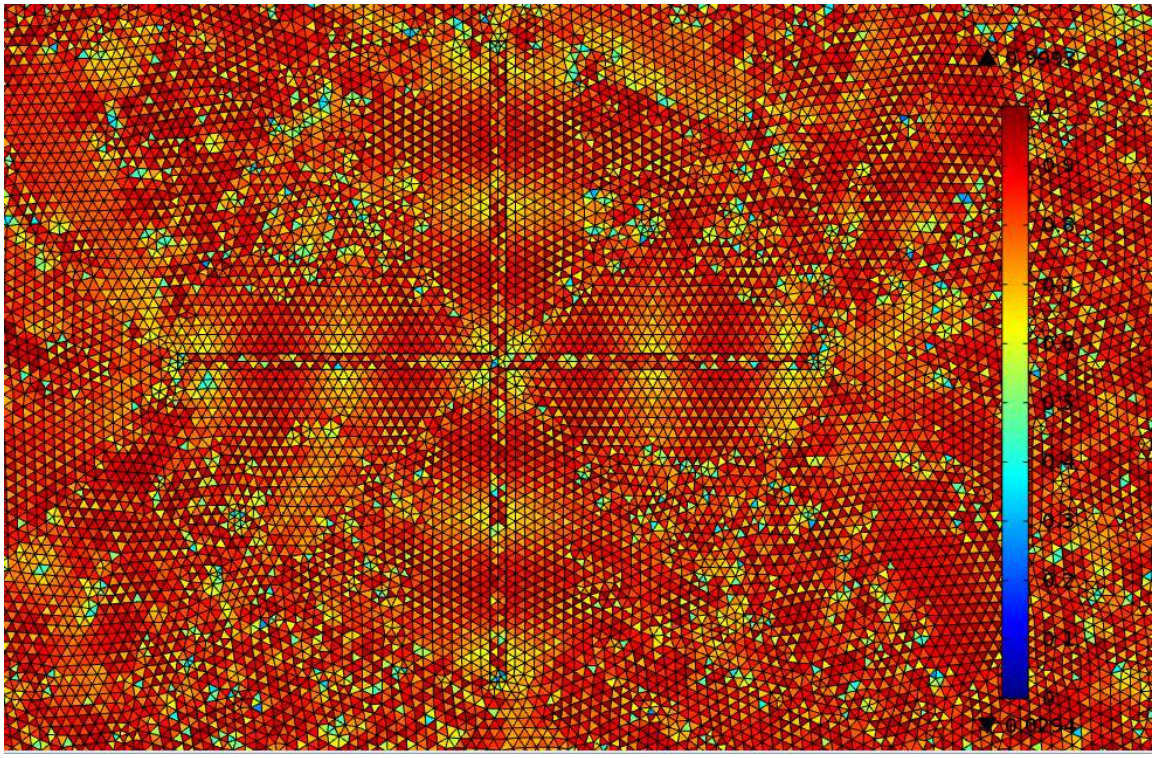


Figure 185: 3D Clamped Scored Plate Model – Finer Mesh Quality (Score View_1)

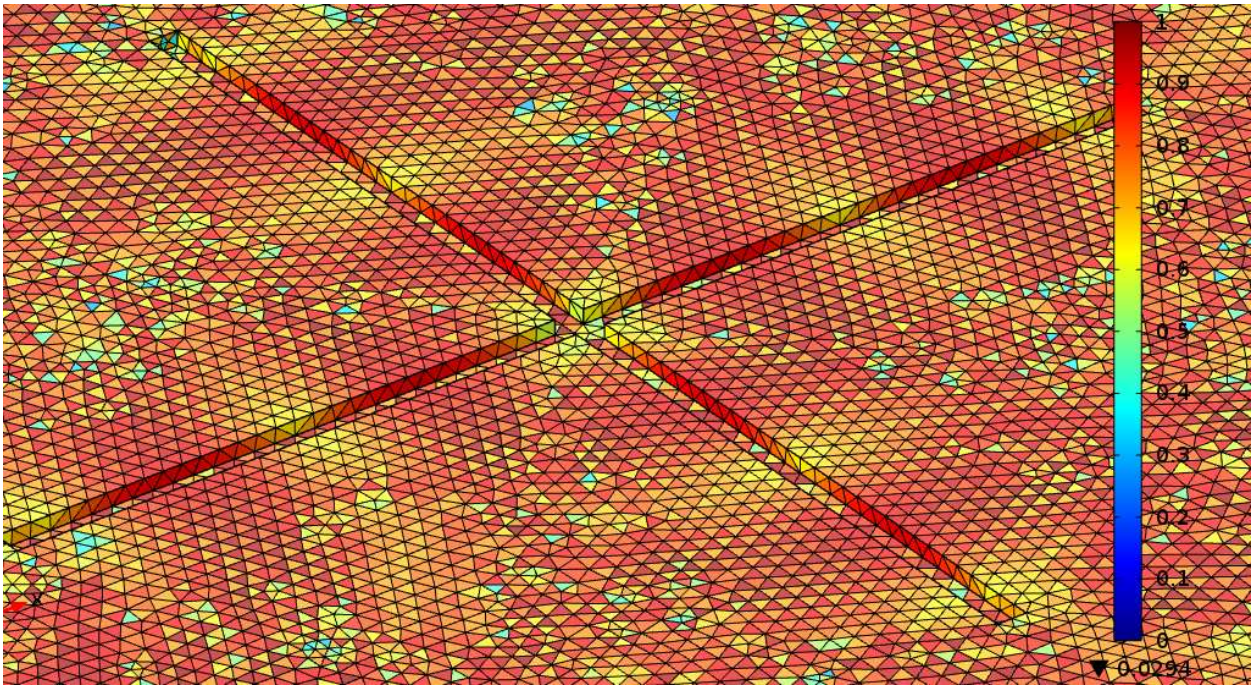


Figure 186: 3D Clamped Scored Plate Model – Finer Mesh Quality (Score View_2)

The displacement field and point evaluation are determined to be

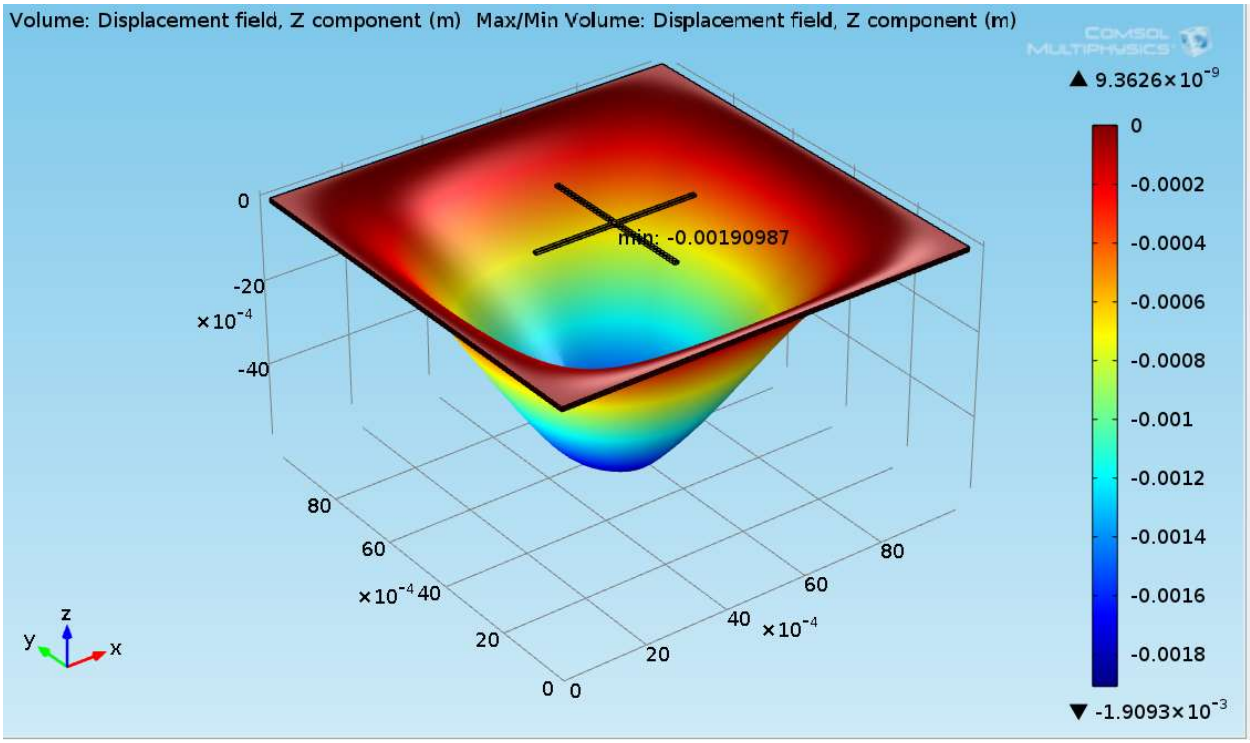


Figure 187: 3D Clamped Scored Plate Model – Finer Mesh Displacement Field (Z) and Point Evaluation

Looking at the stress and strain, we obtain the below plots that show the max stress and strain to occur at the score

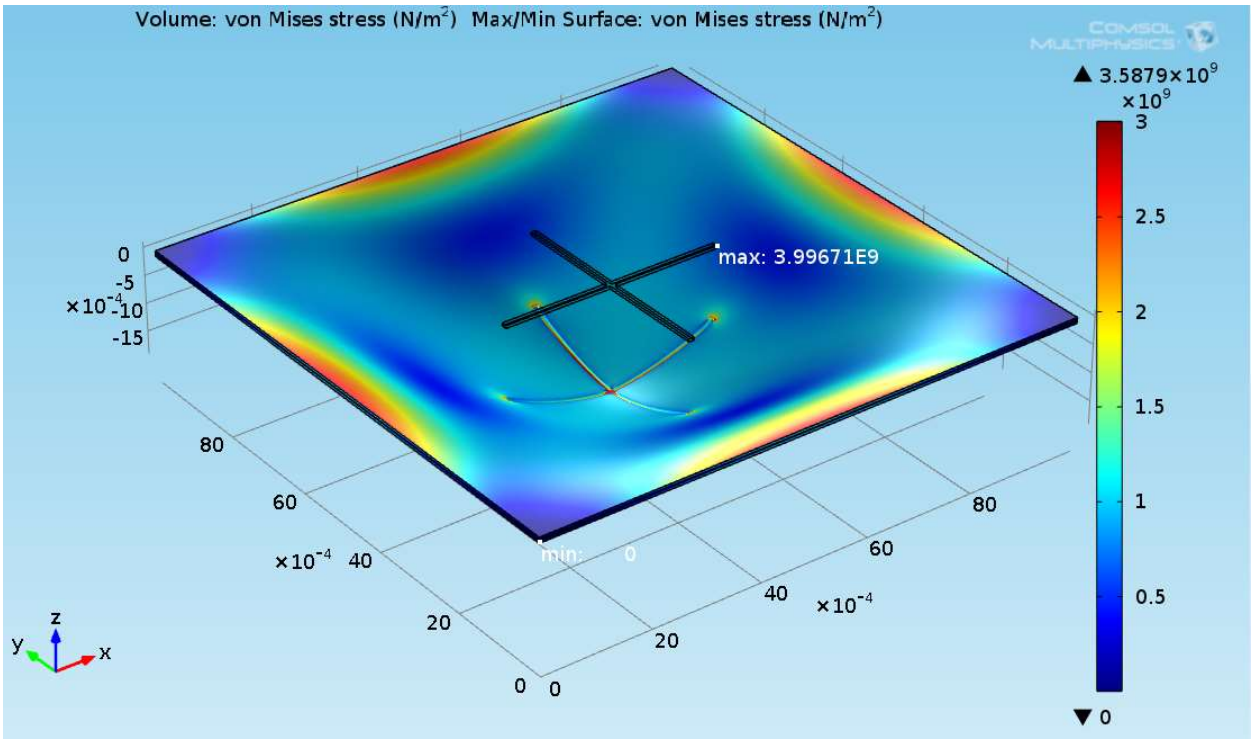


Figure 188: 3D Clamped Scored Plate Model – Finer Mesh von Mises Stress Plot (Isometric View)

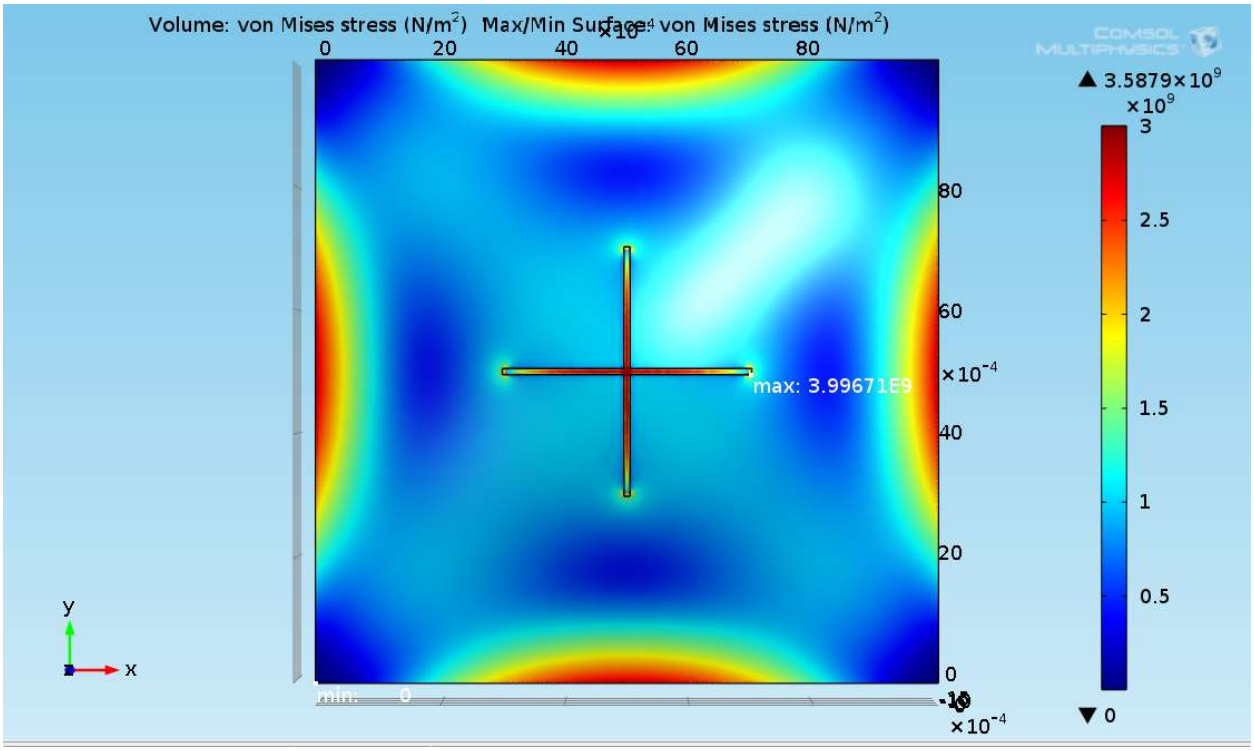


Figure 189: 3D Clamped Scored Plate Model – Finer Mesh von Mises Stress Plot (Top Score View)

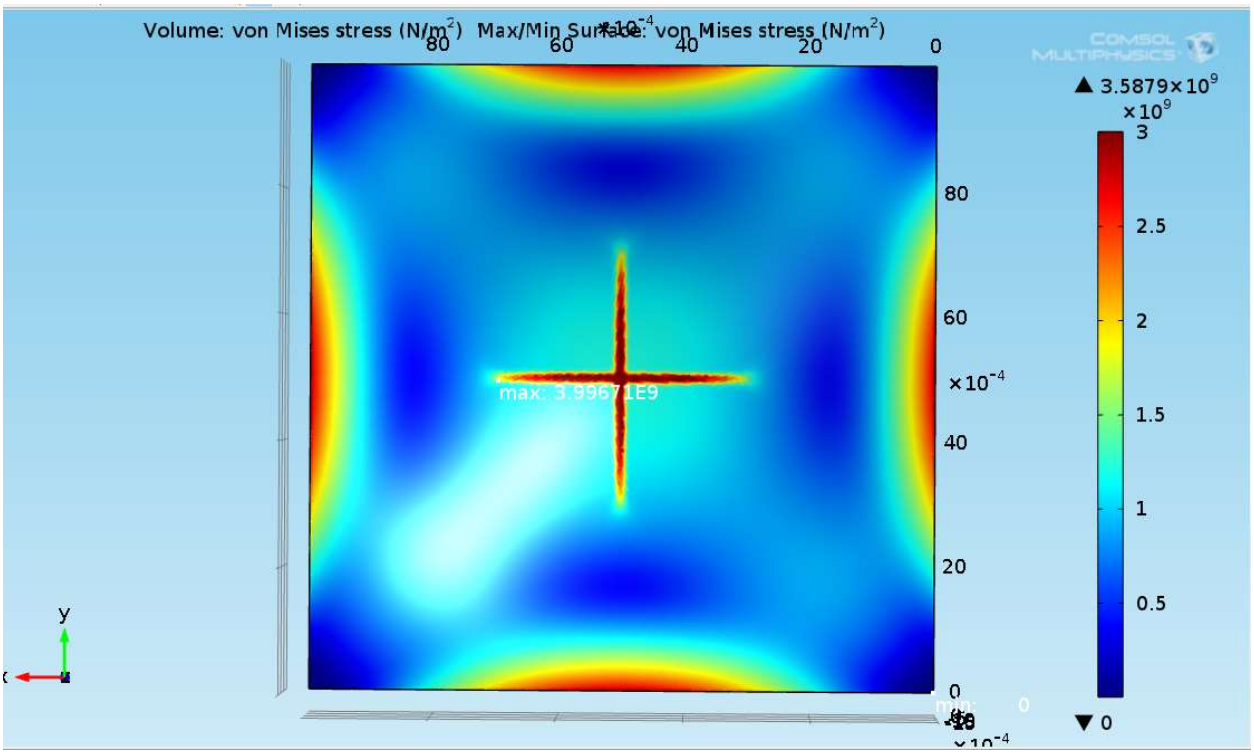


Figure 190: 3D Clamped Scored Plate Model – Finer Mesh von Mises Stress Plot (Bottom Score View)

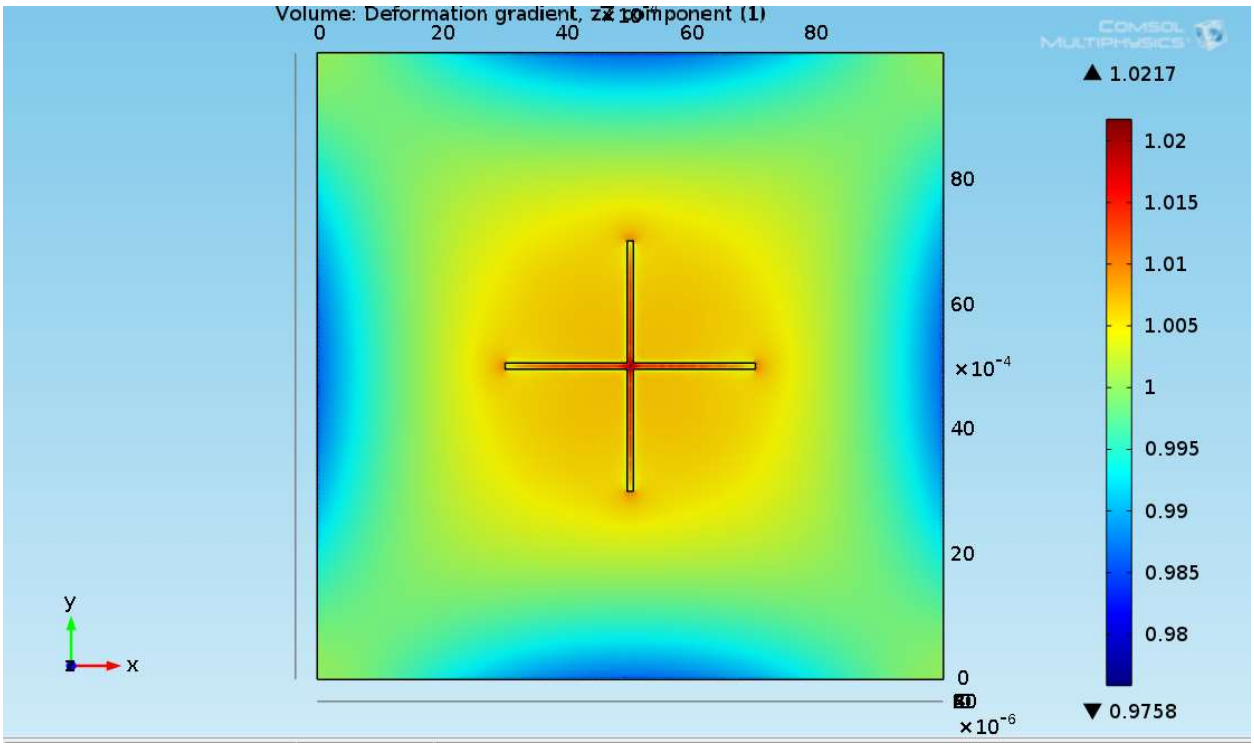


Figure 191: 3D Clamped Scored Plate Model – Finer Mesh Strain/Deformation (zZ) Plot

Discussion

Gaining a basic understanding of beam and plate deflection utilizing a simplified model lays the basic groundwork for the analysis of rupture discs. Simply supported and cantilever beams drive the analysis and understanding from a structures point of view while yielding the appropriate equations to analyze. They also allow us to grasp a full understanding of how differing boundary conditions effect the exact solution and in turn, finite element analysis. The derivation of the preliminary deflection equations using the known boundary conditions and elementary strength of materials concepts is important in being able to obtain the differential equation and then the exact solution of the problems. The exact solution is critical to this problem since it is what the numerical solutions are compared to. The derivation of the exact solution is found to be correct by checking the calculations with Maple software.

Knowing our foundation equations for the deflection of the plate and beam, is also crucial in the development of the differential formulation. With the differential formulation, the variational formulations are derived. Obtaining the variational formulation is the next step to successfully solving a finite element-based problem because it is used in the numerical solution; in our case, the simple Galerkin approximation (using known, admissible functions for the test functions) and the more complex Galerkin approximation (using piecewise global basis functions as the test functions). The variational formulation can be difficult to attain without careful steps being taken such as performing mathematic operations in the form of integration by parts properly, especially when dealing with Dirilecht and Neumann boundary conditions.

The initial Galerkin approximation uses the known, admissible functions which produces an almost perfect match to the solution. As we increase the parameter from a one-parameter to a five-parameter numerical solution for the simply supported beam, we see the error reduce from 0.0000229 to 0.000000415. The error in these instances are so low, because the test functions that are used in the variational formulation are of similar shape to the exact solution. When using the Galerkin approximation by building global basis functions for the test functions, we can see that the error is much greater at small element numbers when compared to the known functions as the test functions. In most cases, we will not know a function that closely represents the shape of the exact solution, so construction of piecewise linear global basis functions is essential and can yield a very close result to the exact solution. We can see the use of the Lagrange polynomials to construct the local basis functions at the nodes of the elements that have been defined. These local basis functions can be turned into the global basis functions to represent the full model by defining piecewise functions across the whole domain. This step is necessary because the test functions have the condition of being piecewise differentiable and these functions are just that. The global basis functions also allow the correct handling of the interior nodes vs the end noes in the Galerkin approximation.

The Galerkin approximation for the simply supported beam started with the two-element model which yielded a global error norm of 0.00545, while the four-element model shows the value to be 0.0027, and finally the eight-element model gives 0.0013. We can see that the error changes at

the same rate as the elements are increased. When doubling the elements from two to four, the error is cut in half. The same phenomenon occurs when doubling the elements from four to eight. This shows that as we increase the number of elements, the numerical solution approaches the exact solution.

This is also true when evaluating a beam with a free end, such as the cantilever beam. The Neumann boundary condition in this case adds a change to the manipulation of the variational form for the last equation of the finite element. If the free end was on the other side of the model, then this equation would occur at this location. This example shows the handling of the differing boundary conditions as well as the construction of the local and global basis functions for a three-element model, which contains two interior nodes rather than the two-element model with the three nodes. We can feel confident that we have a grasp on the concepts of deriving the equations for this code to run successfully no matter what the boundary conditions present.

Though the solutions matched very well for both beam cases with increasing elements, it is always a good idea to explore other verification methods. In our case, we can use a finite element software package called COMSOL. Using this method, we were able to input the geometry, the differential equation, boundary conditions, and define the mesh. After defining the model at hand, we were able to obtain the exact same deflection at the center of the simply supported beam of -0.00781 as seen in Figure 41 as we were able to use the Maple coding. We can also see that we can obtain the same plots for the Galerkin approximation by making a linear graph of the solution based on a defined mesh of the same size as our elements in Figures 43 and 44.

The time saved by utilizing the COMSOL software was unmatched when compared to the Maple coding. What we are deriving and coding in the Maple program is what is happening behind the scenes in a software package like COMSOL. With our solutions and accuracies coinciding between the Maple coding and the COMSOL package, we can see that we have derived and coded correctly, as well as set up our model and conditions in COMSOL correctly. As we move forward in understanding the finite element analysis of a plate in more dimensions, we know that we can rely on COMSOL to obtain the same results as a Maple code, but with much less time and possible troubleshooting of errors.

We now understand that from a 1D standpoint, COMSOL is a useful tool that can produce the same results as our Galerkin approximation and our exact solution results, but to utilize this software for other dimensions, we must study the exact solutions and verify using COMSOL in higher dimensions. We start with the cantilever beam with new material properties. Completing a Maple program to study the 1D and 2D exact solution validates that we can move to a 2D solution in COMSOL and compare. Figure 48 and 50 show the exact solution found using Euler Bernoulli beam theory and Timoshenko Beam theory, respectively. We can see that these both agree in a max displacement of -.04914 m in the y direction. Utilizing COMSOL and a mesh convergence study, we are able to obtain the same answer with a linear and quadratic discretization with less than 0.7% error. It can be seen that the quadratic finite basis functions converge more quickly and

the results are computed with less time and more degrees of freedom when comparing the same mesh refinement. We also acknowledge that the mesh refinement improves the quality of the mesh by looking at the mesh quality plots, showing that an increase in elements and degrees of freedom yield a better-quality mesh. Understanding the mesh extension study and the verification of the 2D COMSOL to the 2D and 1D Maple solutions, it is valid to use COMSOL when moving to a 3D solution. In the 2D COMSOL study, we evaluated the solution and mesh quality with a mapped quadrilateral mesh, so for the 3D, we look to a physics controlled mesh to see how this improves or decreases the accuracy of our answer. A mesh extension study is, again, performed. This time we also included a cubic discretization study to see how this improves solution quality and convergence. As expected, as the finite basis function order increases, so does the convergence and solve time. This can be seen in the mesh extension tables figures 81 – 83. In these tables we can also see that form the linear discretization to the quadratic, the accuracy improves tremendously. The extremely fine mesh of the linear is above 3% in error while the finer mesh for the quadratic sees an error of 0.06%. This even performs better than the cubic discretization, though this still does very well with error or 1%. This may be due to some small fluctuations in the convergence plot that have not stabilized fully, but this error is still very good for a 3D model that may capture more that the 2D and 1D Maple program that do not consider all 3 dimensions since we assume plane stress which does not depend on the z component. The mesh also improves as seen in figure 84 with a mesh quality that contains higher quality elements than that of the coarser mesh. We can confirm that using COMSOL with physics-controlled mesh and either a quadratic or cubic discretization is a suitable study for this more complex geometry with higher dimensionality.

Since we will be analyzing our plate as simply supported, we can also perform the same study of our simply supported beam to verify our results from the 1D Galerkin and exact solution to the 2D exact solution in COMSOL using 2D and 3D studies. This study also confirms our study of the cantilever beam in that we are essentially double checking our results. The 1D and 2D exact solutions of the simply supported beam (1D using Euler Bernoulli and 2D using Timoshenko) can be viewed in Figures 91 and 94. We do see a difference in answers due to the assumptions of Euler Bernoulli and Timoshenko with shear being considered in Timoshenko, which make the beam "less stiff", thus the answer will have a larger deflection. This is seen between the two values, the 1D solution yields -0.01919 m deflection while the 2D solution yields -.01962m deflection. When conducting the 2D COMSOL study of the beam in the same way as the cantilever beam, we see errors in both the linear and quadratic discretization of better than 0.5%, but in the finer mesh we can see up to 1.55%. This is not a particularly poor result with the more complex inertial terms in play. There are many assumptions used in the exact solution calculations that may not be accounted for in the COMSOL result, but we can still say that the FEA software is reliable. With this reliability, we move to the 3D evaluation, again using physics-controlled meshing that proved well with the cantilever beam. Since know from the cantilever beam that the linear discretization takes longer, we can look to study the quadratic and cubic discretization which both show great results with the increasing quality elements. The results returned better than 1% in each case. With this

confirmation of results, we can validate the results from the cantilever beam study as well as the results using COMSOL in general with a mesh convergence study.

The rupture plates that are of interest are just that, plates. In this way, we must evaluate the solution to a plate using Timoshenko plate and shell theory where he introduces the Navier Solution. The same Inconel 625 is used, but the plate is appropriately sized for the applications that it will see. When conducting the evaluation, we find the solution to the displacement in the z direction to be a max at the center and of magnitude -.00553m. Conducting a study in COMSOL will verify the results, we can view the mesh extension study in figures 145 and 146 for the linear and quadratic discretization. For this study of the plate, we employ the mapped mesh over the face and sweep through the thickness to see when we have the largest effect of mesh refinement. Figure 151 shows the mesh convergence plot where we obtain an answer within 1% error of the Navier Solution. This verifies COMSOL is suitable to conduct this plate study in matching the Maple program solution. One point to make is that in the linear discretization convergence curve, there is a jump in the curve. This is because the curve is plotted against degrees of freedom and to properly order the DOF in an increasing manner, a coarser mesh on the face, but five times finer through the thickness is used. This tells us that continuing to increase the mesh though the thickness does not gain much accuracy. The mesh is better off refining over the width and length of the plate.

In the real-world situation of these rupture discs, the disc is not simply supported, rather clamped on all four sides. With the verification of the cantilever beam, the simply supported beam, and then the simply supported plate, we proceed with COMSOL to evaluate the true problem at hand which is evaluating the expected benefit of adding a score to the plate to better control the rupture pattern.

From all of the other studies, we see that using a quadratic finite element basis function is suitable for finding an accurate solution to the problem. We also have learned that using a mapped mesh with a sweep through the thickness is an easy and accurate way to mesh the model, so this is what we will use in the mesh extension study. At the end of the study, we will also include a tetrahedral mesh that we refine to see the effects this has on the study with its' degrees of freedom and computation time. Figure 155 shows the mesh extension study with a converged solution of -.001722m deflection in the center of the plate in the z component. This value agrees with the type of deflection we would expect in the normal operation of this disc. We see that this is converged between 600,000 degrees of freedom to 1,000,000 for the mapped mesh with the computation time ranging 119 to 299 seconds. This solution is also obtained using the tetrahedral refined mesh through the thickness and face with a computation time of 157 seconds with over 1,300,000 degrees of freedom. The added degrees of freedom is a benefit, as well as the computation time for this to be completed compared to the mapped mesh.

Looking to the stress and strain of the disc, we have a von Mises stress of about 10E8 to 15E8 N/m^2 at the center of the plate which remains in the elastic region. The max stresses of 29E8 N/m^2 near the boundaries is also within the elastic region which has a yield point of 41.4E8 – 65.5E8 N/m^2. The strain in the plate is at a max of around 1.01 at the center of the plate, which

is expected. The high stresses and strain at the center of the plate show promise as to where the plate may fail, but with high stresses on the boundaries, this could be locations of failure as well.

Now that we have a baseline for the behavior of a clamped plate, we shall observe the scored plate to evaluate the differences and see if we can create regions of high stress and strain in our controlled form. Logically this should be the case since in the score, the thickness is cut in half and should therefore fail at this location first, but we still need to verify and observe this using the FEA software. We also don't want the score to be too deep, because then we could have premature dislocations and failures in the plate.

Again, since we verified that the quadratic discretization is reliable, we will use this for the meshing. We also will use the physics-controlled mesh since it has also proven to be reliable, but within our mesh extension study, we will refine through the mesh and also refine the elements around the score. We can view the plots of the mesh quality in Figures 177-179 for the coarsest mesh and Figures 183 – 186 for the finest mesh. The study converges to a solution of a deflection in the z component of -.0019m. This is a larger deflection that that of the no score clamped plate, but this is correct since at the score, we have a thinner plate region and therefore will allow more deflection.

Figures 188 through 190 show the von Mises stress in the plate. We see that the max stress is in the score (39E8 N/m^2) rather than at the boundaries. This stress is still occurring in the elastic region for annealed Inconel 625 plate. The stress at the boundaries is still high but is concentrated in the score. Figure 191 also shows the greatest strain in the score, which shows that combined with the stress, this is most likely where the plate with rupture.

This amount of control through the disc will ensure a cleaner rupture mode and keep the customers system less susceptible to foreign object debris. Though we remain in the elastic region with the operation pressure applied, we can predict that as the load is increased, so will the stress and strain until failure. With more complex knowledge of finite element analysis, we would be able to verify and validate the behavior in the plastic region to failure. This can be left to future work, but at this point we have understood and proven how to better control the rupture of these plates in the field.

The use of finite element analysis has proven to be a great way to analyze rupture discs, or any problem for that manner. By progressing through this course, I was able to learn and understand what is going on behind the scenes of these programs and this is able to verify and validate all of the work that is put into the FEA software. Moving forward, for simple problems, the Galerkin method has proven to be a very reliable and effective way to model problems with the use of local and global basis functions. When moving to higher order problems, FEA software has been proven to be a very accurate method to finding a solution, especially when used with a mesh convergence study and comparing different finite basis functions accuracy and computation time. Problems of higher complexity can be quickly modeled and solved using these methods, where the mathematics can take a long while. Time is of the essence when running many studies and obtaining accurate results, so using FEA software while also understanding the meshing to use, meshing refinement

to utilize, and discretization methods will allow engineers like myself to produce highly complex models with accurate results.

References:

[1] RPI MANE 4240. Professor Ernesto Gutierrez Video Meeting Conferences. Fall 2020.

[2] RPI MANE 4240 Class Notes "IFE – Differential and Variational Formulations of Models for Milestones 1 and 2". Fall 2020

[3] RPI MANE 4240 Class Notes "The Classical Method of Galerkin". Fall 2020.

[4] RPI MANE 4240 Class Notes "Differential (D) and Variational Formulation (V) for Boundary Value Problems involving Second Order Ordinary Differential Equations". Fall 2020.

[5] RPI MANE 4240 Class Notes "Module 1 Summary Variational Formulations". Fall 2020.

[6] RPI MANE 4240 Class Notes "Integration by Parts Functions of a Single Independent Variable". Fall 2020.

[7] RPI MANE 4240 Class Notes "Ritz and Galerkin Methods". Fall 2020.

[8] RPI MANE 6720. Professor Frank Cunha Class Notes, PowerPoints, Video Lectures, and Video Meeting Conferences. Summer 2020.

[9] Shigley, Joseph Edward., et al. *Mechanical Engineering Design*. McGraw-Hill, 2011.

[10] Megson, Thomas Henry Gordon. *Introduction to Aircraft Structural Analysis*. Butterworth-Heinemann., 2018.

[11] Corporation, Specialty Metals. "Inconel 625 Alloy." Https://Www.specialmetals.com/Assets/Smc/Documents/Alloys/Inconel/Inconel-Alloy-625.Pdf, 13 Aug. 2013.

[12] "Chapter 3." *Theory of Elasticity*, by Stephen Timoshenko, McGraw-Hill, 1970, pp. 29–39.

[13] "Chapter 3." *Theory of Elasticity*, by Stephen Timoshenko, McGraw-Hill, 1970, pp. 29–43.

[14] "Chapter 5 - Simply Supported Rectangular Plates." *Theory of Plates and Shells*, by Stephen Timoshenko and S. Woinowsky-Krieger, McGraw-Hill, 1959, pp. 105–111.

[15] "Chapter 5 - Rectangular Plates." *Stresses in Plates and Shells*, by Ansel Ugural, Springer Berlin, 2007, pp. 141–158.

[16] Gothall, Hanna. "How to Inspect Your Mesh in COMSOL Multiphysics®." *COMSOL Multiphysics*, 12 May 2017, www.comsol.com/blogs/how-to-inspect-your-mesh-in-comsol-multiphysics/.