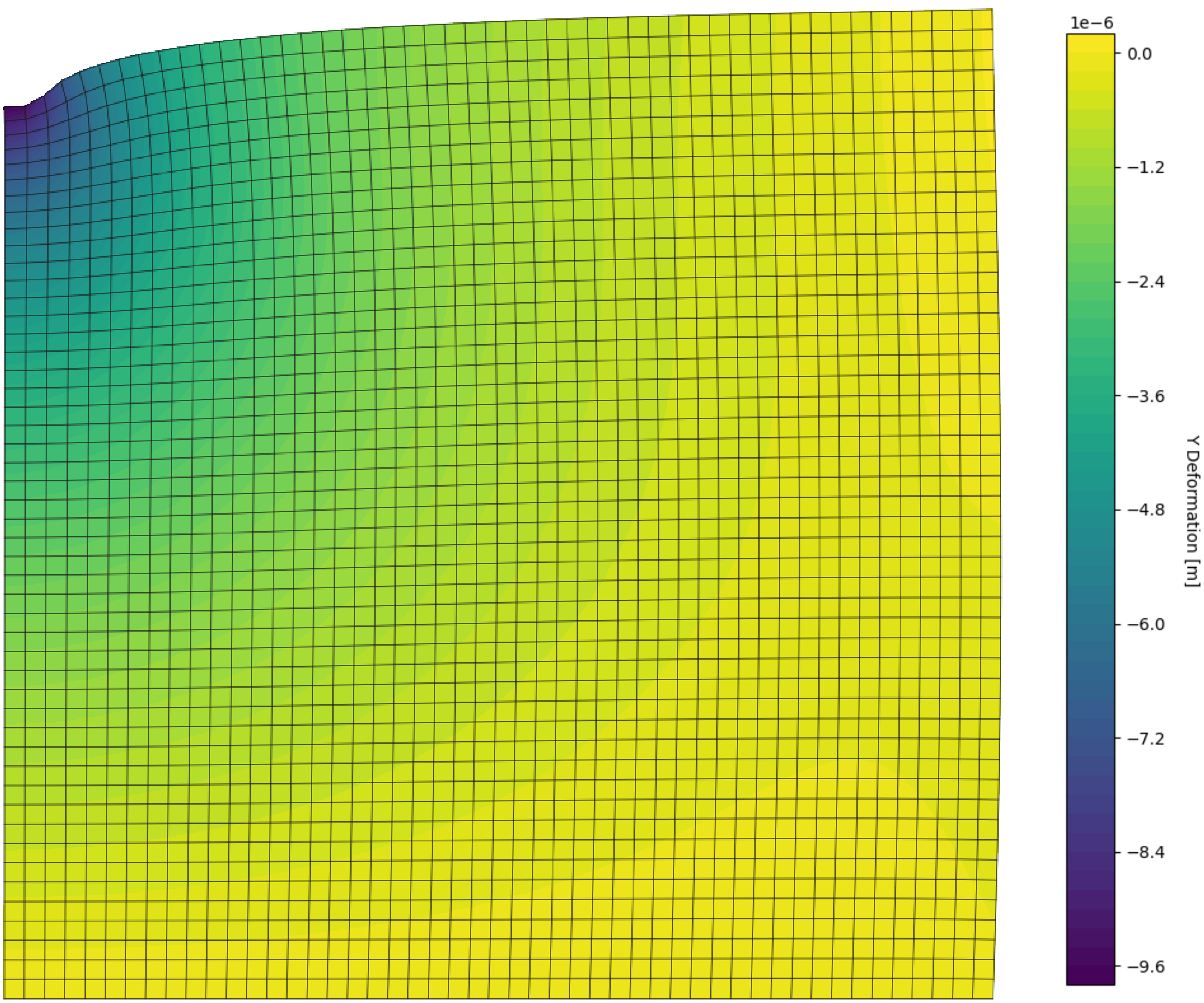
Finite Element Analysis of a Linear Elastic Planar Indentation



Course:	*MECH-462*
Name:	*Thomas Bement*
S/N:	*24099822*
Date:	*Nov 4, 2022*

Table of Contents

1 - Introduction:

This report will focus on the solution of a linear elastic planar indentation problem making use of finite element analysis methods. The problem in question will involve a punch made of an ideal material with infinite rigidity. This punch will provide a means to transfer a punch force to the main material which will deform under the punch force.

This problem will be solved using both Python and Matlab programs which will be validated using Abaqus as well as solid mechanics theory. The purpose of the validation with Abaqus is to obtain a first order validation to ensure the model is providing acceptable values before a final validation with the theory.

2 - Methods:

The analysis of the prescribed displacement punch problem was performed in two different programs. The first was a Matlab code adapted to include prescribed displacements. This code was then ported to Python to take advantage of memory saving techniques for refined meshes. The Python code was run with the same problem conditions as the Matlab code and the stiffness matrices were compared and found to be identical. Following this all analysis was conducted in Python as it was able to run faster due to memory saving methods used. The units being used for this analysis will be the S.I. base units ("SI Units," 2010).

2.1 - Problem Definition:

The problem will be dimensioned using SI units as defined above. To make the problem a representative size it will be dimensioned to be roughly the size of an object that could be tested in a shop scale hydraulic press. The radius (**R**) and height (**H**) of the sample will be taken to be equal while the punch radius (**a**) will be taken to be 20 times smaller than **R**. The dimensions of the sample are summarized below (Figure 1)(Table 1).

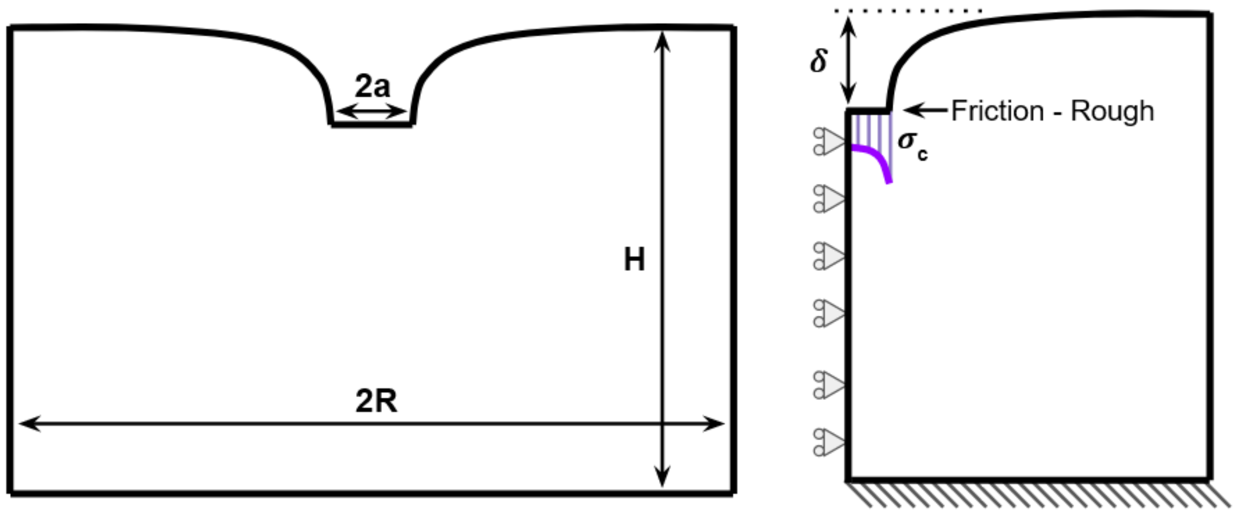


Figure 1. Overall sample geometry as well as symmetric geometry used for this analysis.

Table 1. Summary of parameters used for this analysis.

Name	Value	Units	Name	Value	Units
Elastic Modulus	70E+09	[Pa] [kg · m^{-1} · s^{-2}]	***R***	1.5E-01	[m]
Poisson's Ratio	0.3	[N.a.]	***H***	1.5E-01	[m]
Punch Displacement (δ)	-1.0E-05	[m]	***a***	7.5E-03	[m]

2.2 - Boundary Conditions:

The problem will be simplified to a single body system by replacing the punch with prescribed displacements on the nodes it would normally be contacting. Additionally these nodes will only be allowed to displace in the vertical direction as a no slip (or rough friction) condition will be applied on the punch contact area (Figure 1).

The size of the system will also be reduced by taking a symmetry condition along the center of the punch. This condition will be enforced by only allowing vertical displacement for these nodes ensuring they do not separate from the center line (Figure 1).

Finally a fixed condition will be applied to the base of the sample describing a support below the sample during the compression test with the punch (Figure 1). These boundary conditions are described below in terms of horizontal (**u**) and vertical (**v**) displacements (Table 2).

Table 2. Summary of boundary conditions used in this analysis.

Boundary Condition	u [m]	v [m]
Punch	0	δ
Symmetry	0	Free
Fixed Base	0	0

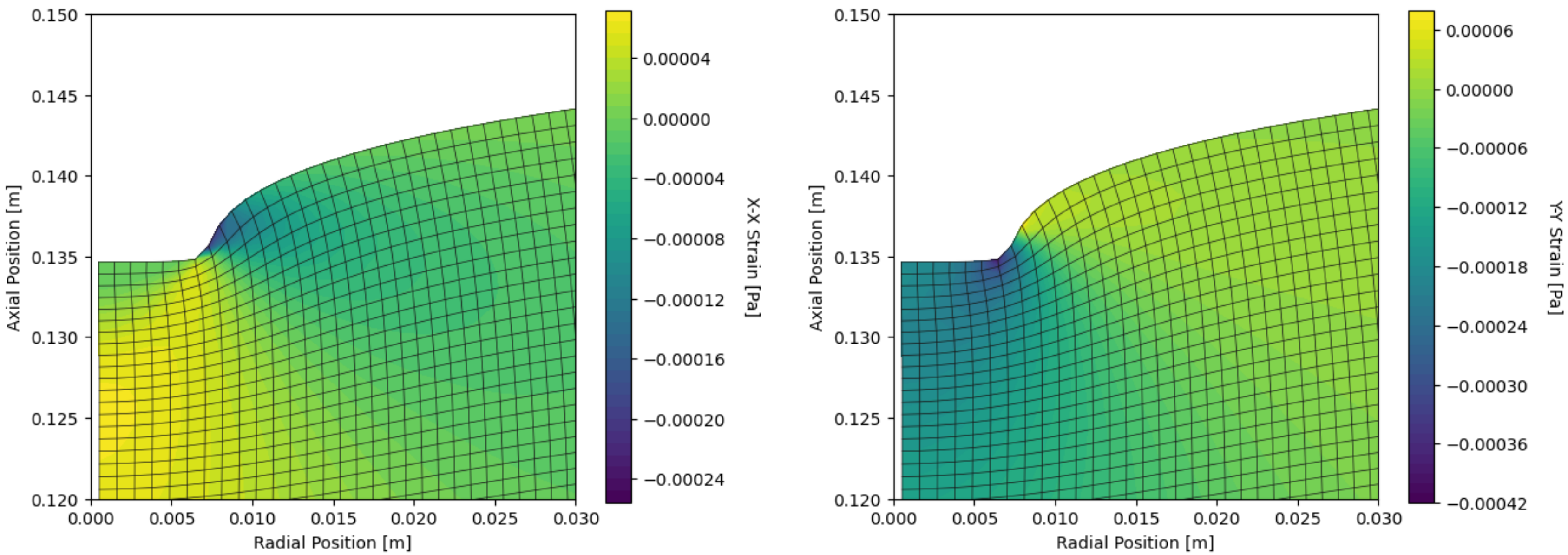
2.3 - Elements and Mesh:

The elements used in this analysis were quadrilateral bilinear elements. Quadrilateral elements were selected because of the rectangular shape of the sample meaning they could easily be meshed in the sample geometry. Bilinear elements were chosen for the first round of analysis due to their simplicity. For the full analysis a total of 150 elements was chosen due to its balance between runtime and refinement around the punch area. The 150 element mesh had a total of 8 nodes on the surface directly contacting the punch area. The number of elements here matters a great deal since linear elements were used so the number of elements in the area around the punch essentially determines the accuracy of the interpolation between element values.

Additional mesh refinement on Abaqus was performed to determine if further refinement of the 150x150 element mesh around the punch location provided any different results.

3 - Results:

By plotting the strain values we observe normal results where the strain at the top left of the figures (Figs. 2.a-2.b) show essentially zero strain in the X-X direction. This is because these nodes are fixed in the X or **u** direction. Directly to the right of these elements we see the peak in the X-X strain where elements to the right of the punch are pulled in and down from the punches prescribed displacement. Indeed this is where the peak is observed for both X-X and Y-Y strain. As such we expect this area to have a very high strain due to Hooke's law, the governing equation behind this FEM analysis.

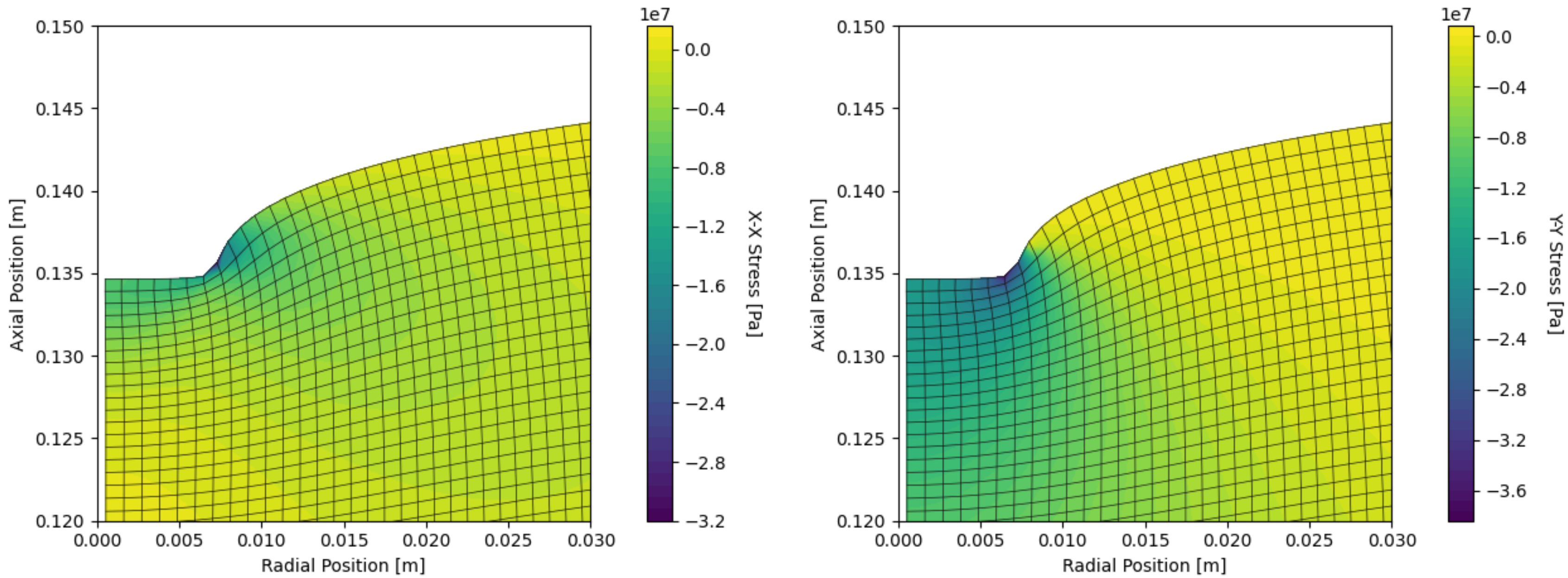


(a) *(b)*

Figure 2. Strain for 150 element mesh with no slip condition on showing X-X (a) and Y-Y (b) strain fields zoomed into the top left area where the punch is acting. Full images are shown in appendix 6.1 (Appendix 6.1).

This is also what is observed in our X-X and Y-Y stress fields showing the relevant area of the sample around the punch (Figs. 3.a-3.b). Unsurprisingly the stress fields share very similar qualitative features as the strain fields. It should be noted that the max stresses observed are on the order of 10^7 Pa while the modulus of elasticity specified for this analysis is on the order of 10^{10} (Table 1). This means that the linear elasticity assumption used in this analysis is well within the acceptable range.

The stress fields show that the X-X stress reaches values close to zero much faster than the Y-Y stress in the axial or Y direction of the sample. Additionally the X-X stress is mostly confined to a small area where the material around the punch was drawn in the radial or X direction. This makes sense as the applied force (result of the prescribed displacement) is applied in the axial direction so most of the stress from this load is described by the Y-Y stress.



(a) (b)

Figure 3. Stress for 150 element mesh with no slip condition on showing X-X (a) and Y-Y (b) stress fields zoomed into the top left area where the punch is acting. Full images are shown in appendix 6.1 (Appendix 6.1).

4 - Validation and Limitations:

4.1 - Validation with Abaqus

Initial validation with Abaqus showed very good quantitative agreement with the displacement and stress values reported by the model used in this analysis (< 15% max error)(Table 3). The setup with Abaqus was identical to what was used in this analysis with the same geometry and material parameters as described in table 1 as well as the same boundary conditions described in table 2 (Figure 4).

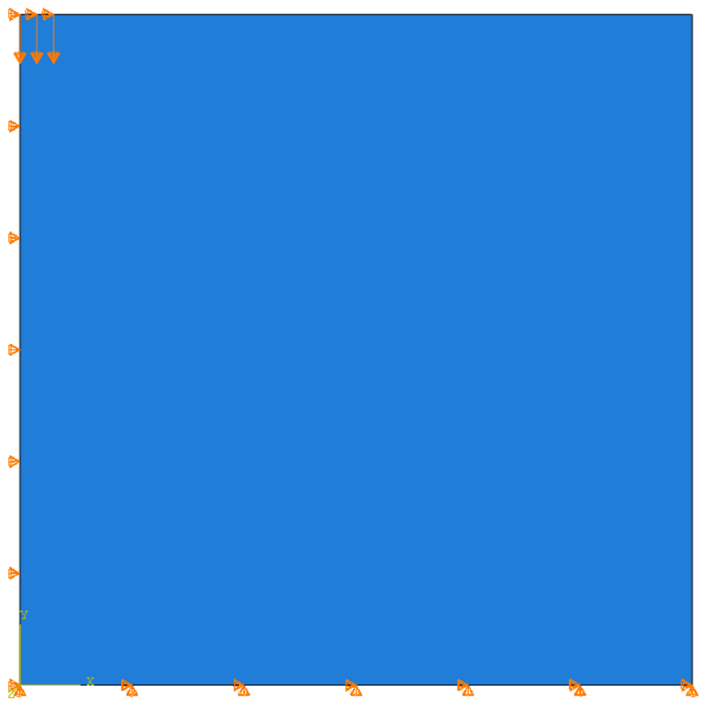


Figure 4. Boundary conditions for Abaqus model.

The Abaqus model was solved using the “Static, General” solver with plane strain linear quadrilateral dominated mesh elements. Sufficient mesh refinement was used to achieve 8 nodes on the punch surface (the same as in the Python analysis) to provide adequate fidelity in the interpolated results.

4.1.1 - Mesh Refinement:

Mesh refinement with Abaqus (Fig. 5) was used to determine if the simple 150x150 element mesh was capturing all the information required to solve this problem. By using more elements around the punch area better interpolation was achieved, however only a 6% difference was observed between the 150x150 mesh and refined mesh results from Abaqus. As such this added fidelity wasn't deemed necessary for the Python analysis being done.

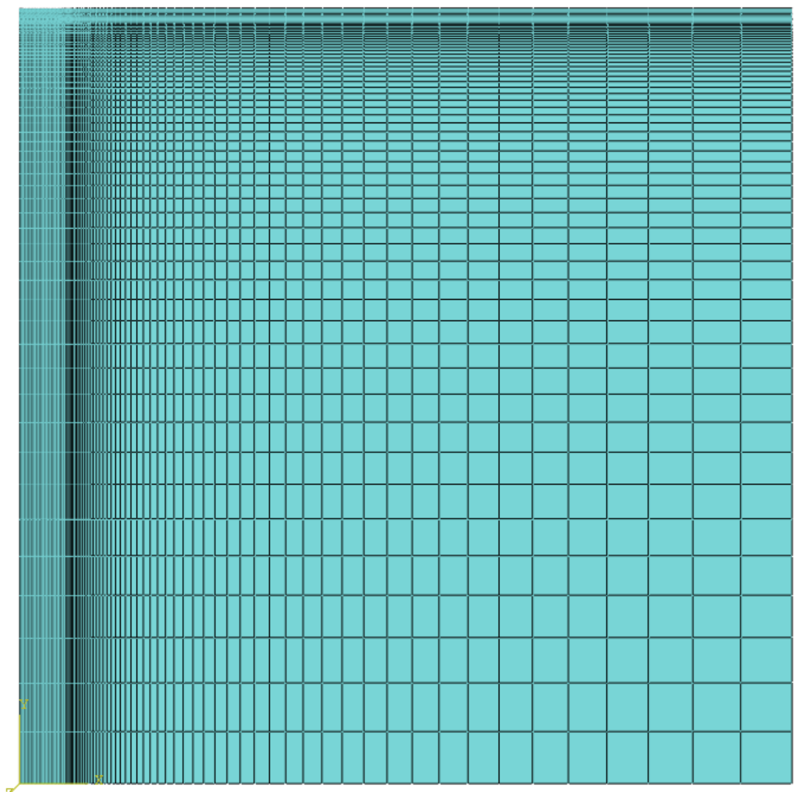


Figure 5. Refined mesh biased toward the punch location (top left).

4.2 - Validation with Theory

Validation with theory allows us to compare the displacement (Eqs. 1-2) and pressure (Eq. 3) curves for the theoretical equivalent of the problem. The constants F_2 and D_2 were found by ensuring the vertical displacement was equal to δ inside the punch radius and the vertical displacement was zero at the outer edge of the sample (**R**).The most common solution to this kind of problem makes use of complex stress functions (Bower, 2009; *EN224: LINEAR ELASTICITY*, n.d.; Johnson, 1985). This solution makes the assumption that the punch is lubricated or frictionless. Due to this there is a difference in the boundary conditions between the problem being solved in this analysis and the validating equations used. This difference can be attributed to some of the difference between theoretical results and the initial computationally calculated results (Table 3).

$v(r) = \frac{-F_2(1-\upsilon)}{\pi G} log(a) + d_2 \, for \, r < a$	*Equation 1*
$v(r) = \frac{-F_2(1-\upsilon)}{\pi G} log(r + \sqrt{r^2 - a^2}) + d_2 \, for \, r > a$	*Equation 2*
$P(r) = \frac{F_2}{\pi\sqrt{a^2-r^2}} \, for \, r < a \, where \, F_2 \, is \, found \, to \, give \, v(r) = \delta \, for \, r < a$	*Equation 3*

Table 3. Validation data between Python model and Abaqus/theoretical results.

Validation Parameter	Abaqus Error		Theoretical Error	
	Max [%]	Average [%]	Max [%]	Average [%]
Vertical Displacement	13.38	3.91	34.17	24.73
Vertical Stress	12.15	9.47	34.92	24.88

The error with Abaqus was quite small and was mainly attributed to differences in the mesh. The errors listed above were obtained using the refined Abaqus mesh which showed the largest difference due to mismatched positions in the comparison. Exact mesh matching yielded smaller max error for Abaqus with 10 and 6 % for max displacement and stress error respectively (~2% average error for both parameters).

The error with the theory was around double that of the Abaqus validation error. As mentioned above the difference is likely due to the fact that the frictionless punch allows points to move and slip out from under the punch. To try and obtain more accurate and representative validation a different theoretical set of equations was used. These equations were obtained from the solution to the flat punch problem where the punch is perfectly bonded to the sample material (no slip)(Adams, 2016). The solution is not quite as straightforward and requires some integrals to be computed to obtain the required results (Appendix 6.2).

Taking the normal condition where **T** is zero the theoretical results for no slip were compared to the frictionless punch results (Bower, 2009) by varying values of **P** to achieve the desired δ. Finally the difference between the two stress values was obtained to determine the effect the no slip condition had on the Python and Abaqus analysis. The two curves for Y-Y stress matched very closely but slowly diverged towards the corner of the punch (Fig. 6). This indicates that the main effect of the no slip condition is on the points that are forced to remain fixed at the corner of the punch. Overall this error was relatively small with a maximum value of 20% towards the corner of the punch.

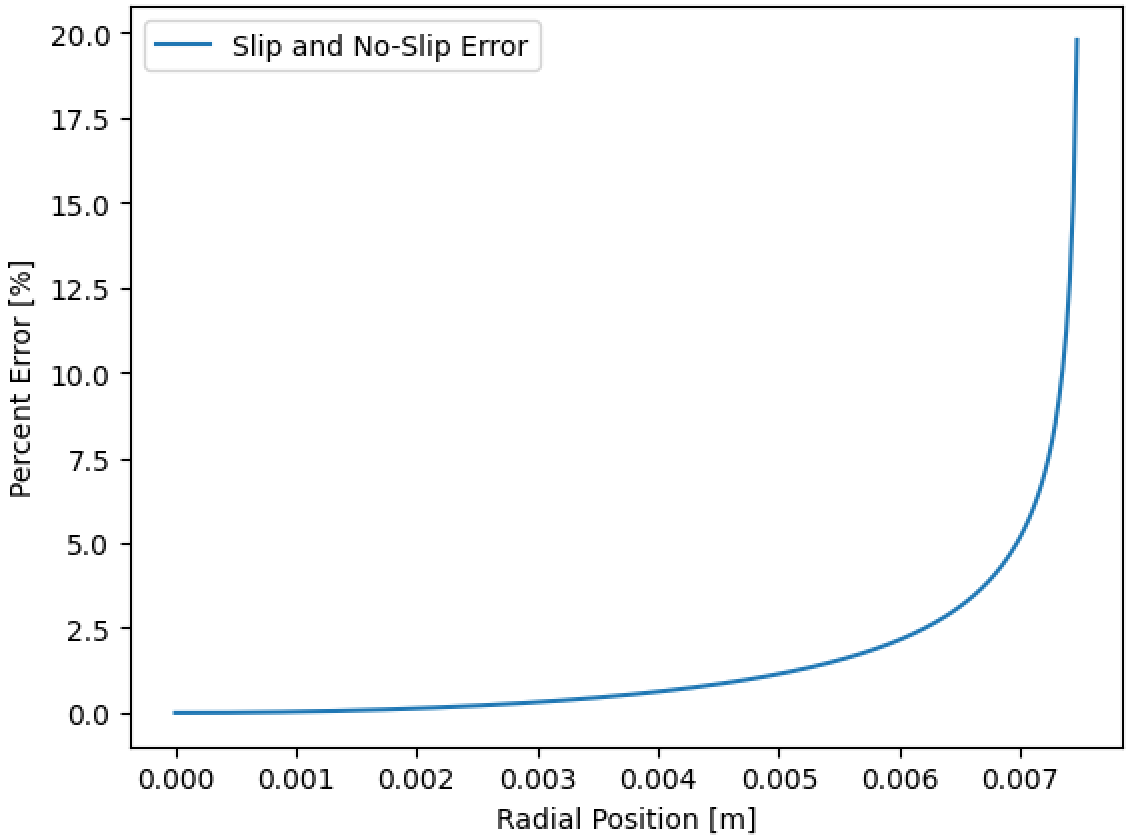


Figure 6. Percent error between slip and no slip theory equations for Y-Y stress.

4.3 - Overall Validation

The validation methods explored yielded promising results. The Abaqus validation matched the closest with the findings found from analysis with the Python based FEM code (Figs. 7-8). The main difference between the Python and Abaqus results are the methods used to calculate the values as well as the location the stress is evaluated at. For the Python code the stress is evaluated at the centroid while in Abaqus it is evaluated on the top surface. Despite this the error is quite acceptable (~13% max error) indicating the Python code is solving the problem correctly. Validation with theory was trickier since there are multiple differences in the way the problem is solved when using theoretical equations. In general the validation was quite good with maximum errors of ~30%, 20% of which can be partially attributed to the slip versus no slip condition. The rest of the error is likely due to the imperfect refinement of the mesh used (~6% error) as well as the difference in centroid areas for the stress comparison.

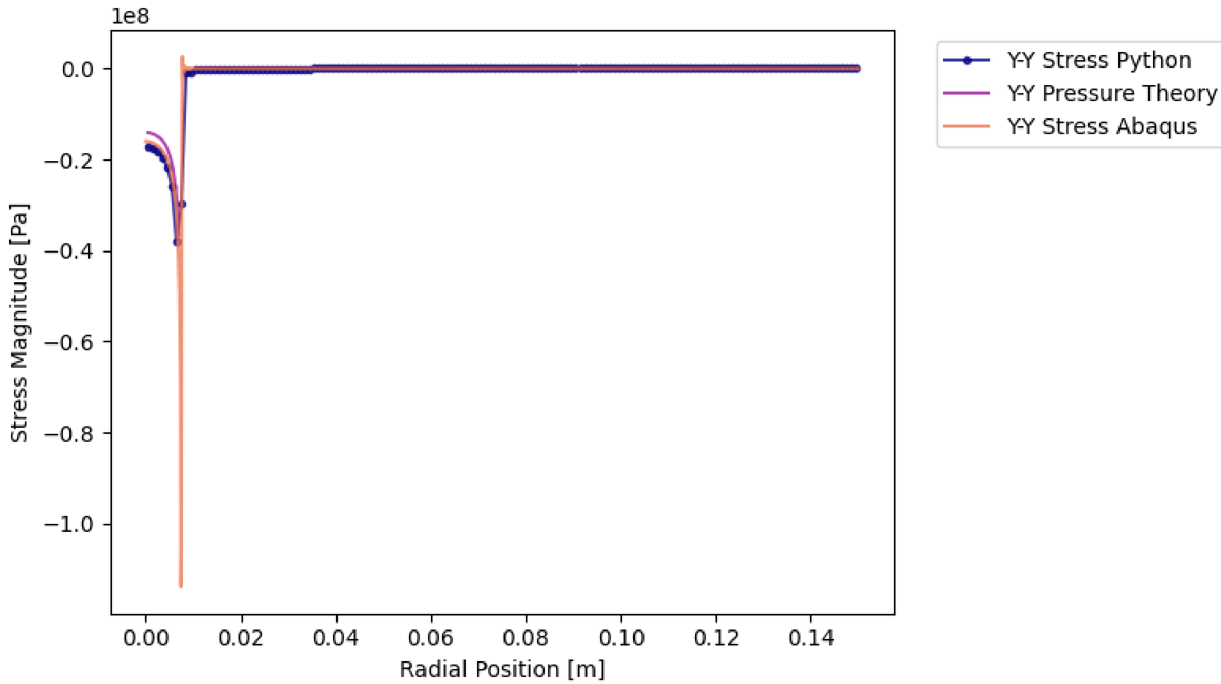


Figure 7. Comparison of the Y-Y stress distribution along the top elements for the Python based FEM analysis as well as the two validation methods.

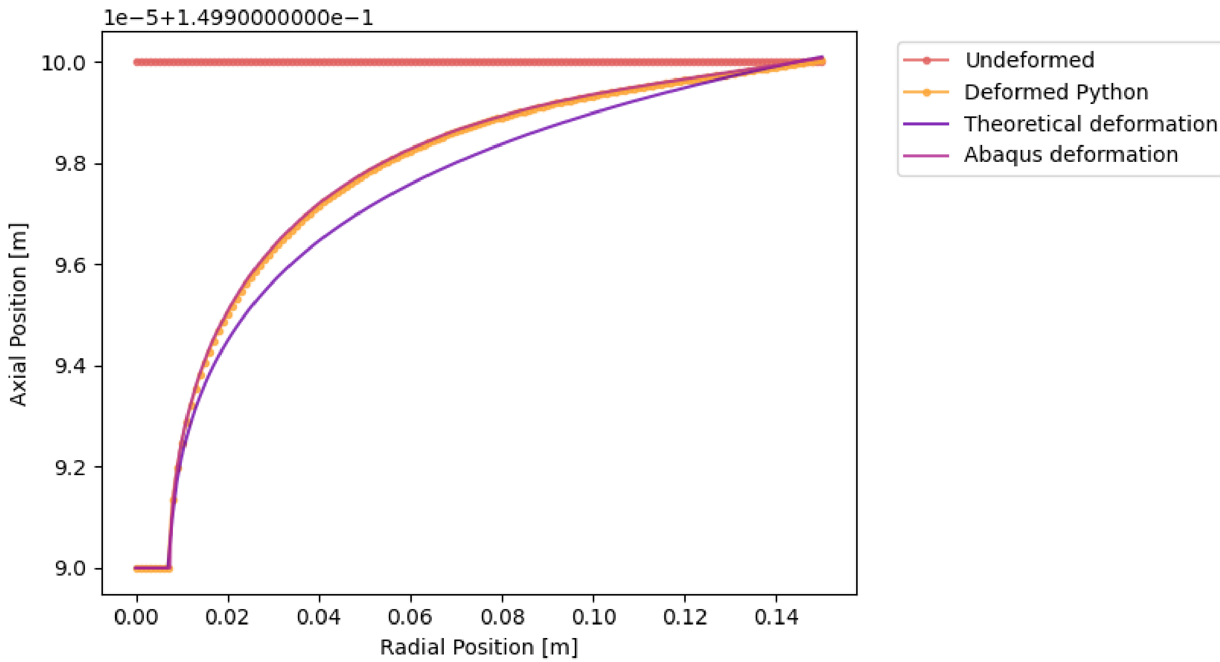


Figure 8. Full field view of the deformed shape of the Python based FEM analysis. Overlaid on the plot are the two validation method curves for the deformation of the top of the sample surface. All plotted deformations have the same scale factor applied.

5 - References:

Adams, G. G. (2016). Frictional slip of a rigid punch on an elastic half-plane. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *472*(2191), 20160352. https://doi.org/10.1098/rspa.2016.0352

Bower, A. F. (2009). *Applied Mechanics of Solids*. CRC Press. https://doi.org/10.1201/9781439802489

EN224: LINEAR ELASTICITY. (n.d.). Retrieved November 25, 2022, from https://www.brown.edu/Departments/Engineering/Courses/EN224/axicontact/axicontact.html

Johnson, K. L. (1985). *Contact Mechanics*. Cambridge University Press. https://doi.org/10.1017/CBO9781139171731

SI Units. (2010). *NIST*. https://www.nist.gov/pml/owm/metric-si/si-units

6 - Appendices:

6.1 - Supplemental Images

Additional images have been generated using a mesh of 50x50 elements for clearer visualization of the mesh. All images can be downloaded from the following Git repository: **LINK**

6.2 - Source Code

Python and Matlab source code can be downloaded from the following Git repository: **LINK**

6.3 - Abaqus Model

The Abaqus model used for validation can downloaded from the following Git repository: **LINK**

6.4 - Perfectly Bonded Punch Analysis

The analysis of the perfectly bonded punch analysis (Adams, 2016) is much more complex than the complex stress analysis performed for the frictionless punch. The initial equation for this method are the integral form of the displacement components ($\overline{u}$, $\overline{v}$),

$\frac{d\overline{u}}{dx} =- \frac{B}{4\pi}\overline{p}(x) - \frac{A}{4\pi}\int_{-a}^{a} \frac{\overline{q}(\xi)}{x-\xi}d\xi$	*Equation A.1*
$\frac{d\overline{v}}{dx} = \frac{A}{4\pi}\int_{-a}^{a} \frac{\overline{p}(\xi)}{x-\xi}d\xi - \frac{B}{4\pi}\overline{q}(x)$	*Equation A.2*

where A and B are constants (Eqs. A.3-A.4) and $\overline{p}$ and $\overline{q}$ are the normal and shear tractions respectively.

$A = 4\frac{(1-\upsilon)}{G}$	*Equation A.3*	$B = 2\frac{(1-2\upsilon)}{G}$	*Equation A.4*

These equations are solved using a complex function allowing the calculation of $\overline{p}$ and $\overline{q}$ (Eqs. A.5-A.6).

$\overline{p}(x) = \frac{\lvert C\rvert}{\sqrt{a^2-x^2}}cos(\varepsilon \cdot ln(\frac{a+x}{z-x}) + \varphi)$	*Equation A.5*
$\overline{q}(x) = \frac{\lvert C\rvert}{\sqrt{a^2-x^2}}sin(\varepsilon \cdot ln(\frac{a+x}{z-x}) + \varphi)$	*Equation A.6*
$C = \lvert C\rvert e^{i\varphi}\ where\ \lvert C\rvert = \frac{cosh(\pi\varepsilon)}{\pi}\sqrt{P^2 + T^2}$	*Equation A.7*
$\varphi = tan^{-1}(\frac{T}{P})\ for\ \frac{-\pi}{2} < \varphi < \frac{\pi}{2}$	*Equation A.8*