University of Colorado - Boulder

ASEN 3802: Aerospace Sciences Laboratory II

February 15th, 2024

Lab 1: Structural Mechanics

Author:
Jordan Richardson

Author:
Zachary Selleck

Author:
Brady Sivey

Author:
Jared Steffen

Professor:
Erik Knudsen

Ann and H.J. Smead
Aerospace Engineering Sciences
UNIVERSITY OF COLORADO **BOULDER**

I. Introduction

The primary objective is to demonstrate how the integration of real-world measurements with computer simulations can provide insights into the behavior of structures under varying conditions. Specifically, the focus is on a 3D 16-bay truss subjected to static loading, akin to scenarios found in beam-like support setups.

The data collected from experiments will be compared with what is expected from theoretical models. The software ANSYS was used to run simulations based on the Finite Element Method. This software allows for the creation of detailed digital models of the truss and examination deflection, reaction forces, and internal forces in response to various loading scenarios.

The analytical model was hand derived and the truss was assumed to be an equivalent beam for this model. Utilizing the derived analytical model, our objective is to ascertain the loading conditions for two additional scenarios primarily relying on reaction forces. The initial scenario involves a load applied at a single node positioned away from the center. Subsequently, the second scenario presents the challenge of identifying two equal loads applied at unknown nodes within the truss structure.

By conducting these analyses, the aim is to not only assess the accuracy of the models but also to gauge the sensitivity of predictions to factors such as variations in material properties and experimental data. This approach allows for a thorough examination of the impact of potential sources of error, including imperfect joints, free-play in load cells, friction at joints, manufacturing imperfections, and assumptions regarding stress distribution.

II. Methodology

A. MoI Calculation

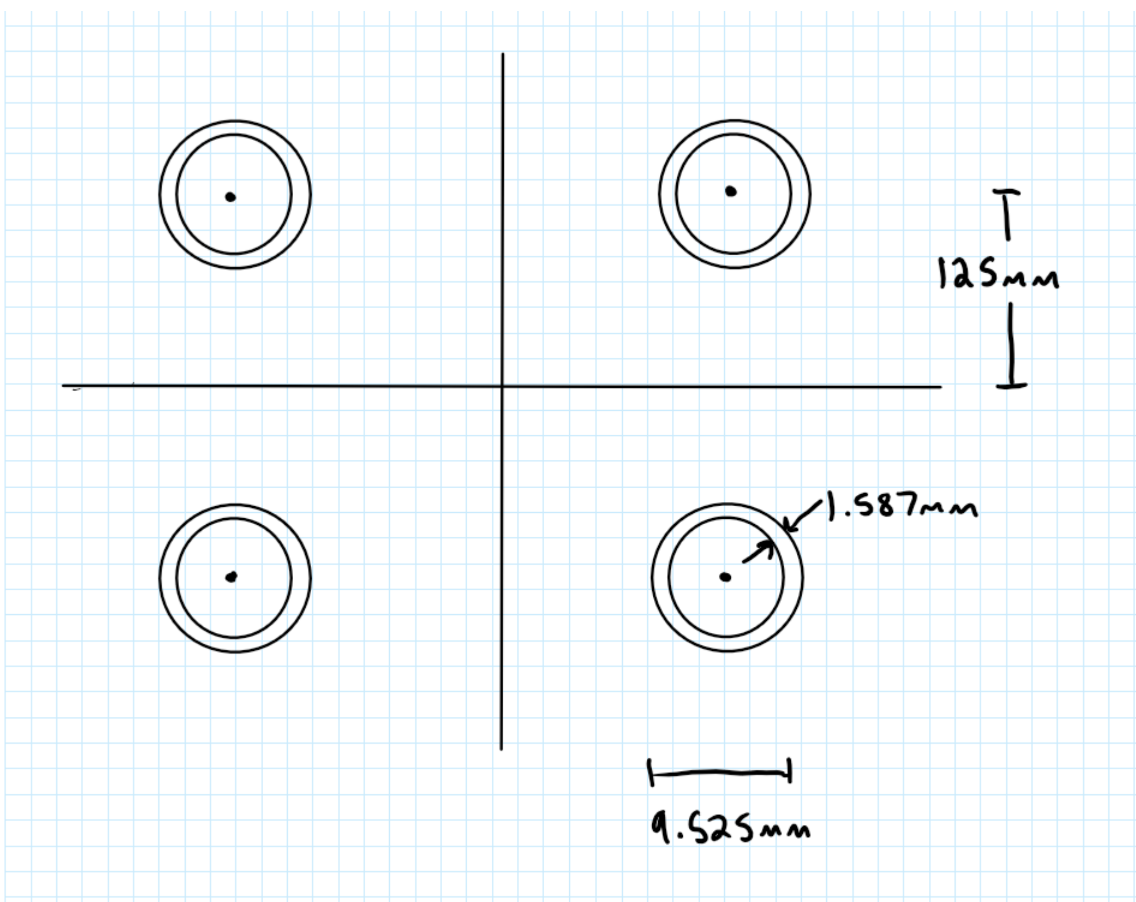


Fig. 1 Beam Equivalent Model Cross Section

For the analytical model, the truss was modeled as an equivalent beam, only considering the four struts that go the entire length of the truss and not the diagonal two force members. Figure 1 has the dimensions for the cross-sectional area of this equivalent beam model. Parallel-axis theorem was then used on the cross-section to find the area moment of inertia. The final result after using the parallel axis theorem was 2.476×10^{-6} m^4. Below are the equations used and tabulated results for determining the area moment of inertia.

1. Equations

$A_{ring} = \pi(r_{outer}^2 - r_{inner}^2)$
$I_z = \frac{\pi}{2}(r_{outer}^4 - r_{inner}^4)$
$I_z = \Sigma\overline{I_z} + Ad^2$

2. MoI Table

Table 1 Area Moment of Inertia Tabulated Data

Segment	Area (A) [m^2]	Distance (d) [m]	Ad^2 [m^4]	I_z [m^4]
1	3.957×10^{-5}	0.125	6.183×10^{-7}	6.483×10^{-10}
2	3.957×10^{-5}	0.125	6.183×10^{-7}	6.483×10^{-10}
3	3.957×10^{-5}	0.125	6.183×10^{-7}	6.483×10^{-10}
4	3.957×10^{-5}	0.125	6.183×10^{-7}	6.483×10^{-10}
Σ	–	–	2.473×10^{-6}	2.593×10^{-9}

B. Beam Bending Derivation

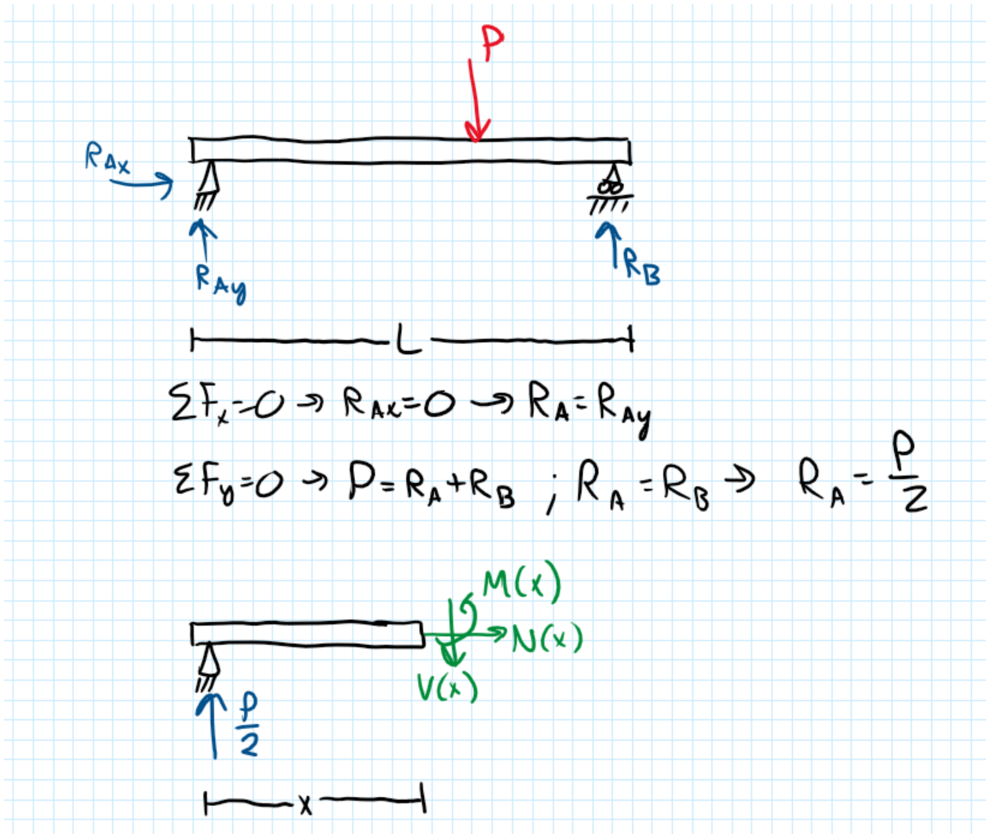


Fig. 2 Beam Bending FBD

The FBD in Figure 2 was used to perform a statics calculation to determine $M(x)$, which in turn can lead to $\upsilon(x)$, the beam deflection up to the midpoint of the truss.

$$\Sigma M_x = M(x) - \frac{P}{2}x = 0 \therefore M(x) = \frac{P}{2}x$$

$$\Theta(x) = \frac{1}{EI}\int M(x)dx = \frac{1}{EI}\left[\frac{P}{4}x^2 + C_1\right]$$

$$\upsilon(x) = \frac{1}{EI}\int \Theta(x)dx = \frac{1}{EI}\left[\frac{P}{12}x^3 + C_1x + C_2\right]$$

At the midpoint of the truss $\Theta = 0$ and at the beginning of the truss $\upsilon = 0$. Therefore:

$$\Theta(2) = \frac{1}{EI}[P + C_1] = 0 \therefore C_1 = -P$$

$$\upsilon(0) = \frac{1}{EI}C_2 = 0 \therefore C_2 = 0$$

$$\therefore \upsilon(x) = \frac{1}{EI}\left[\frac{P}{12}x^3 - Px\right]$$

III. Results

A. Question 1: Experimental results

1. Experimental Data Analysis

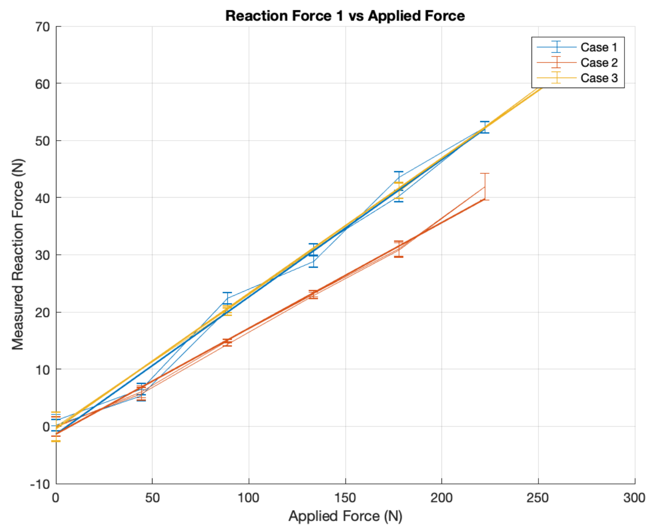


Fig. 3 Experimental Reaction Force 1 for All 3 Cases with Error Bars

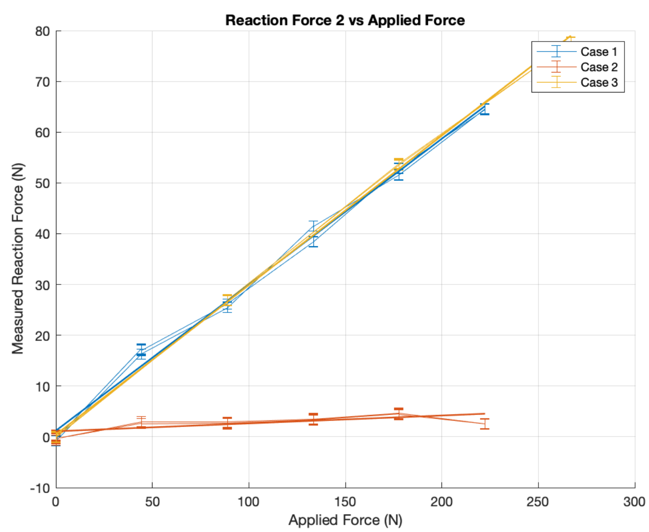


Fig. 4 Experimental Reaction Force 2 for All 3 Cases with Error Bars

Figures 3 and 4 show the reaction forces at one end of the truss as a function of increasing applied load. For cases 1 and 3, the loading curves have similar values and slopes for each sensor. This makes sense because the loads are determined to be at the midpoint of the truss for both of these cases. This is given to us for the first case, and the derivation for the third case can be seen in Appendix A. Regarding the second case, the most probable reason for why the reaction forces are different is due to the fact that the load is off center. This is explored more in depth in question 3.

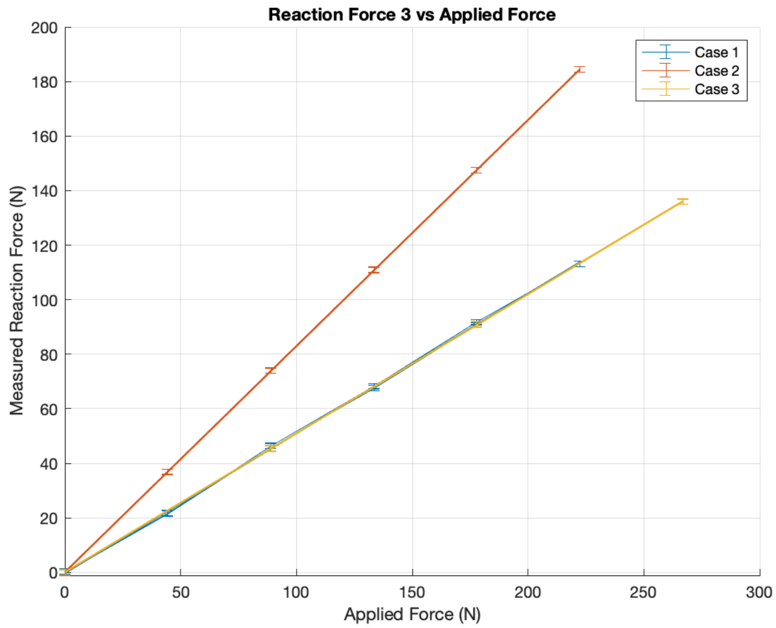


Fig. 5 Experimental Reaction Force 3 for All 3 Cases with Error Bars

Figure 5 shows the reaction force at the other end of the truss as a function of increasing applied load. Case 1 and 3 show a very similar loading trend while case 2 exhibits a much higher load as compared to the two. This means that for case 2, it can be inferred that the load was placed much closer to load sensor 3 as compared to load sensor 1 or 2. Further, it can be seen that based on these results and the results from figures 3 and 4, the loading distribution is very similar for cases 1 and 3 and can be predicted to be positioned near the center of the truss.

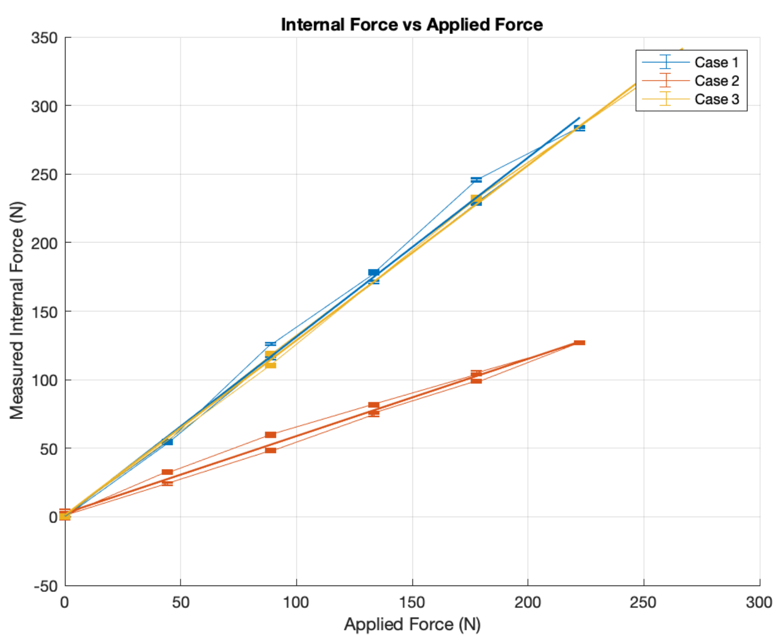


Fig. 6 Experimental Internal Forces for All 3 Cases with Error Bars

Figure 6 shows the internal force within the truss at the lengthwise midpoint as a function of increasing applied load. For cases 1 and 3, the internal force is very similar throughout. However, case 2 shows much lower internal forces. This makes sense because it's been assumed that the resultant force for case 2 is not centered at the middle while case 1 and 3 show a likelihood to have a resultant force positioned in the center. Because the internal force is measured at the midpoint, it makes sense that it would be less for case 2 as compared to cases 1 and 3.

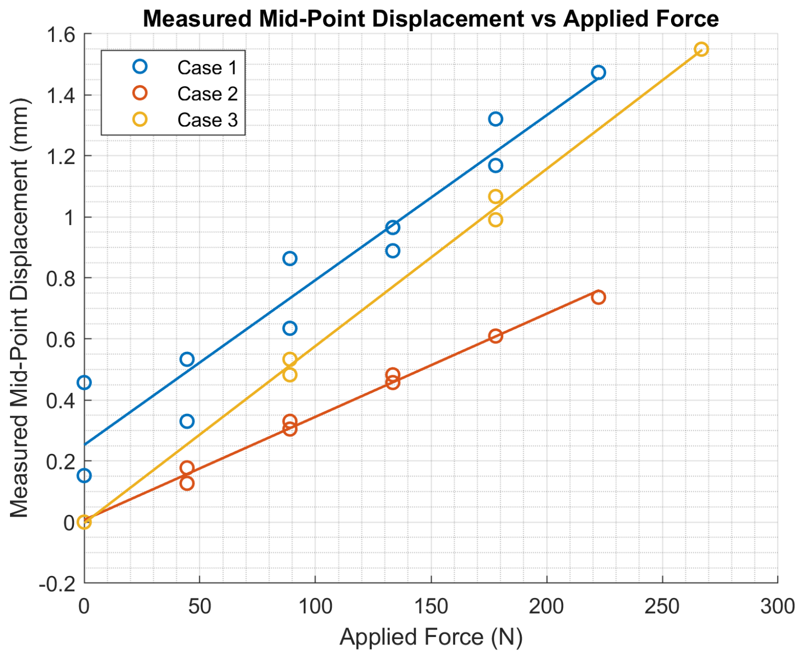


Fig. 7 Experimental Midpoint Displacement For All 3 Cases

Figure 7 shows the mid-point displacement as a function of increasing applied force. Right away it should be noted that case 1 shows an initial displacement at zero loading. This could be due to the sensor not being zeroed before measurements were taken. Conversely, case 2 and 3 both start at zero, however case 3 has much more overall displacement than case 2 and similar overall displacement to case 1. This makes sense because, again it was inferred that the resultant loading for cases 1 and 3 is centered at the midpoint which would give the greatest displacement where the sensor is located.

Table 2 Experimental Data

Measurement	Case 1	Case 2	Case 3
Reaction Force 1 [N]	52.3289	41.9289	63.5651
Reaction Force 2 [N]	64.5837	4.6528	77.7282
Reaction Force 3 [N]	113.2428	184.4411	135.9465
Internal Midpoint Force [N]	284.6328	127.4980	337.1094
Midpoint Displacement [mm]	1.4732	0.7366	1.5494

2. *Linear Regression and Uncertainty Analysis*

Table 3 R Squared Values

Model	Case 1	Case 2	Case 3
Reaction Force 1	0.9857	0.9940	0.9995
Reaction Force 2	0.9935	0.5637	0.9993
Reaction Force 3	0.9997	1.0000	1.0000
Internal Midpoint Force	0.9959	0.9908	0.9989
Midpoint Displacement	0.9272	0.9952	0.9978

The lines of best fit were determined through the least squares method for all of the plots is Figures 3-7. Tables 3 shows the R Squared values for the linear regression model fitted to the experimental data for all of the sensors. By looking at the table, almost all of the values are very close to 1 which represents a perfectly fitting linear regression line to the data. This thereby verifies the linearity of the measurements versus external load magnitude.

Figures 3-6 show error bars based on the standard deviation at each point. For all of the cases, the linear regression models are well within these error bars. This further proves the linearity of the regression line and the data.

B. Question 2: Comparison with analytical and FEM results

1. *Analytical Results*

Through the use of statics, the reaction forces at each corner of the truss were calculated to be $55.6N$. Additionally, the internal forces, which were assumed to be constant through the bars, were calculated to be equal to the load, which was $222.4N$. Using the values of $E = 70GPa$, $P = 222.4N$, and $I = 2.476 \times 10^{-6} m^4$, $x = 2m$ as well as the derived equation for $\upsilon(x)$, the magnitude of the deflection at the midpoint comes out to be $0.001711m$ or $1.71mm$

2. *FEM Results*

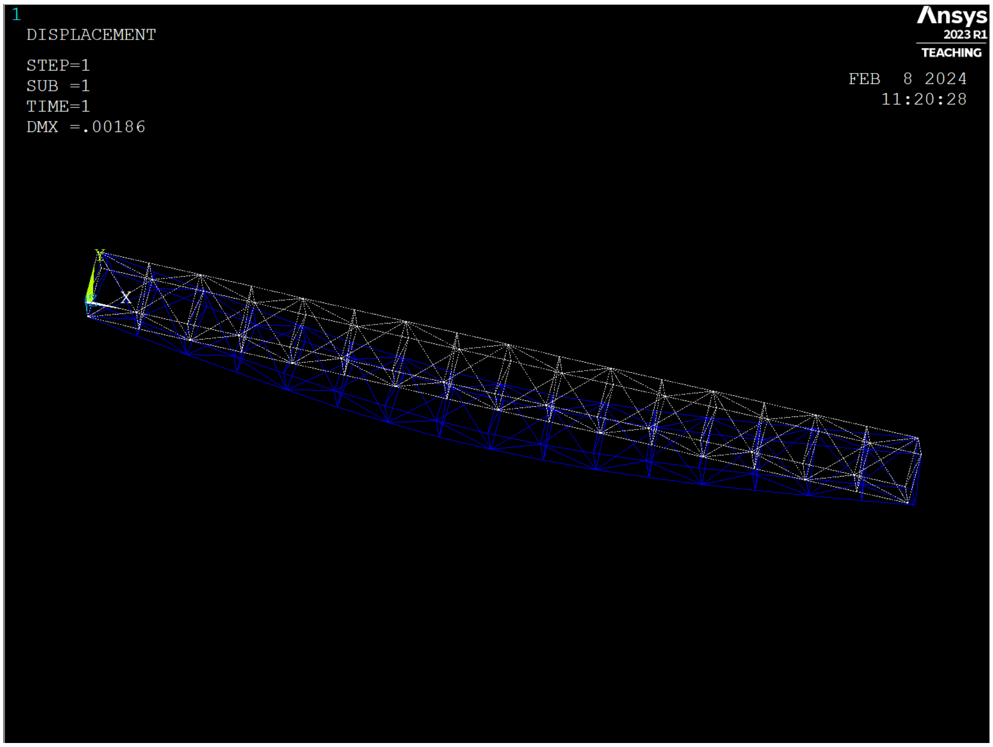


Fig. 8 Deformed vs Undeformed Truss Structure (Exaggerated)

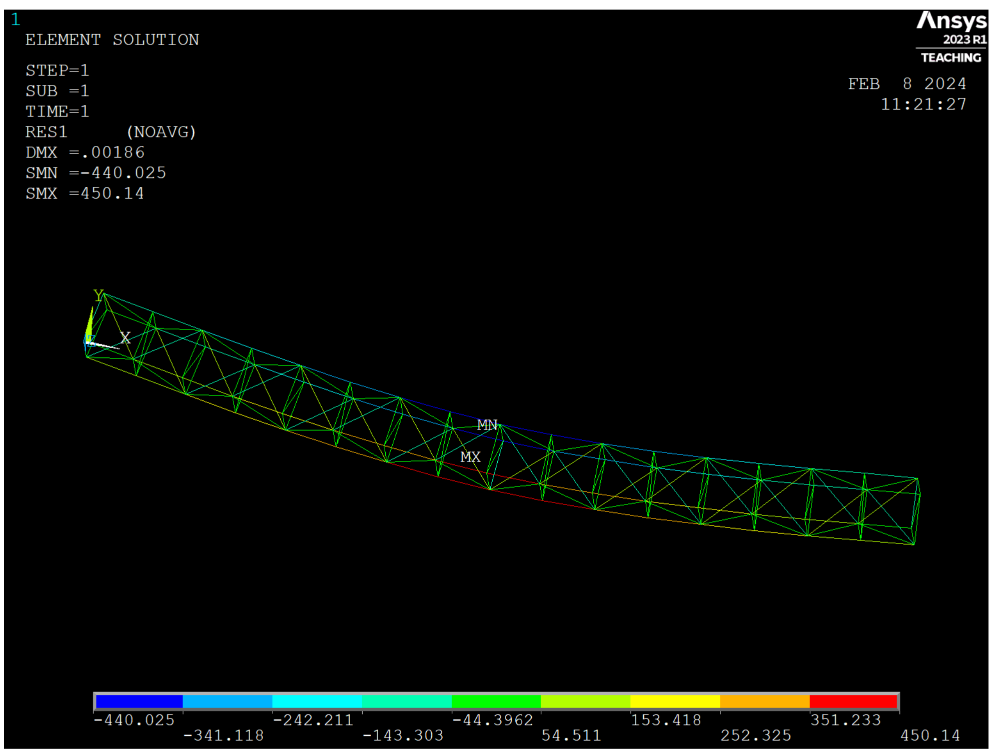


Fig. 9 Internal Forces Contour Plot

Figures 8 and 9 show the deformed vs undeformed truss and the contour plot for the internal forces respectively. With a force of $-111.2N$ acting on each of the middle nodes, the truss had a midpoint deflection of $1.85mm$. The maximum internal force was found to be $450.14N$ with reaction forces right around $55.6N$ for each corner joint connected to a supporting joint.

3. Comparison

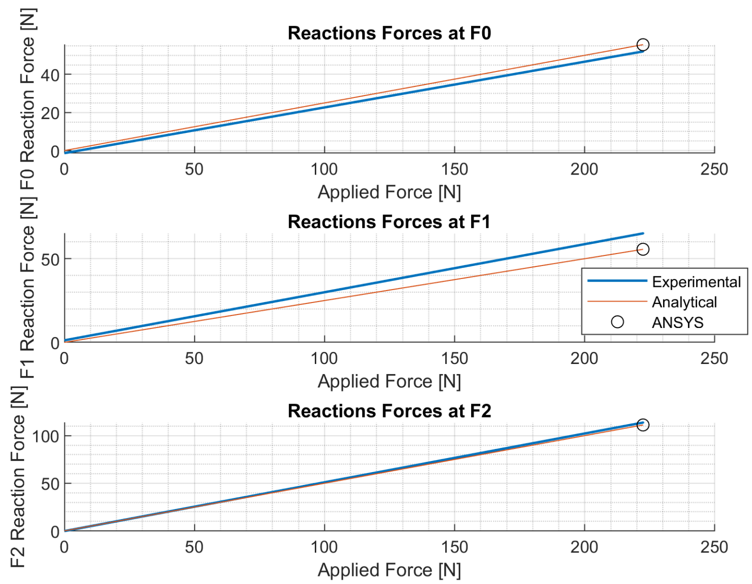


Fig. 10 Reaction Force Comparison For All 3 Models

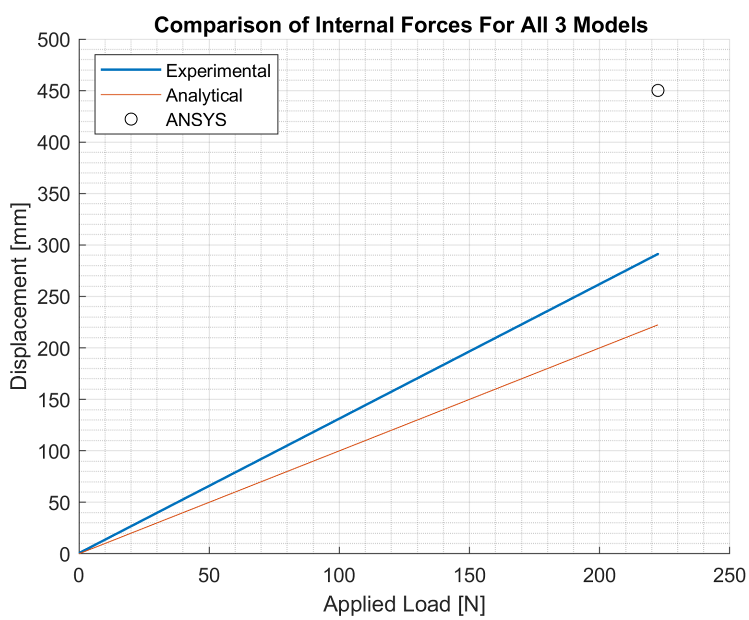


Fig. 11 Internal Force Comparison For All 3 Models

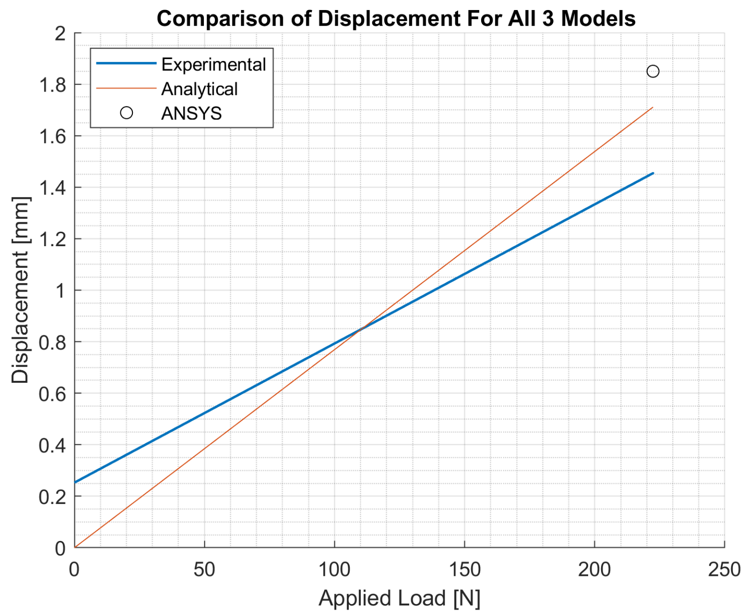


Fig. 12 Midpoint Displacement Comparison For All 3 Models

Figures 10-12 depict the graphical results of all 3 models. Note that the FEM model only tested the maximum loading scenario, which is why there is only one data point at the maximum loading of $222.4N$. The two tables below show the numerical results as well as the percent errors for each model compared to the experimental data. For the analytical model, the midpoint displacement sees a 17.61% error, while the internal forces see a 23.65% error. The reaction forces are much smaller, with 6.86%, 14.63%, and 2.19% for F0, F1, and F2. For the ANSYS analysis, quite a large difference in midpoint deflection and maximum internal force, with 23.65% and 54.54% differences. The reaction forces have much smaller percent error, similar to the analytical model, with 6.78%, 14.59%, and 2.21% for F0, F1, and F2.

The main reasons behind the differences between the experimental results and the analytical and FEM models is due to sources of error, which are talk about in section IV, and due to the simplifications and assumptions made to make analysis of the models easier. The main assumptions for the analytical model include the fact that the truss is made of only 4 struts that run the full length of the structure, as well as no friction at the supports. The main assumptions for the FEM model are that the struts actually are links rather than beams, and that every joint in the truss is perfect and allows no free-play.

Table 4 Comparison from Experimental Data

Model	Midpoint Displacement [mm]	Max Internal Force [N]	Reaction Forces [N]
Experimental	1.45	291.31	F0: 52.03 F1: 65.13 F2: 113.79
Analytical	1.71	222.40	F0: 55.6 F1: 55.6 F2: 111.20
FEM	1.85	450.14	F0: 55.56 F1: 55.63 F2: 111.19

Table 5 Percent Error from Experimental Data

Model	Midpoint Displacement Error	Max Internal Force Error	Reaction Forces Error
Analytical	17.61%	23.65%	F0: 6.86% F1: 14.63% F2: 2.19%
FEM	27.24%	54.54%	F0: 6.78% F1: 14.59% F2: 2.21%

C. Question 3: Using modeling to identify the other loading conditions

1. Loading Case 2

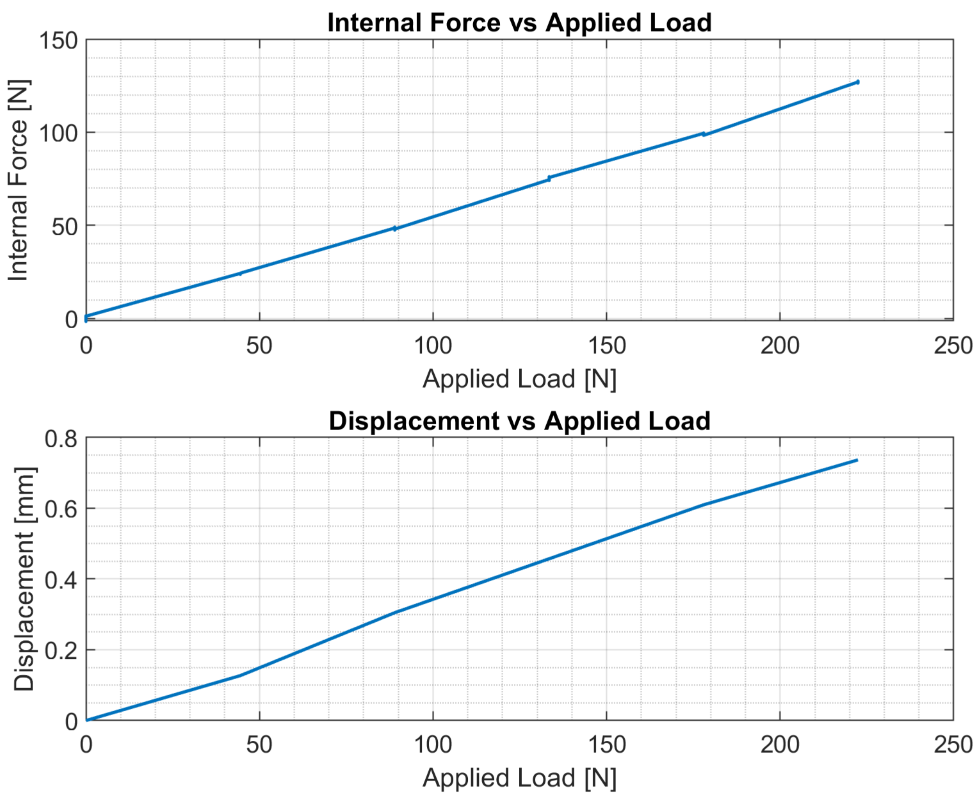


Fig. 13 Internal Force and Displacement For Loading Case 2

For the first unknown loading case, there was one load at a unknown location with the reaction forces, midpoint deflection, and internal forces given form the experimental data. It was possible to determine the loading location due to the fact that there was only one unknown, the loading location distance from the start of the truss. The derivation for determining this loading placement using statics can be seen in Appendix A, but the resulting equation for the location of the load was:

$$a = \frac{R_B L}{P}$$

The final resulting location was at $3.31m$. Knowing that the loads could only be placed at increments of $0.25m$, it is safe to assume that the actual loading position was at $3.25m$. Putting this value into the equivalent beam model, the resulting deflection at the midpoint if the load is at $3.25m$ is $0.24mm$, while the actual result was $0.7366mm$. This yields a 66.38% error.

The experimental midpoint internal force was $127.5N$. For the equivalent beam model, the internal force is assumed to be constant through the whole bar, with a magnitude of $222.4N$. This yields a 74.43% error. These results indicate that the equivalent beam model is not an accurate model if the displacement is off center like it is in the second case. The assumed reason for such a large percent error between the experimental and equivalent beam model is due to the fact that the bending stiffness provided by the struts and the bending moments transmitted by the joints are going to be different to the left and right of the load when the load is off center and the equivalent beam model doesn't account for either of these. The experimental data for midpoint displacement and internal force at the middle of the truss can be seen in Figure 13.

2. *Loading Case 3*

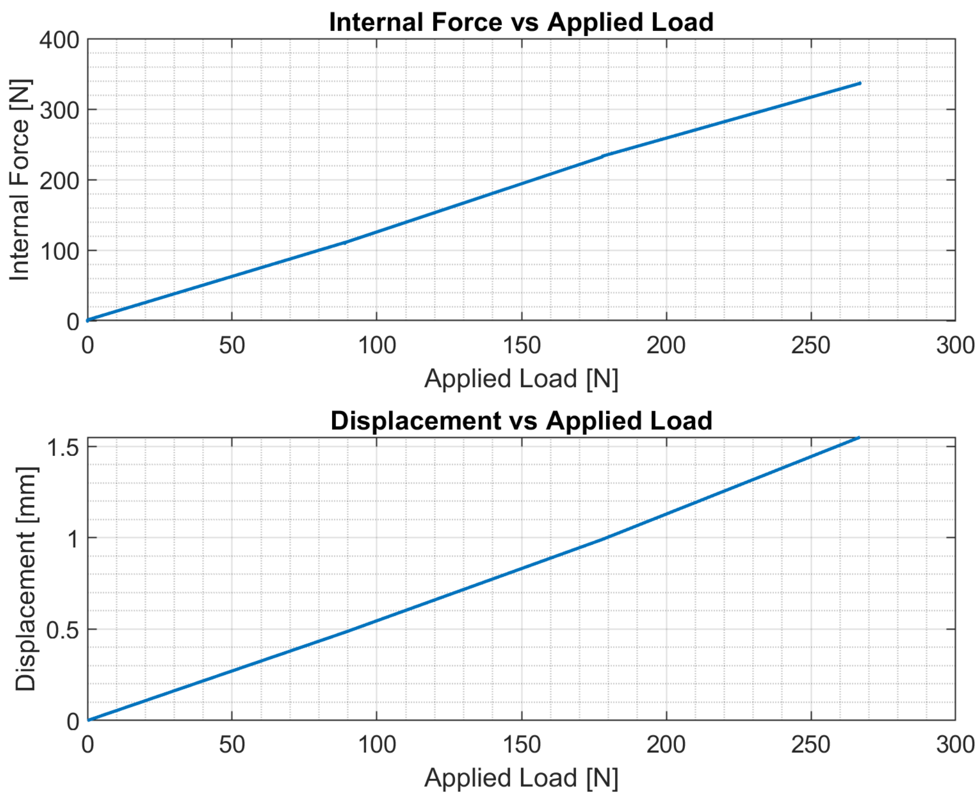


Fig. 14 Internal Force and Displacement For Loading Case 3

The second unknown loading case involved finding the location of two equal forces. This was done by treating the two forces as a single resultant force in the center of the two forces. With this, we are able to identify the midpoint between the two forces. From there, we derive the deflection each force individually. These derivations can be seen in Appendix A, titled "Case 3". Using our derived deflection equations for the left and right sides of the beam, we are able to superimpose them in order to find the deflection at the center of the truss. As opposed to the previous problem, we need to know the location and magnitude of the resultant force, that both the deflection and slope at the resultant force are equal for both the left and right beam deflection, and the constant parameters like Young's Modulus, area moment of inertia, and length of the truss. Also, the deflection at the beginning and end of the beam must be zero. These were necessary boundary conditions and constants within our derivations to solve for the constants of integration. From our experimental data, we observed the deflection at the center of the beam to be $1.549mm$. Knowing this and plugging all possible combinations of locations into MATLAB, we noticed that having a force at $0.5m$ and $3.5m$ resulted in a deflection of $1.456mm$ at the center of the beam. This tells us that these locations for force one and force two are correct with approximately 6.093% error.

For this case, the experimental internal force was $266.9N$ where the internal force for the equivalent beam model was $341.8N$. This results in an error of 28.1%. These values can be seen in Figure 14. This shows that the equivalent beam model for this case is a better approximation then it was for the previous case, but it is still not a great approximation. This could be for the same reason as discussed for the previous case. The reason why this might be a better approximation than the previous case is due to the fact that the resultant force of the two loads is closer to the center of the beam, therefore better satisfying the equivalent beam model.

IV. Discussion

A. Uncertainty Analysis

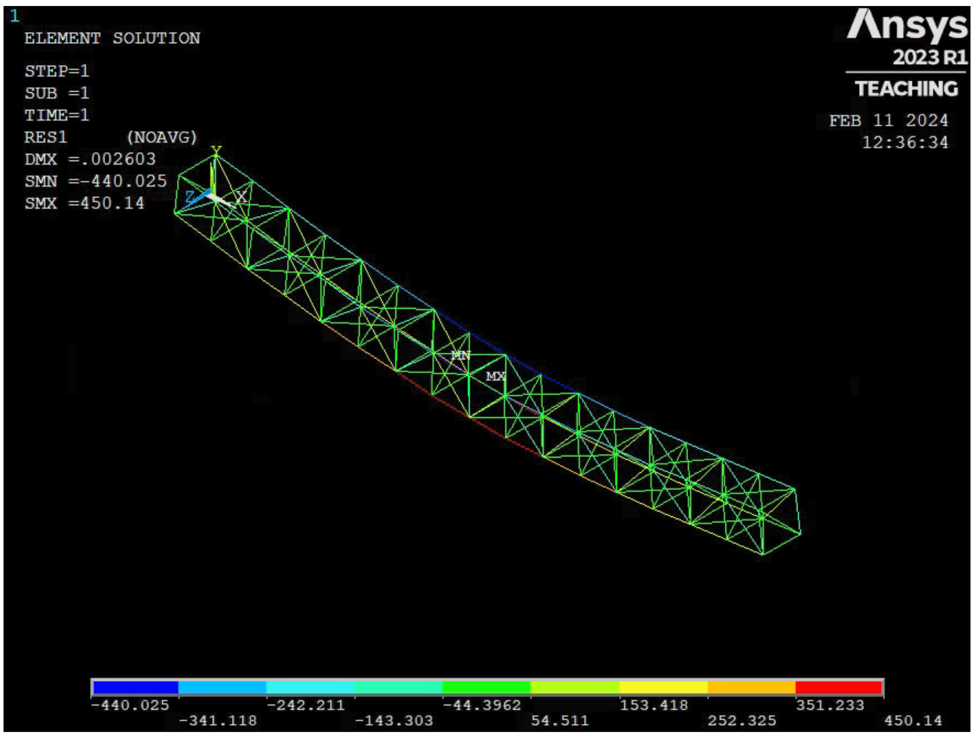


Fig. 15 Reduced Stiffness in Bars Contour Plot

The first source of error tested was reduced stiffness in the bars due to imperfect joints, such as free-play. This mean that there is some degree of movement or flexibility in the bars within the truss. The ANSYS FEM analysis of this source of error can be seen in Figure 15. For this test, the Young's Modulus was set to $50GPa$, down from the $70GPa$ that is a material feature of the 6061-T6 Aluminum bars used. While the internal forces and reaction forces were the same from loading case 1, $55.66N$ and $450.14N$ respectively, the midpoint deflection of the beam increases by 71.1% to $2.60mm$. This source of error sees the largest change in midpoint deflection out of all the cases tested, making it the most important source of error. The results for all sources of error can be seen in the table below.

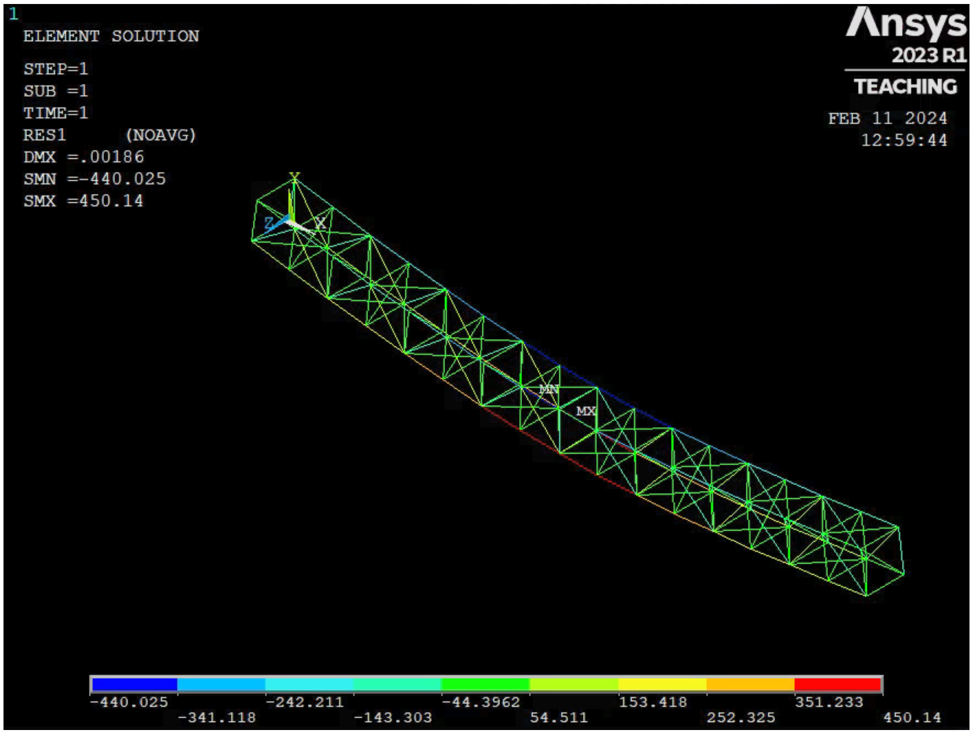


Fig. 16 Reduced Stiffness in Bar with Load Cell Contour Plot

The next source of error tested was reduced stiffness in the bar containing the load cell. This is a fair assessment of error do to the fact that the load cell introduces additional joints within the bar and is not made of 6061-T6 Aluminum. The Young's Modulus was also set to $50GPa$, but only for this one bar. The reaction forces and internal forces were identical to that of the first case. Additionally, the midpoint deflection of the beam was $1.85mm$. The original model had the same deflection, so there was no change between ANSYS models for this source of error. The most likely reason due to this is that only 1 of the 213 elements was changed, and with the model using links, the whole truss was not really affected. The contour plot for this test can be seen in Figure 16.

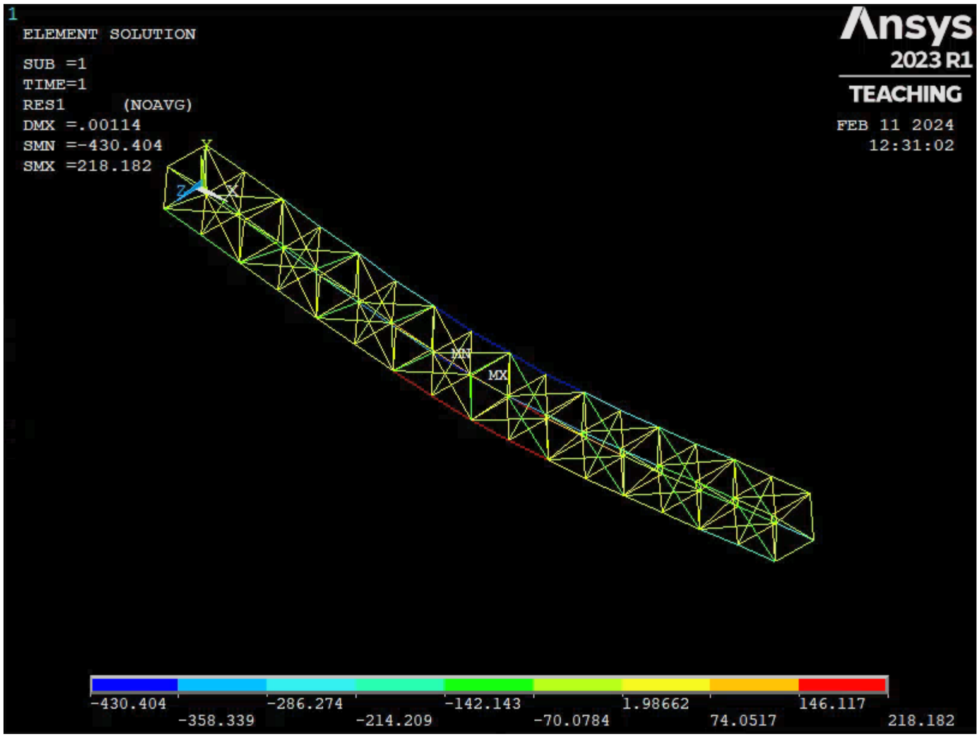


Fig. 17 Increased Friction at Joints Contour Plot

The third source of error tested was increased friction at the supported joints. Specifically, infinite friction at the joints was tested. This was modeled by assuming the roller joint was actually a pin joint, resulting in two pin joints for the truss. This test resulted in a far lower value of a maximum internal force of $229.38N$, down from 450.14, in tension, and a slightly lower value for compression with a value of $431.19N$, down from $440.02N$. That is a 49% decrease in tension and a 2% decrease in compression. The midpoint deflection also decreased from $1.85mm$ to $1.17mm$, which is a 36.75% decrease. While there is friction at the supported joints in the actual experiment, assuming that it is infinite is not necessarily an accurate assumption. While the results are interesting, it actually decreases the midpoint deflection and internal forces, which means assuming there is no friction at the roller joint is actually a better assumption than infinite friction, especially when doing the factor of safety calculation. The corresponding contour plot for this test is shown in Figure 17.

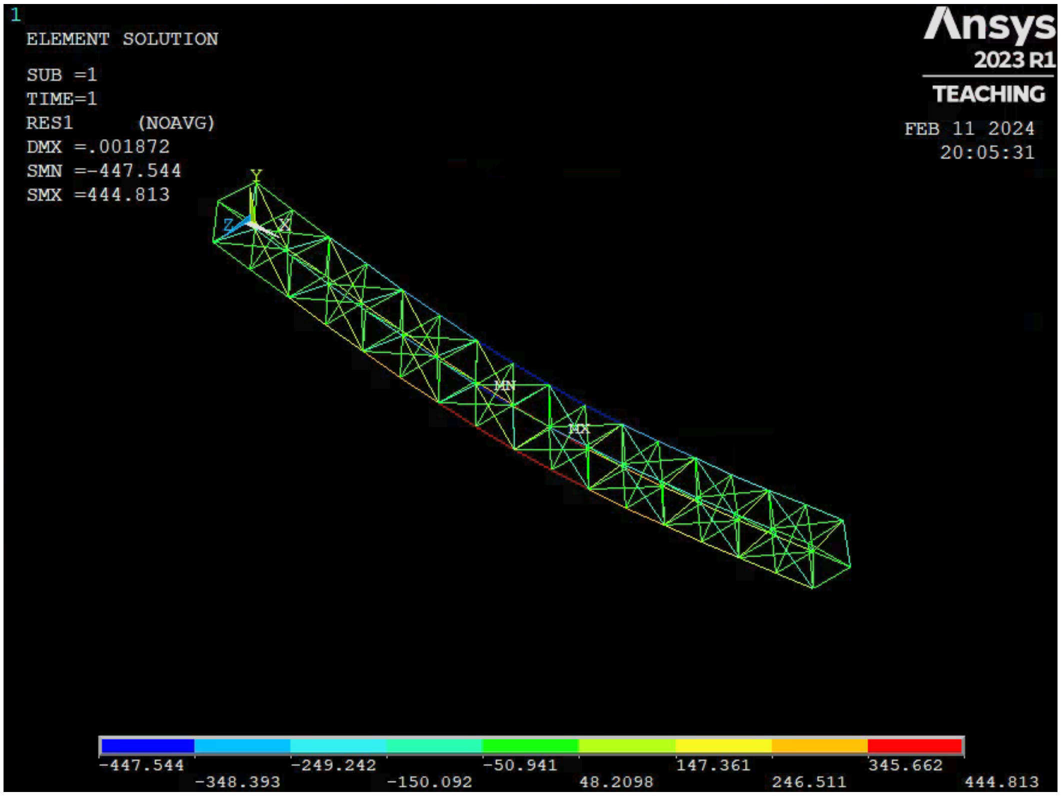


Fig. 18 Manufacturing Imperfections Contour Plot

The fourth condition tested for a possible source of error was manufacturing imperfections. To conduct this analysis, certain bars were given slightly lower and slightly higher values in length than the given value of $250mm$. The range of values used was between $240mm$ and $260mm$ and the contour plot for this model can be seen in Figure 18. While this leads to slightly deformed model of the truss, the actual results do not change by a large degree, which came as a surprise. The maximum internal tension force decreased by $5.33N$,the maximum internal compression force increased by $7.52N$, and the midpoint deflection only increased by $0.02mm$. While the change in the reaction forces was also relatively small, it was the only source of error that caused them to change by more than $1N$. On the left side of the truss, nodes 1 and 52, went from $55.56N$ to $57.62N$ and $55.63N$ to $53.68N$ respectively. On the right side of the truss, nodes 17 and 68, went from $55.63N$ to $51.35N$ and $55.56N$ to $59.56N$ respectively. This is a 3% increase in node 1, 7.37% decrease in node 17, 3.18% decrease in node 52, and 7.19% increase in node 63. A range of $20mm$ is a pretty large range of imperfection by machined parts, so these small increases in values between models would indicate that a typical manufacturing imperfection would lead to very small sources of error.

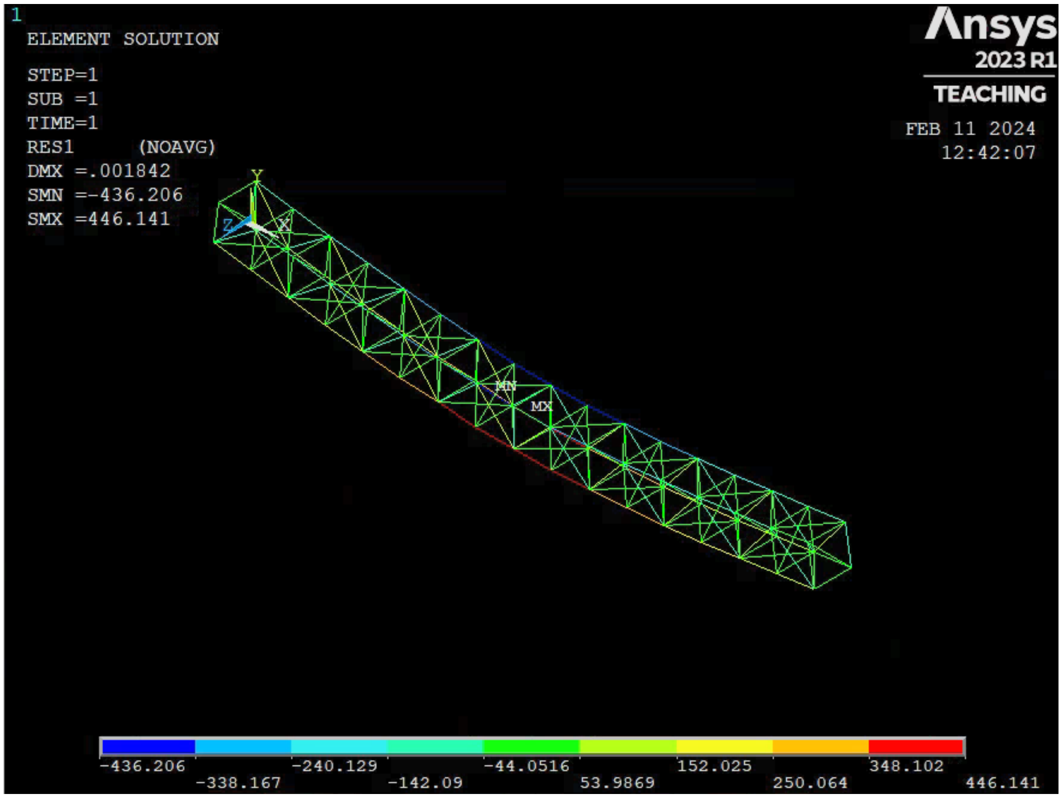


Fig. 19 Circular Tube Contour Plot

The final source of error simulated was to change the assumption that the bars act as links, and rather were the circular tubes and that the joints could transmit bending moments and the bars have bending stiffness. This was simply done by changing the cross section type in ANSYS and rerunning the simulation. Surprisingly, the change in the element type of the bar had very little to no effect on the results. The internal forces both decreased by about $4N$, the midpoint deflection decreased by $0.01mm$, and the internal forces fluctuated by $\pm 0.01N$. These minuscule changes prove that the assumption of the bars act as links instead of circular tubes is a good assumption to make for the ANSYS model. The results for this test are shown in Figure 19.

Table 6 Uncertainty Analysis of Different Test Conditions

Test Condition	Max Deflection [mm]	Max Internal Forces [N]	Reaction Forces [N]
Original Test	1.85	Tension: 450.14 Compression: 440.02	Node 1: 55.56 Node 17: 55.63 Node 52: 55.63 Node 68: 55.56
Reduced Stiffness in Bars	2.60	Tension: 450.14 Compression: 440.02	Node 1: 55.56 Node 17: 55.63 Node 52: 55.63 Node 68: 55.56
Reduced Stiffness in Bar with Load Cell	1.85	Tension: 450.14 Compression: 440.02	Node 1: 55.56 Node 17: 55.63 Node 52: 55.63 Node 68: 55.56
Increased Friction at Support Joints	1.17	Tension: 229.38 Compression: 431.19	Node 1: 55.6 Node 17: 55.6 Node 52: 55.6 Node 68: 55.6
Manufacturing Imperfections	1.87	Tension: 444.81 Compression: 447.54	Node 1: 57.62 Node 17: 51.35 Node 52: 53.86 Node 68: 59.56
Circular Tube Model	1.84	Tension: 446.14 Compression: 436.21	Node 1: 55.55 Node 17: 55.64 Node 52: 55.64 Node 68: 55.55

After modeling the 5 above sources of error in ANSYS, the most likely cause for the differences between the FEM model and the experimental results is due to increased friction at the support joints. It is the only one of the tested cases that leads to a lower maximum deflection and lower maximum internal force, which is the case for the experimental results, while keeping the reaction forces almost equal to the experimental results. While the assumption of no friction is better than the assumption of infinite friction for the FEM model, it is safe to assume there is some degree of friction at the roller joint that results in a decrease for both midpoint deflection and maximum internal force.

B. Factor of Safety

When determining the optimal factor of safety for the 16-bay truss, we assessed the uncertainty condition that resulted in the highest change in deflection. This occurred under conditions simulating reduced stiffness in bars ($E = 50GPa$). By dividing the experimental Young's Modulus by the reduced stiffness Young's Modulus, we obtained a ratio of 1.4. However, it's important to note that this approach to determining the factor of safety only considers one potential source of error. In reality, various sources of error could be present simultaneously or in combination with each other. Thus, a more comprehensive factor of safety should account for multiple sources of error within the truss system. Based on our analysis, a factor of safety ranging from 1.5 to 1.8 is considered ideal for addressing potential uncertainties effectively and is around the typical factor of safety values in the aerospace industry. If this were a part of a structure where loss of life or mission failure was a possibility if it failed, the factor of safety would scale accordingly.

V. Conclusions

The point of this lab was to analyze reaction forces, internal forces, and midpoint deflection of a truss structure through the use of three models; an experimental model, an analytical model, and an ANSYS model. By using these models we were able to predict the locations of unknown loading forces exerted on the truss. We were given experimental data about the reaction forces and the deflection of the truss as an unknown load was applied. We were able to use the equivalent beam model to create a prediction for the locations of these loading forces. It's important to note however that when using any model, there will always be error and uncertainty. Because of this, it is important to create a factor of safety to account for the error when analyzing the exact locations of these forces. This is even more important when analyzing structures for points of failure. We explored a range of possible causes of uncertainty using the ANSYS model and derived a reasonable factor of safety based on the level of uncertainty calculated within our models.

A. Appendix

A. Complete derivations

1. Case 2/Unknown Case 1

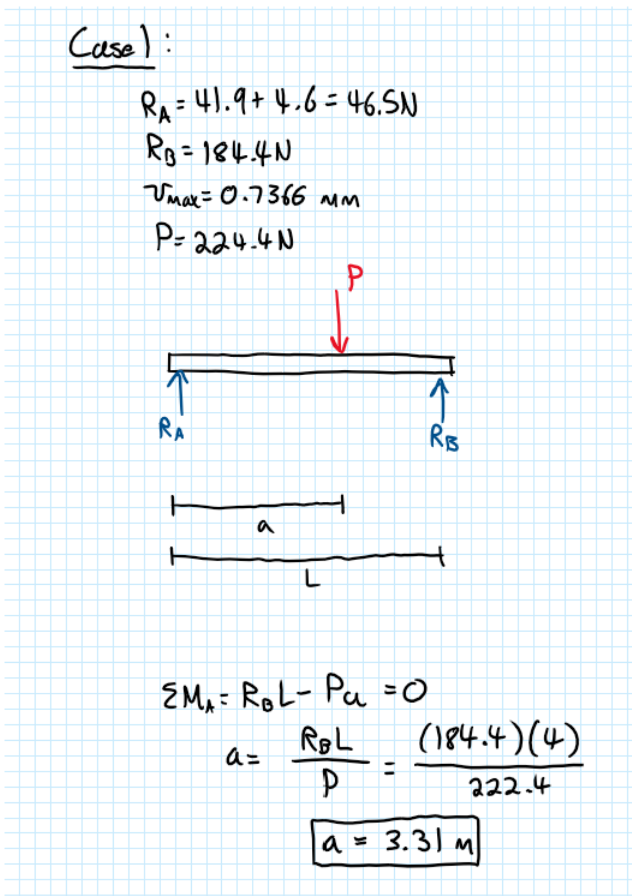
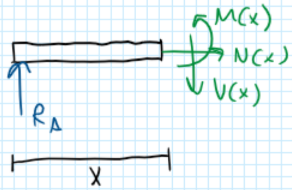


Fig. 20 Unknown Case 1 Part 1 Derivation

Solving for deflection at center:



$$\Sigma M_x = M(x) - R_A x = 0 \rightarrow M(x) = R_A x$$

$$\theta(x) = \frac{1}{EI}\int M(x)\,dx$$

$$= \frac{1}{EI}\left[\frac{R_A}{2}x^2 + C_1\right]$$

$$v(x) = \frac{1}{EI}\int \theta(x)\,dx$$

$$v(x) = \frac{1}{EI}\left[\frac{R_A}{6}x^3 + C_1 x + C_2\right]$$

$$v(0) = \frac{1}{EI}C_2 = 0 \rightarrow C_2 = 0$$

$$\theta(3.25) = \frac{1}{EI}\left[5.281 R_A + C_1\right] = 0$$

$$C_1 = -5.281 RA$$

$$v(x) = \frac{1}{EI}\left[\frac{R_A}{6}x^3 - 5.281 R_A x\right]$$

at midpoint:

$$v(2) = \frac{1}{EI}\left[\frac{R_A}{6}(2)^3 - 5.281 R_A (2)\right]$$

$$v(2) = -0.24\,mm$$

Fig. 21 Unknown Case 1 Part 2 Derivation

2. Case 3/Unknown Case 2

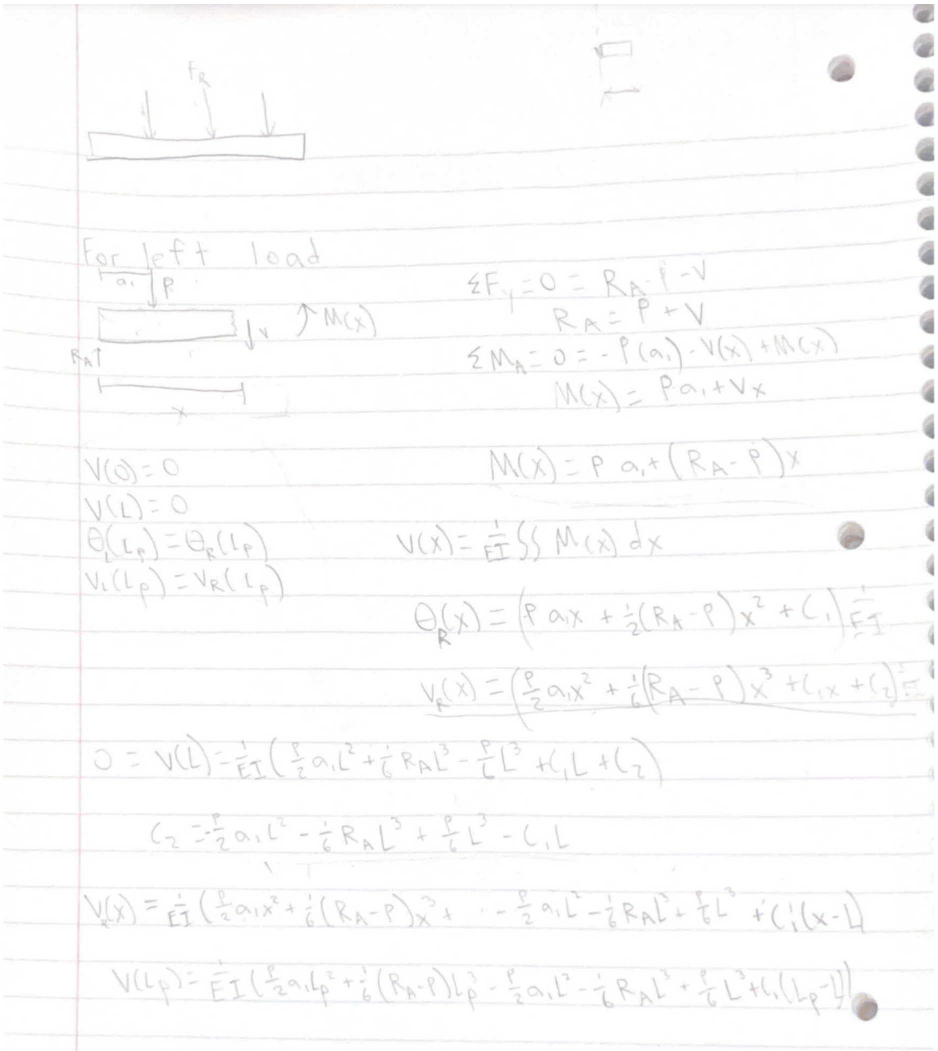
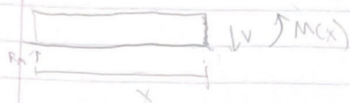


Fig. 22 Unknown Case 2 Part 1 Derivation

For right load

$\Sigma F_y = 0 = R_A - V$

$R_A = V$



$\Sigma M_A = 0 = -Vx + M(x)$

$M(x) = Vx$

$V_L(x) = \frac{1}{EI}\iint M(x)\,dx$

$\theta_L(x) = \frac{1}{EI}\left(\frac{1}{2}R_A x^2 + C_3\right)$

$\theta_L(L_p) = \frac{1}{EI}\left(\frac{1}{2}R_A L_p^2 + C_3\right)$

$V_L(x) = \frac{1}{EI}\left(\frac{1}{6}Vx^3 + C_3 x + C_4\right)$

$C_4 = -\frac{1}{6}R_A L^3 + C_3 L$

$V_L(x) = \frac{1}{EI}\left(\frac{1}{6}R_A x^3 + C_3 x + C_4\right.$

$V_L(L_p) = \frac{1}{EI}\left(\frac{1}{6}R_A L_p^3 + C_3 L_p + C_4\right)$

Fig. 23 Unknown Case 2 Part 2 Derivation

$V_L(L_p) = V_R(L_p)$

$\cancel{\frac{1}{6}R_A L_p^3} + C_3 L_p = \frac{P}{2}a_1 L_p^2 + \cancel{\frac{1}{6}R_A L_p^3} - \frac{1}{6}PL_p^3 - \frac{P}{2}a_1 L^2 - \frac{1}{6}R_A L^3$
$+ \frac{P}{6}L^3 + C_1(L_p - L)$

$C_3 L_p = \frac{P}{2}a_1 L_p - \frac{1}{6}PL_p^3 - \frac{P}{2}a_1 L^2 - \frac{1}{6}R_A L^3 + \frac{P}{6}L^3 + C_1(L_p - L)$

$C_3 = \frac{P}{2}a_1 - \frac{1}{6}PL_p^2 + \left(\frac{1}{L_p}\right)\left(-\frac{P}{2}a_1 L^2 - \frac{1}{6}R_A L^3 + \frac{P}{6}L^3 + C_1(L_p - L)\right)$

Fig. 24 Unknown Case 2 Part 3 Derivation

$\theta_R(L_p) = \theta_L(L_p)$

$Pa_1 L_p + \cancel{\frac{1}{2}R_A L_p^2} - \frac{P}{2}L_p^2 + C_1 = \cancel{\frac{1}{2}R_A L_p^2} + C_3$

$Pa_1 L_p - \frac{P}{2}L_p^2 + C_1 = C_3$

$C_1 = C_3 - Pa_1 L_p + \frac{P}{2}L_p^2$

$C_1(L_p - L) = C_3 L_p - C_3 L - Pa_1 L_p^2 + Pa_1 L_p L + \frac{P}{2}L_p^3 - \frac{P}{2}L_p^2 L$

$C_3 - C_3 L_p + C_3 L = C_3(1 - L_p + L)$

$$C_3 = \frac{\frac{P}{2}a_1 - \frac{1}{6}PL_p^2 + \left(\frac{1}{L_p}\right)\left(-\frac{P}{2}a_1 L^2 - \frac{1}{6}R_A L^3 + \frac{P}{6}L^3 - Pa_1 L_p^2 + Pa_1 L_p L + \frac{P}{2}L_p^3 - \frac{P}{2}L_p^2 L\right)}{(1 - L_p + L)}$$

Fig. 25 Unknown Case 2 Part 4 Derivation

B. Matlab Code

```
%% House Keeping
clear,clc,close all

%% Read In Data
case_1_data = readmatrix("Case 1.txt");
case_2_data = readmatrix("Case 2.txt");
case_3_data = readmatrix("Case 3.txt");

case_1_data = case_1_data * 4.4482216153;
case_2_data = case_2_data * 4.4482216153;
case_3_data = case_3_data * 4.4482216153;


%% Sort Data

% Applied force for each case
case_1_applied_force = case_1_data(:,1);
case_2_applied_force = case_2_data(:,1);
case_3_applied_force = case_3_data(:,1);

% Reaction forces for each case
case_1_load_sensor_1 = case_1_data(:,2); % Case 1
case_1_load_sensor_2 = case_1_data(:,3);
case_1_load_sensor_3 = case_1_data(:,4);

case_2_load_sensor_1 = case_2_data(:,2); % Case 2
case_2_load_sensor_2 = case_2_data(:,3);
case_2_load_sensor_3 = case_2_data(:,4);

case_3_load_sensor_1 = case_3_data(:,2); % Case 3
case_3_load_sensor_2 = case_3_data(:,3);
case_3_load_sensor_3 = case_3_data(:,4);

% Internal force for each case
case_1_inline_load_cells = case_1_data(:,5) - mean(case_1_data(1:10,5));
case_2_inline_load_cells = case_2_data(:,5) - mean(case_2_data(1:10,5));
case_3_inline_load_cells = case_3_data(:,5) - mean(case_3_data(1:10,5));

% Displacement for each case
case_1_LVDT = (case_1_data(:,6)./4.4482216153)*25.4;
case_2_LVDT = (case_2_data(:,6)./4.4482216153)*25.4;
case_3_LVDT = (case_3_data(:,6)./4.4482216153)*25.4;

%% Finding Loading Location

% Both cases assume that loading sensor 3 = Rb
% Case 2
L = 4;
a1 = (L.*case_2_load_sensor_3)./case_2_applied_force;

% Case 3
a2 = (L.*case_3_load_sensor_3)./case_3_applied_force;

E = 70E9;
I = 2.4761E-6;
RA = max(case_3_load_sensor_1 + case_3_load_sensor_2);
RB = max(case_3_load_sensor_3);
```

```
P = (RA+RB)/2;
L = 4;
Lp = 2;
a = 0:.25:2;
% a3 = 0.08:0.01:2.04;
% a4 = 4:-0.01:2.04;
%
% v_left = 1/(E*I) .*((RA/6)*x^3 - (RA/2) .* a3.^2.*x);
% v_right = 1/(E*I) .*((RA/6 -
    P/12)*x^3+(P/4).*a3.*x^2+((P/4).*a4.^2-(P/2).*a3.*a4-(RA/2).*a4.^2).*x);
% v_left = v_left .* 1000;
% v_right = v_right .* 1000;
%
% v_tot = v_left + v_right;
%
v_c = max(case_3_LVDT);

C3 = ((P/2).*a -(1/6)*P*(Lp^2)+(1/Lp).*((-P/2).*a.*(L.^2) -
    (1/6)*RA*(L^3)+(P/6)*(L^3)-P.*a.*(Lp.^2)+P.*a.*L.*Lp+(P/2)*Lp^3-(P/2)*(Lp^2)*L))./(1-Lp+L);
C1 = C3 -P.*a.*Lp+(P/2)*Lp;
C2 = (-P/2).*a.*L.^2 - (1/6)*RA*L^3+(P/6)*L^3-C1.*L;

v_r = (1/(E*I)).*((P/2).*a.*Lp^2 + (1/6)*(RA-P)*(Lp^3)+C1*Lp+C2);
v_l = (1/(E*I)).*((1/6)*RA*(Lp^3)+C3*Lp);

v_tot = v_l + v_r;

location1 = a(3); %m
location2 = (L - location1); %m

errorv = abs((-v_tot(3)*1000 - v_c)/v_c) *100;


% %% Linear Regression

% Case 1
P_R1_1 = polyfit(case_1_applied_force, case_1_load_sensor_1, 1);
P_R2_1 = polyfit(case_1_applied_force, case_1_load_sensor_2, 1);
P_R3_1 = polyfit(case_1_applied_force, case_1_load_sensor_3, 1);
P_I_1 = polyfit(case_1_applied_force, case_1_inline_load_cells, 1);
P_D_1 = polyfit(case_1_applied_force, case_1_LVDT, 1);

fit_R1_1 = polyval(P_R1_1, case_1_applied_force);
fit_R2_1 = polyval(P_R2_1, case_1_applied_force);
fit_R3_1 = polyval(P_R3_1, case_1_applied_force);
fit_I_1 = polyval(P_I_1, case_1_applied_force);
fit_D_1 = polyval(P_D_1, case_1_applied_force);

% Case 2
P_R1_2 = polyfit(case_2_applied_force, case_2_load_sensor_1, 1);
P_R2_2 = polyfit(case_2_applied_force, case_2_load_sensor_2, 1);
P_R3_2 = polyfit(case_2_applied_force, case_2_load_sensor_3, 1);
P_I_2 = polyfit(case_2_applied_force, case_2_inline_load_cells, 1);
P_D_2 = polyfit(case_2_applied_force, case_2_LVDT, 1);

fit_R1_2 = polyval(P_R1_2, case_2_applied_force);
fit_R2_2 = polyval(P_R2_2, case_2_applied_force);
fit_R3_2 = polyval(P_R3_2, case_2_applied_force);
fit_I_2 = polyval(P_I_2, case_2_applied_force);
```

```
fit_D_2 = polyval(P_D_2, case_2_applied_force);

% Case 3
P_R1_3 = polyfit(case_3_applied_force, case_3_load_sensor_1, 1);
P_R2_3 = polyfit(case_3_applied_force, case_3_load_sensor_2, 1);
P_R3_3 = polyfit(case_3_applied_force, case_3_load_sensor_3, 1);
P_I_3 = polyfit(case_3_applied_force, case_3_inline_load_cells, 1);
P_D_3 = polyfit(case_3_applied_force, case_3_LVDT, 1);

fit_R1_3 = polyval(P_R1_3, case_3_applied_force);
fit_R2_3 = polyval(P_R2_3, case_3_applied_force);
fit_R3_3 = polyval(P_R3_3, case_3_applied_force);
fit_I_3 = polyval(P_I_3, case_3_applied_force);
fit_D_3 = polyval(P_D_3, case_3_applied_force);

%% Standard Deviation of the forces
N = 110; %The number of measurements taken

%Reaction Force #1
sigma_force_R1_1 = ((1)/(N-1)) .* (mean(case_1_load_sensor_1) - case_1_load_sensor_1).^2;
sigma_force_R1_C1 = sqrt(sigma_force_R1_1);

sigma_force_R1_2 = ((1)/(N-1)) .* (mean(case_1_load_sensor_2) - case_1_load_sensor_2).^2;
sigma_force_R1_C2 = sqrt(sigma_force_R1_2);

sigma_force_R1_3 = ((1)/(N-1)) .* (mean(case_1_load_sensor_3) - case_1_load_sensor_3).^2;
sigma_force_R1_C3 = sqrt(sigma_force_R1_3);


%Reaction Force #2
sigma_force_R2_1 = ((1)/(N-1)) .* (mean(case_2_load_sensor_1) - case_2_load_sensor_1).^2;
sigma_force_R2_C1 = sqrt(sigma_force_R2_1);

sigma_force_R2_2 = ((1)/(N-1)) .* (mean(case_2_load_sensor_2) - case_2_load_sensor_2).^2;
sigma_force_R2_C2 = sqrt(sigma_force_R2_2);

sigma_force_R2_3 = ((1)/(N-1)) .* (mean(case_2_load_sensor_3) - case_2_load_sensor_3).^2;
sigma_force_R2_C3 = sqrt(sigma_force_R2_3);

%Reaction Force #3
sigma_force_R3_1 = ((1)/(N-1)) .* (mean(case_3_load_sensor_1) - case_3_load_sensor_1).^2;
sigma_force_R3_C1 = sqrt(sigma_force_R3_1);

N3 = 70;
sigma_force_R3_2 = ((1)/(N3-1)) .* (mean(case_3_load_sensor_2) - case_3_load_sensor_2).^2;
sigma_force_R3_C2 = sqrt(sigma_force_R3_2);

sigma_force_R3_3 = ((1)/(N-1)) .* (mean(case_3_load_sensor_3) - case_3_load_sensor_3).^2;
sigma_force_R3_C3 = sqrt(sigma_force_R3_3);

%Internal Forces
Sigma_internal_force_1 = ((1)/(N-1)) .* (mean(case_1_inline_load_cells) -
    case_1_inline_load_cells).^2;
Sigma_internal_force_C1 = sqrt(Sigma_internal_force_1);

Sigma_internal_force_2 = ((1)/(N-1)) .* (mean(case_2_inline_load_cells) -
    case_2_inline_load_cells).^2;
Sigma_internal_force_C2 = sqrt(Sigma_internal_force_2);
```

```
Sigma_internal_force_3 = ((1)/(N-1)) .* (mean(case_3_inline_load_cells) -
    case_3_inline_load_cells).^2;
Sigma_internal_force_C3 = sqrt(Sigma_internal_force_3);

%Displacement
Sigma_displacement_1 = ((1)/(N-1)) .* (mean(case_1_LVDT) - case_1_LVDT).^2;
Sigma_displacement_C1 = sqrt(Sigma_displacement_1);

Sigma_displacement_2 = ((1)/(N-1)) .* (mean(case_2_LVDT) - case_2_LVDT).^2;
Sigma_displacement_C2 = sqrt(Sigma_displacement_2);

Sigma_displacement_3 = ((1)/(N3-1)) .* (mean(case_3_LVDT) - case_3_LVDT).^2;
Sigma_displacement_C3 = sqrt(Sigma_displacement_3);


%% Uncertainty Analysis


%% R^2

% SSR Values
SSR_R1_1 = 0; % Case 1
SSR_R2_1 = 0;
SSR_R3_1 = 0;
SSR_I_1 = 0;
SSR_D_1 = 0;

SSR_R1_2 = 0; % Case 2
SSR_R2_2 = 0;
SSR_R3_2 = 0;
SSR_I_2 = 0;
SSR_D_2 = 0;

SSR_R1_3 = 0; % Case 3
SSR_R2_3 = 0;
SSR_R3_3 = 0;
SSR_I_3 = 0;
SSR_D_3 = 0;

% Find sum of the residuals
for i = 1:length(case_1_load_sensor_1) % Case 1 and 2

   SSR_R1_1 = SSR_R1_1 + ((case_1_load_sensor_1(i) - fit_R1_1(i)).^2);
   SSR_R2_1 = SSR_R2_1 + ((case_1_load_sensor_2(i) - fit_R2_1(i)).^2);
   SSR_R3_1 = SSR_R3_1 + ((case_1_load_sensor_3(i) - fit_R3_1(i)).^2);
   SSR_I_1 = SSR_I_1 + ((case_1_inline_load_cells(i) - fit_I_1(i)).^2);
   SSR_D_1 = SSR_D_1 + ((case_1_LVDT(i) - fit_D_1(i)).^2);

   SSR_R1_2 = SSR_R1_2 + ((case_2_load_sensor_1(i) - fit_R1_2(i)).^2);
   SSR_R2_2 = SSR_R2_2 + ((case_2_load_sensor_2(i) - fit_R2_2(i)).^2);
   SSR_R3_2 = SSR_R3_2 + ((case_2_load_sensor_3(i) - fit_R3_2(i)).^2);
   SSR_I_2 = SSR_I_2 + ((case_2_inline_load_cells(i) - fit_I_2(i)).^2);
   SSR_D_2 = SSR_D_2 + ((case_2_LVDT(i) - fit_D_2(i)).^2);

end

for i = 1:length(case_3_load_sensor_1) % Case 3

   SSR_R1_3 = SSR_R1_3 + ((case_3_load_sensor_1(i) - fit_R1_3(i)).^2);
```

```
    SSR_R2_3 = SSR_R2_3 + ((case_3_load_sensor_2(i) - fit_R2_3(i)).^2);
    SSR_R3_3 = SSR_R3_3 + ((case_3_load_sensor_3(i) - fit_R3_3(i)).^2);
    SSR_I_3 = SSR_I_3 + ((case_3_inline_load_cells(i) - fit_I_3(i)).^2);
    SSR_D_3 = SSR_D_3 + ((case_3_LVDT(i) - fit_D_3(i)).^2);
end

% SST values
SST_R1_1 = 0; % Case 1
SST_R2_1 = 0;
SST_R3_1 = 0;
SST_I_1 = 0;
SST_D_1 = 0;

SST_R1_2 = 0; % Case 2
SST_R2_2 = 0;
SST_R3_2 = 0;
SST_I_2 = 0;
SST_D_2 = 0;

SST_R1_3 = 0; % Case 3
SST_R2_3 = 0;
SST_R3_3 = 0;
SST_I_3 = 0;
SST_D_3 = 0;

% Mean values
M_R1_1 = mean(case_1_load_sensor_1); % Case 1
M_R2_1 = mean(case_1_load_sensor_2);
M_R3_1 = mean(case_1_load_sensor_3);
M_I_1 = mean(case_1_inline_load_cells);
M_D_1 = mean(case_1_LVDT);

M_R1_2 = mean(case_2_load_sensor_1); % Case 2
M_R2_2 = mean(case_2_load_sensor_2);
M_R3_2 = mean(case_2_load_sensor_3);
M_I_2 = mean(case_2_inline_load_cells);
M_D_2 = mean(case_2_LVDT);

M_R1_3 = mean(case_3_load_sensor_1); % Case 3
M_R2_3 = mean(case_3_load_sensor_2);
M_R3_3 = mean(case_3_load_sensor_3);
M_I_3 = mean(case_3_inline_load_cells);
M_D_3 = mean(case_3_LVDT);

% Finding SST Values
for i = 1:length(case_1_load_sensor_1) % Case 1 and 2

    SST_R1_1 = SST_R1_1 + ((case_1_load_sensor_1(i) - M_R1_1).^2);
    SST_R2_1 = SST_R2_1 + ((case_1_load_sensor_2(i) - M_R2_1).^2);
    SST_R3_1 = SST_R3_1 + ((case_1_load_sensor_3(i) - M_R3_1).^2);
    SST_I_1 = SST_I_1 + ((case_1_inline_load_cells(i) - M_I_1).^2);
    SST_D_1 = SST_D_1 + ((case_1_LVDT(i) - M_D_1).^2);

    SST_R1_2 = SST_R1_2 + ((case_2_load_sensor_1(i) - M_R1_2).^2);
    SST_R2_2 = SST_R2_2 + ((case_2_load_sensor_2(i) - M_R2_2).^2);
    SST_R3_2 = SST_R3_2 + ((case_2_load_sensor_3(i) - M_R3_2).^2);
    SST_I_2 = SST_I_2 + ((case_2_inline_load_cells(i) - M_I_2).^2);
    SST_D_2 = SST_D_2 + ((case_2_LVDT(i) - M_D_2).^2);

```

```matlab
end

for i = 1:length(case_3_load_sensor_1) % Case 3

    SST_R1_3 = SST_R1_3 + ((case_3_load_sensor_1(i) - M_R1_3).^2);
    SST_R2_3 = SST_R2_3 + ((case_3_load_sensor_2(i) - M_R2_3).^2);
    SST_R3_3 = SST_R3_3 + ((case_3_load_sensor_3(i) - M_R3_3).^2);
    SST_I_3 = SST_I_3 + ((case_3_inline_load_cells(i) - M_I_3).^2);
    SST_D_3 = SST_D_3 + ((case_3_LVDT(i) - M_D_3).^2);

end

% Finding R^2 Values
case_1_R_Squared(1) = 1 - (SSR_R1_1./SST_R1_1); % Case 1
case_1_R_Squared(2) = 1 - (SSR_R2_1./SST_R2_1);
case_1_R_Squared(3) = 1 - (SSR_R3_1./SST_R3_1);
case_1_R_Squared(4) = 1 - (SSR_I_1./SST_I_1);
case_1_R_Squared(5) = 1 - (SSR_D_1./SST_D_1);

case_2_R_Squared(1) = 1 - (SSR_R1_2./SST_R1_2); % Case 2
case_2_R_Squared(2) = 1 - (SSR_R2_2./SST_R2_2);
case_2_R_Squared(3) = 1 - (SSR_R3_2./SST_R3_2);
case_2_R_Squared(4) = 1 - (SSR_I_2./SST_I_2);
case_2_R_Squared(5) = 1 - (SSR_D_2./SST_D_2);

case_3_R_Squared(1) = 1 - (SSR_R1_3./SST_R1_3); % Case 3
case_3_R_Squared(2) = 1 - (SSR_R2_3./SST_R2_3);
case_3_R_Squared(3) = 1 - (SSR_R3_3./SST_R3_3);
case_3_R_Squared(4) = 1 - (SSR_I_3./SST_I_3);
case_3_R_Squared(5) = 1 - (SSR_D_3./SST_D_3);


%% Plots

figure(1) % Reaction Force 1
hold on

%scatter(case_1_applied_force, case_1_load_sensor_1,"LineWidth", 1.2) %Blue
errorbar(case_1_applied_force,case_1_load_sensor_1,ones(size(sigma_force_R1_C1)))
%scatter(case_2_applied_force, case_2_load_sensor_1,"LineWidth", 1.2) %Red
errorbar(case_2_applied_force,case_2_load_sensor_1,sigma_force_R2_C1)
%scatter(case_3_applied_force, case_3_load_sensor_1,"LineWidth", 1.2) %Yellow
errorbar(case_3_applied_force,case_3_load_sensor_1,sigma_force_R3_C1)

plot(case_1_applied_force, fit_R1_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])
plot(case_2_applied_force, fit_R1_2,"LineWidth", 1.2, "Color", [0.8500 0.3250 0.0980])
plot(case_3_applied_force, fit_R1_3,"LineWidth", 1.2, "Color", [0.9290 0.6940 0.1250])



xlabel("Applied Force (N)")
ylabel("Measured Reaction Force (N)")
title("Reaction Force 1 vs Applied Force")
legend('Case 1', 'Case 2', 'Case 3')
hold off


figure(2) % Reaction Force 2
hold on
```

```
%scatter(case_1_applied_force, case_1_load_sensor_2,"LineWidth", 1.2)
errorbar(case_1_applied_force,case_1_load_sensor_2,ones(size(sigma_force_R1_C2)))
%errorbar(case_1_applied_force,case_1_load_sensor_2,sigma_force_R3_C3)
%scatter(case_2_applied_force, case_2_load_sensor_2,"LineWidth", 1.2)
errorbar(case_2_applied_force,case_2_load_sensor_2,ones(size(sigma_force_R2_C2)))
%scatter(case_3_applied_force, case_3_load_sensor_2,"LineWidth", 1.2)
errorbar(case_3_applied_force,case_3_load_sensor_2,ones(size(sigma_force_R3_C2)))

plot(case_1_applied_force, fit_R2_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])
plot(case_2_applied_force, fit_R2_2,"LineWidth", 1.2, "Color", [0.8500 0.3250 0.0980])
plot(case_3_applied_force, fit_R2_3,"LineWidth", 1.2, "Color", [0.9290 0.6940 0.1250])

xlabel("Applied Force (N)")
ylabel("Measured Reaction Force (N)")
title("Reaction Force 2 vs Applied Force")
legend('Case 1', 'Case 2', 'Case 3')
hold off

figure(3) % Reaction Force 3

hold on
%scatter(case_1_applied_force, case_1_load_sensor_3,"LineWidth", 1.2)
errorbar(case_1_applied_force,case_1_load_sensor_3,ones(size(sigma_force_R1_C3)))
%scatter(case_2_applied_force, case_2_load_sensor_3,"LineWidth", 1.2)
errorbar(case_2_applied_force,case_2_load_sensor_3,ones(size(sigma_force_R2_C3)))
%scatter(case_3_applied_force, case_3_load_sensor_3,"LineWidth", 1.2)
errorbar(case_3_applied_force,case_3_load_sensor_3,ones(size(sigma_force_R3_C3)))

plot(case_1_applied_force, fit_R3_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])
plot(case_2_applied_force, fit_R3_2,"LineWidth", 1.2, "Color", [0.8500 0.3250 0.0980])
plot(case_3_applied_force, fit_R3_3,"LineWidth", 1.2, "Color", [0.9290 0.6940 0.1250])

xlabel("Applied Force (N)")
ylabel("Measured Reaction Force (N)")
title("Reaction Force 3 vs Applied Force")
legend('Case 1', 'Case 2', 'Case 3')
hold off

figure(4) % Internal Force

hold on
%scatter(case_1_applied_force, case_1_inline_load_cells,"LineWidth", 1.2)
errorbar(case_1_applied_force,case_1_inline_load_cells,ones(size(Sigma_internal_force_C1)))
%scatter(case_2_applied_force, case_2_inline_load_cells,"LineWidth", 1.2)
errorbar(case_2_applied_force,case_2_inline_load_cells,ones(size(Sigma_internal_force_C2)))
%scatter(case_3_applied_force, case_3_inline_load_cells,"LineWidth", 1.2)
errorbar(case_3_applied_force,case_3_inline_load_cells,ones(size(Sigma_internal_force_C3)))

plot(case_1_applied_force, fit_I_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])
plot(case_2_applied_force, fit_I_2,"LineWidth", 1.2, "Color", [0.8500 0.3250 0.0980])
plot(case_3_applied_force, fit_I_3,"LineWidth", 1.2, "Color", [0.9290 0.6940 0.1250])

xlabel("Applied Force (N)")
ylabel("Measured Internal Force (N)")
title("Internal Force vs Applied Force")
legend('Case 1', 'Case 2', 'Case 3')
hold off
```

```
figure(5) % Displacement

hold on
scatter(case_1_applied_force, case_1_LVDT,"LineWidth", 1.2)
%errorbar(case_1_applied_force,case_1_LVDT,ones(size(Sigma_displacement_C1)))
scatter(case_2_applied_force, case_2_LVDT,"LineWidth", 1.2)
%errorbar(case_2_applied_force,case_2_LVDT,ones(size(Sigma_displacement_C2)))
scatter(case_3_applied_force, case_3_LVDT,"LineWidth", 1.2)
%errorbar(case_3_applied_force,case_3_LVDT,ones(size(Sigma_displacement_C3)))
% xlim([0 300])
% ylim([-0.2 1.6])

plot(case_1_applied_force, fit_D_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])
plot(case_2_applied_force, fit_D_2,"LineWidth", 1.2, "Color", [0.8500 0.3250 0.0980])
plot(case_3_applied_force, fit_D_3,"LineWidth", 1.2, "Color", [0.9290 0.6940 0.1250])

xlabel("Applied Force (N)")
ylabel("Measured Mid-Point Displacement (mm)")
title("Measured Mid-Point Displacement vs Applied Force")
legend('Case 1', 'Case 2', 'Case 3')
hold off

%% Plots Question 2

P1 = 0:44.48:222.4;
x = 2;
% Reaction Force 1

figure(6)
subplot(3,1,1)
title('Reactions Forces at F0')
hold on
plot(case_1_applied_force, fit_R1_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])
grid on; grid minor
plot(P1,P1/4)
plot(222.4,55.56,'ok')
xlabel('Applied Force [N]')
ylabel('F0 Reaction Force [N]')
hold off
subplot(3,1,2)
title('Reactions Forces at F1')
hold on
plot(case_1_applied_force, fit_R2_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])
grid on; grid minor
plot(P1,P1/4)
plot(222.4,55.63,'ok')
xlabel('Applied Force [N]')
ylabel('F1 Reaction Force [N]')
hold off
subplot(3,1,3)
title('Reactions Forces at F2')
hold on
plot(case_1_applied_force, fit_R3_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])
grid on; grid minor
plot(P1,P1/2)
plot(222.4,111.19,'ok')
xlabel('Applied Force [N]')
ylabel('F2 Reaction Force [N]')
Lgnd = legend('Experimental','Analytical','ANSYS','Location','bestoutside');
```

```
Lgnd.Position(1) = 0.75;
Lgnd.Position(2) = 0.45;
hold off
print('rf comparison','-r300','-dpng')

% Internal Forces

figure(7)
hold on
grid on; grid minor
plot(case_1_applied_force, fit_I_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])
plot(P1,P1)
plot(222.4,450.14,'ok')
xlabel('Applied Load [N]')
ylabel('Displacement [mm]')
title('Comparison of Internal Forces For All 3 Models ')
legend('Experimental','Analytical','ANSYS','Location','northwest')
hold off
print('internal comparison','-r300','-dpng')

% Displacement

figure(8)
hold on
plot(case_1_applied_force, fit_D_1,"LineWidth", 1.2, "Color", [0 0.4470 0.7410])

v_x_case1 = abs((1/(E*I) * (P1/12 * x.^3 - P1.*x)) .* 1000);
grid on; grid minor
plot(P1,v_x_case1)
plot(222.4,1.85,'ok')
xlabel('Applied Load [N]')
ylabel('Displacement [mm]')
title('Comparison of Displacement For All 3 Models ')
legend('Experimental','Analytical','ANSYS','Location','northwest')
hold off
print('displacement comparison','-r300','-dpng')

%% Plots Question 3

% First Case
figure(9)
subplot(2,1,1)
plot(case_2_applied_force(1:60,:), case_2_inline_load_cells(1:60,:),"LineWidth", 1.2)
title("Internal Force vs Applied Load")
xlabel("Applied Load [N]")
ylabel("Internal Force [N]")
grid on; grid minor
subplot(2,1,2)
plot(case_2_applied_force(1:60,:), case_2_LVDT(1:60,:),"LineWidth", 1.2)
title("Displacement vs Applied Load")
xlabel("Applied Load [N]")
ylabel("Displacement [mm]")
grid on; grid minor
print('case 2','-r300','-dpng')
% Second Case
figure(10)
subplot(2,1,1)
plot(case_3_applied_force(1:40,:), case_3_inline_load_cells(1:40,:),"LineWidth", 1.2)
title("Internal Force vs Applied Load")
```

```
xlabel("Applied Load [N]")
grid on; grid minor
ylabel("Internal Force [N]")
subplot(2,1,2)
plot(case_3_applied_force(1:40,:), case_3_LVDT(1:40,:),"LineWidth", 1.2)
title("Displacement vs Applied Load")
xlabel("Applied Load [N]")
ylabel("Displacement [mm]")
grid on; grid minor
print('case 3','-r300','-dpng')
```