

Project Number: ME-CF-NP08

INVERSE METHOD OF NANOINDENTATION BY LASER INTERFEROMETRY

A Major Qualifying Project Report

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

in Mechanical Engineering

by

Anthony DiOrio

Russell Morin

Eric Wilusz

Date: 10 March, 2008

Keywords:

1. nanoindentation
2. inverse method
3. laser interferometry

Professor Cosme Furlong

Abstract

As miniaturization technology has progressed there has been an increased need for characterization of physical properties at the micro- and nano-scales. Nanoindentation has become a process which can be used to determine the physical properties of engineering materials. It is characterized by the continuous measurement of the force applied to and resulting displacement of the sample during the indentation event. A commercial apparatus capable of performing these tests, however, costs upwards of \$100,000, and is beyond the budget of many colleges and universities. Thus, a low-cost module using a unique inverse method of nanoindentation was developed in this project. Using this method, the force and displacement data to construct the force-displacement curve are obtained indirectly from the interferometer images, rather than being measured directly. An apparatus capable of proving this method was designed, analyzed, and constructed. Lastly, a series of indentation events were performed, validating both the mechanism and the inverse approach to nanoindentation.

Acknowledgements

We would like to acknowledge Professor Cosme Furlong for all his efforts and support that made this project possible. We would also like to thank him for pushing us to produce a great project. Without his persistent questions like “how does it work?” or “why does it work?”, we would have completed the project without truly understanding the subject matter involved.

The members of this group would further like to acknowledge and offer our appreciation to the secretaries in the Worcester Polytechnic Institute Mechanical Engineering Department, Dr. Gretar Tryggvason, head of the WPI Department of Mechanical Engineering, the staff of the WPI Electrical and Computer Engineering Shop, and to the graduate and post-graduate students working in the Center for Holographic Studies and Laser micro-mechaTronics. We would also extend our thanks to the staff of the Washburn Shops of the Worcester Polytechnic Institute, and to Mr. Boquan Li, the manager of the Materials Characterization Labs. We spent many hours on the telephone with the sales and support personnel at Thorlabs, Veeco, National Aperture, Nano and More, and Pacific Nanotechnologies, and we would like to thank their respective staffs for helping us troubleshoot the components we purchased. Without their help and support, this project would not have been nearly as successful, if it were possible at all.

Table of Contents

Abstract	i
Acknowledgements	ii
Table of Contents	iii
List of Figures	v
List of Tables	vii
List of Nomenclature	viii
Executive Summary	xi
1. Introduction.....	1
2. Computational and Experimental Approach to Nanoindentation	3
2.1. Obtaining Deflection Information with Laser Interferometry	4
2.1.1. Laser Interferometry	5
2.1.2. Application of Laser Interferometry for Nanoindentation.....	7
2.1.2.1. Unwrapping the Phase	9
2.1.2.2. Extracting Height Information from the Unwrapped Phase	9
2.1.3. Determining the Necessary Sampling Frequency.....	10
2.1.4. Determining the Optimum Number of Elements.....	12
2.2. Solving an Inverse Beam Deflection Problem.....	14
2.2.1. Cantilever Beam Deflection Equations.....	14
2.2.2. A Finite Element Approach to Solving an Inverse Problem.....	17
2.2.3. Mathematics behind the ANSYS Inverse Solution.....	18
2.2.4. Extracting Force and Displacement Data from ANSYS Solution	20
2.3. Validation of Algorithm with Comparison to Theoretical Results	21
2.3.1. Creation of Simulated Indentation Data	21
2.3.2. Extraction of Force and Displacement Information from Dataset.....	24
2.3.3. Comparison of Algorithm Data to Theoretical Data.....	24
3. Indentation Approaches	25
3.1. Previous Designs.....	25
3.1.1. Macroscale Cantilever Concept	26
3.1.2. Voice Coil Actuation Concept	28
3.1.3. Hydraulic Column and Pressure Transducer Concept	29
3.1.4. AFM Cantilever and Laser Interferometer	31
4. Implementation	35
4.1. Final Design of the Nanoindenter	37
4.2. Construction and Setup of Nanoindenter.....	42
4.2.1. Laser Interferometer.....	43
4.2.2. Piezo: Open Loop vs. Closed Loop	43
4.2.3. Cantilever Beam.....	45
4.2.4. Control Strategy	45
4.2.5. Sample Preparation	46
4.3. Indentation Procedure	47
5. Procedure for Data Analysis	65
6. Mechanics of Indentation.....	82
6.1. Elastic Regime	84
6.2. Elastic-Plastic Regime	95

6.3. Plastic Regime	97
6.4. Non Spherical Indenter Geometries	99
6.5. Mathematical Modeling of Indentation and Retraction	101
7. Results	103
8. Conclusions	108
9. Recommendations	109
10. Future Work	110
11. References	114
12. Appendices	117
12.1. Appendix A – Derivation of Hertzian Contact Formulae	117
12.2. Appendix B: Component-Level Research	134
12.2.1. Piezoelectric Actuation	134
12.2.2. Linear Variable Differential Transformers	136
12.2.3. Strain Gauges	139
12.2.4. Parallel Plate Capacitive Displacement Sensing	141
12.2.5. Commercially Available Nanoindenters	142
12.2.5.1. CSIRO	142
12.2.5.2. MTS Nano Instruments	142
12.2.5.3. Micro Photonics Inc.	142
12.2.5.4. Asylum Research	143
12.2.5.5. Other Companies	143
12.2.6. Tip Characterization for Nanoindenters	143
12.2.6.1. Berkovich Tip	144
12.2.6.2. 90-Degree tips	144
12.2.6.3. Conospherical tips	145
12.2.6.4. Specialty Tips	146
12.3. Appendix C – Python Scripts	147
12.3.1. IndentationImageProcessor. py	147
12.3.2. ANSYSMacroCreator. py	160
12.3.3. ConverttoCSV. py	163
12.3.4. RPolyFit. py	164
12.3.5. MQPModel. py	165
12.4. Appendix D – Drawings of Manufactured Components	173
12.5. Appendix E – Attached Files	180

List of Figures

Fig. 0-1. Analytical indentation solution with experimental indentation data.....	xv
Fig. 2-1. Flow chart of the indentation process.	3
Fig. 2-2. A schematic of how a laser interferometer is configured (National Instruments, 2008). 6	
Fig. 2-3. Free body diagram of the cantilever directly after contact with the sample.	15
Fig. 2-4. Python code to add beam and material properties.....	21
Fig. 2-5. Python code to create the nodes and elements for the ANSYS script.....	23
Fig. 2-6. Comparison to theoretical force displacement data.	24
Fig. 3-1. Macroscale Cantilever Concept	27
Fig. 3-2. Voice Coil Actuation Concept	28
Fig. 3-3. Hydraulic Column and Pressure Transducer Concept	30
Fig. 3-4. Three dimensional model of the piezo actuated AFM cantilever concept.	31
Fig. 4-1. Picture of completed module, mounted and ready to use	36
Fig. 4-2. A side view of the module that is mounted underneath the laser interferometer.	36
Fig. 4-3. Final configuration of the nanoindenter.	37
Fig. 4-4. Model of the interferometer and the goniometer.	37
Fig. 4-5. Piezo enclosure model.....	38
Fig. 4-6. Three axis translation stage component of the nanoindenter.	38
Fig. 4-7. Recessed area in the cantilever with the probe holder.	39
Fig. 4-8. Laser interferometer in the CHSLT.	42
Fig. 4-9. Live view of LaserView with the cantilever brought to the top of the motorized z-stage.	47
Fig. 4-10. The HyperTerminal showing the z-stage retracted to top of the z-axis.	48
Fig. 4-11. APT software used to zero the strain gauge reader for the closed loop piezo (Thorlabs).....	48
Fig. 4-12. The settings menu which shows the closed loop tab selected to put the piezo into closed loop mode. Note: The SMA Cable must be connected from the strain gauge reader to the piezo driver.	49
Fig. 4-13. The Piezo is now zeroed in closed loop mode and also the two APT modules are active as shown in the software.	50
Fig. 4-14. Close contact mode probe clip holder (Pacific Nanotechnologies, 2008).	50
Fig. 4-15. Special tweezers used to insert AFM probes in to Probe Clip Holder (TDI International, Inc. , 2008).....	51
Fig. 4-16. Placing the AFM probe into the Probe Clip Holder and using the group's own Probe Clip Holder opening device.	52
Fig. 4-17. The sample in which we are going to indent which has a surface roughness of 0. 050 microns.....	53
Fig. 4-18. The module ready for the insertion of the sample as indicated by the arrow.	54
Fig. 4-19. Probe brought into focus and before adjustments were conducted with goniometer..	55
Fig. 4-20. Probe now in focus and goniometers adjusted to have only one fringe, the ideal situation before indenting.	56
Fig. 4-21. The probe actuated down from the limit stop closer to the sample using the motorized z-axis.	57
Fig. 4-22. The probe brought into contact with the sample using the motorized z-stage. Smaller steps with the motorized stage were used as the probe got closer to the sample.....	58

Fig. 4-23. Shifter voltage enabled and set to 5.52 volts; Also indicated in the image window...	59
Fig. 4-24. In the View Mode, the optical phase + 180 is select to see the wrapped phase.....	60
Fig. 4-25. Enable the reference and take a new reference. The image window will also indicate that a new reference was taken.	60
Fig. 4-26. Before starting indentation and capturing pictures, indicate the directory to store the images and file name.	61
Fig. 4-27. The cantilever is in contact and actuation has begun.....	62
Fig. 4-28. This picture shows the indentation process. As the piezo is actuated the fringes increase.	63
Fig. 4-29. Micrometers used to move the piezo, cantilever, and probe to different positions to indent.....	64
Fig. 5-1. Downloading Python version 2. 5 (Python).....	65
Fig. 5-2. Download screen for MatPlotLib (Matplotlib, 2008).	66
Fig. 5-3. Download page for NumPy (NumPy, 2008).....	67
Fig. 5-4. Finite element program, ANSYS, used for analysis of image data, (ANSYS, 2008)...	68
Fig. 5-5. Pasting the analysis files into the image directory.	69
Fig. 5-6. Creation of CSV file with piezo displacement data.	70
Fig. 5-7. Obtaining the beam rectangle from MS Paint using cursor position.	71
Fig. 5-8: Editing the beam rectangle in the program.	72
Fig. 5-9. Initial screen for image processing program.	72
Fig. 5-10. Entering initial image number to the processor.	73
Fig. 5-11. Prompt to enter the base filename into the processor.....	74
Fig. 5-12. Prompt for the piezo displacement data filename.	75
Fig. 5-13. All prompts answered before beginning image processing.	75
Fig. 5-14. ANSYS log files created by the image processor.	76
Fig. 5-15. Prompt for base filename in ANSYSMacroCreator. py.....	77
Fig. 5-16. Prompts for number of files to be processed by ANSYS.....	78
Fig. 5-17. Click File->Read Input From.	79
Fig. 5-18. Open the log files to begin the ANSYS processing.	80
Fig. 5-19. Program to convert to “*.csv”.	81
Fig. 6-1. Cross section of an indentation tip.	90
Fig. 6-2. The expanding cavity model (Fischer-Cripps, 2005).....	96
Fig. 6-3. Summary of the Elastic, Elastic-Plastic, and Plastic Regimes (Fischer-Cripps, 2005)	99
Fig. 6-4. Theoretical loading and unloading functions for indentation plot from MathCAD. ..	102
Fig. 7-1. Unwrapped cantilever contour during indentation.....	103
Fig. 7-2. Unwrapped cantilever contour during retraction.	104
Fig. 7-3. Experimental indentation and retraction data with analytical data.	105
Fig. 7-4. Three indentations created in aluminum sample with the nanoindenter developed as part of this project.	106
Fig. 7-5. Two indentations created with the nanoindenter developed as part of this project. ...	107
Fig. 12-1. Cross section of an indentation tip.	127
Fig. 12-2. A schematic of a strain gauge.	140

List of Tables

Table 2-1. Convergence to Expected Force with Increased Number of Elements	13
Table 4-1. Summary of nanoindenter components.	41
Table 4-2. Commands for controlling the motorized Z axis.....	57
Table 12-1. Statistics for a Berkovich Tip.....	144
Table 12-2. Statistics for a 90-Degree Tip.....	145
Table 12-3. Data for Conospherical Tips.....	146

List of Nomenclature

AFM – atomic force microscope.

ANSYS – a program used to solve finite element problems.

APT User – a program used to control the piezo driver and strain gauge reader.

Atomic force microscope probe – a probe consisting of a base silicon wafer which has a silicon cantilever beam of specified cross-section, natural frequency, stiffness, and length which is grown on the base wafer. At the free end of the cantilever, a tip of specified geometry is grown on the cantilever; for this experiment, it is this tip which was brought into contact with the sample and performed the indentation.

Berkovich Tip – commercially available nanoindenter tip.

CSV – comma separated values.

Fringes – patterned variations between black and white in the images acquired with the interferometer.

Hyperterminal – a serial communication program.

Inverse method of nanoindentation – method of indentation where the force-displacement curve is constructed using data obtained from images. Most commercial nanoindenters utilize force and displacement sensors to determine the forces applied to and resulting deformation of the sample, respectively. Using the inverse method, however, the indentation is performed, and the images obtained using the laser interferometer are then post-processed, yielding the force and displacement data. Using this method, each image captured is representative of one point on the force-displacement curve.

Indentation – term used to identify the first phase of the indentation event where the force applied on the indenter tip is increasing, and the penetration depth the tip travels into the sample

is increasing. Indentation technically stops when the indenter tip has reached the maximum force or displacement for the experiment. The term indentation is also frequently used to describe the whole process, however, including the phase called retraction.

Knoop Tip - commercially available nanoindenter tip.

LabView – a program from National Instruments used to automate and control various laboratory equipment.

Laser interferometry – a principle by which a laser beam is emitted, then split, with a portion of the beam reflecting off a mirror at a known distance, and the remaining portion reflecting off the sample being measured. The reflected beams are then recombined, and the distance to the sample can be calculated using the phase shift between the reflected beams.

LaserView – a program written by Ellery Harrington to control a laser interferometer and acquire images.

MathCAD – a sophisticated mathematical simulation program

Nanoindentation – a process by which material properties of a sample can be determined.

Nanoindentation is characterized by the continuous measurement of force and displacement, permitting the construction of a force-displacement curve. Traditional macroscale indentation, by contrast, is often characterized by the measurement of the resulting indentation, resulting in a single data point, rather than a series of data points gathered over the entire indentation event.

Using the standard set by National Institute of Standards and Technology, nanoindentation experiments must have a total range of less than one micrometer.

Nanoscale – anything measured in nanometers.

National Aperture – a company that makes the translation stages used to construct the nanoindenter.

Piezo – an actuator that makes use of the piezoelectric effect. It increases in length proportional to an applied voltage. Also referenced as piezo actuator and piezoelectric actuator.

ProEngineer – a computer aided modeling program. Also referenced as ProE.

Python – a scripting language used to process the data obtained in this project.

Retraction – term used to identify the second phase of the indentation event where the indenter tip is being retracted from the maximum force or displacement limit during the indentation event.

During the retraction phase of the indentation event, the force and displacement are monitored, as with the indentation phase; the measurements taken during retraction provide information as to the elastic restorative forces present within the sample, as the retraction data for many materials creates a hysteresis loop when plotted on the same set of axes as the indentation data.

T-Cubes – common name of the controllers used to control the piezo.

Vibration Isolation Table – a pneumatic table that isolates its contents from surrounding vibrations.

Vickers Tip - commercially available nanoindenter tip.

Executive Summary

As the state of technology and modern materials science continues to progress, scientists and materials researchers now have the capability to examine materials on the atomic scale. Whereas previous tests for various properties were completed by the application of a large force and measuring the resultant macroscale displacement, testing at the nanoscale permits the same properties to be observed by applying very small loads and measuring the resulting nanoscale deformation. This permits non-destructive testing of not only metallic samples, but also permits tests to be performed on biological materials.

Attention is then turned to an apparatus that is capable of achieving these measurements with an acceptable level of precision. Several commercial nanoindentation machines are available, with each having slight differences from competing models. The commonality that they all share, however, is that the displacement of the indenter tip and the force that is applied to the indenter tip at the specified displacement are monitored continuously, providing the information necessary to create a force-displacement curve from which properties such as elastic modulus, Poisson's Ratio, and hardness can be determined for metallic samples. Previous hardness tests, by contrast, are characterized by the use of a constant load, often on the order of hundreds of pounds of magnitudes, and the measurement of the resulting deformation. Only hardness data can be obtained from this system of measurement, however, as it yields only two data points – the surface of the sample before the test is performed, and the resulting deformation after the test. Nanoindentation also yields a measure of the elastic restorative force within a particular sample, as many common engineering materials exhibit a hysteresis loop. This phenomena is observed when retracting the indenter tip from its maximum depth in the sample in a controlled manner, resulting in a second set of data.

To obtain the data necessary to determine the material properties mentioned above, a unique inverse method was used. Whereas many of the commercial nanoindentation devices use sensors to directly measure the force and displacement being applied on the sample at any given instant, this method requires images taken using a laser interferometer. From these images, the force and displacement data can be extracted and analyzed to produce the same force-displacement curve that could be produced using direct measurements. The inverse methodology was selected over the previous direct concepts for several reasons, among which are measurable resolution, the number of components needed to achieve the indentation, and the cost of the apparatus.

Thus, once the approach to nanoindentation had been determined, work commenced to synthesize a design for an indenter apparatus that would have the capabilities of the expensive commercial models while costing under \$5,000. To begin, a survey of available components was undertaken, including available products to control the fine motion of the indenter, as well as to measure force and displacement. From this initial research, a concept was synthesized around a macroscale cantilever that held the indenter tip. For this concept, the fine motion of the assembly was controlled by a piezoactuator, while force was measured by an array of strain gauges, and the displacement was measured by an inductive sensor. A second concept featured an indenter actuated by a voice coil, where the voice coil, the sample, and the indenter tip were mounted to three separate plates connected by folded, leaf-type springs. The third concept utilized the force-multiplying properties of a piston-cylinder assembly utilizing incompressible fluid, where the force would be measured by a pressure transducer, and the displacement would be measured by the closed-loop piezoactuator.

Each of the designs described above can be characterized by the use of mechanical or electrical sensors to directly output force and displacement. For the final concept, which ultimately became the apparatus that was analyzed and constructed, a different approach was taken. For this design, cantilever probes designed for use with atomic force microscopes are used. The probes are mounted in a specialized clip which is magnetically held in a larger macroscale cantilever. The macroscale cantilever is attached to a piezoactuator, which provides fine motion during the indentation event. An enclosure made of aluminum ensures the proper positioning and alignment of the piezo, and also provides a point of attachment to the stages which permit coarse motion. Coarse motion is provided by both motorized and manually-adjusted stages, allowing the vertical movement of the entire indentation assembly for the insertion of samples, as well as lateral motion, permitting repeated indentations without having to manually move the sample. Since the work space beneath the interferometer is limited, the entire assembly is mounted on a large cantilever, which moves the manually-adjusted components closer to the operator while simultaneously positioning the probe beneath the interferometer beam. Beneath the module itself, various stages and goniometers permit optimal positioning beneath the interferometer assembly.

To determine the forces and displacements the probe undergoes, images from a laser interferometer were used. Using the principle of laser interferometry, a series of images can be captured and analyzed to recover force and displacement data from the interferometric images using an inverse solution. Once the images have been captured, they are input to a finite-element-analysis program, where each image yields a point on the force-displacement curve. Once the experimental data has been analyzed, the forces and displacements measured experimentally can be compared with the results from theoretical analysis.

Using a derivation based primarily on the work of Robert Hooke and Heinrich Hertz, a series of equations can be used to predict theoretical forces and displacements that the sample will experience during the indentation process. Depending on the geometry of the tip that is used to perform the indentation, the resulting deformation can be characterized by a series of equations corresponding to three regimes: the elastic, elastic-plastic, and fully plastic regime. Much of the work that has been done in this field to date is for spherical geometries, which are used when testing brittle samples so as to not induce instant fracture. Much of the theory has been modified, however, to be usable for tips of other geometries, including pyramidal, as was used in this experiment.

Once the experimental data had been collected and analyzed, the force-displacement curve based on the experimental data was compared to the theoretical curve. Comparison of the curves shows that this method is a valid approach to indentation; both curves exhibit the same trends and predict forces and displacements of the same magnitude.

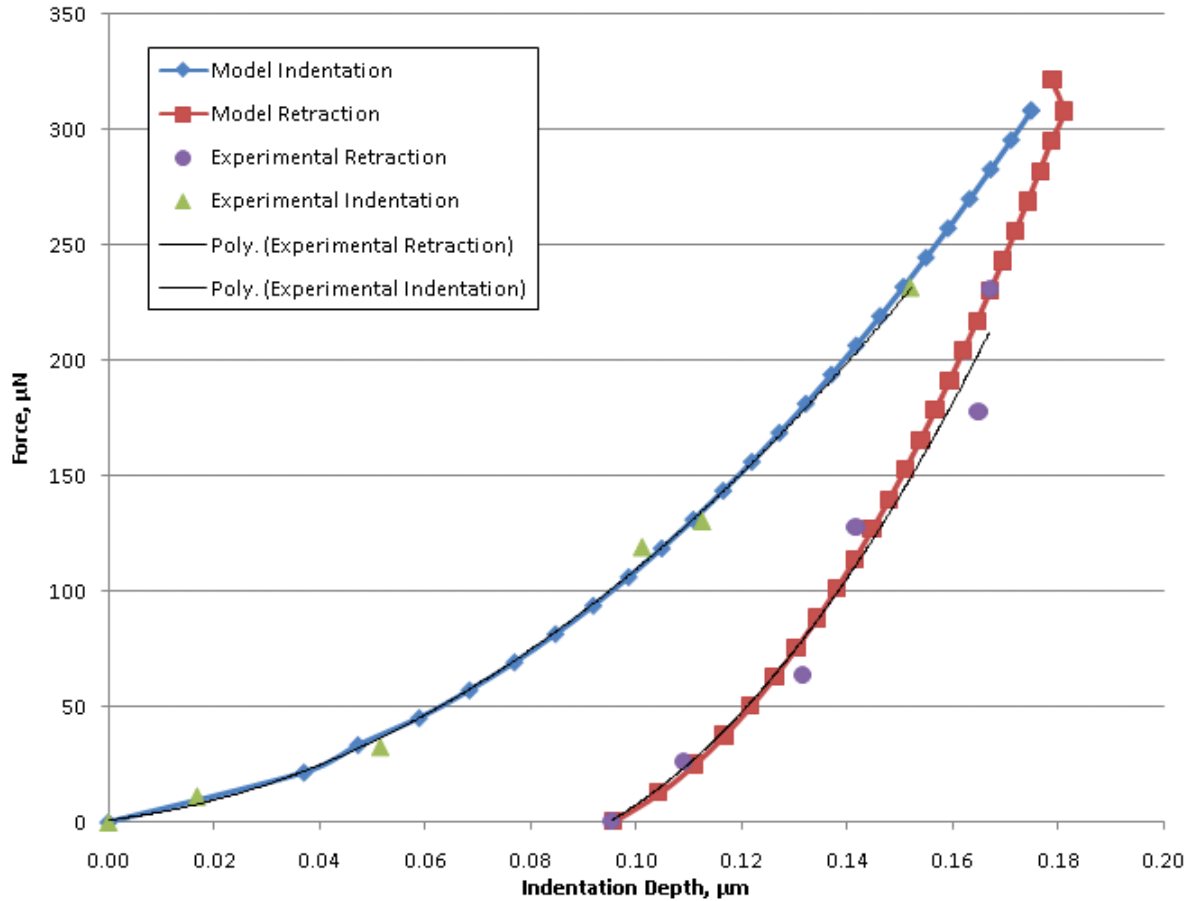


Fig. -. Analytical indentation solution with experimental indentation data.

Thus, the goals set for this experiment were met. The module used to perform these experiments was designed, analyzed, and constructed for less than the target cost of \$5,000, and the method used to obtain the force-displacement curve was also validated. Despite these accomplishments, there are some areas of the project which could use further improvement, and other areas of the project that could be explored further. Further automation of the indentation process would result in more accurate data, as the indentation could be performed as an uninterrupted, single-speed process where data is collected continuously, rather than the manner in which the validation tests were performed. Automation of the data collection and analysis process would allow the indentation events to be completed in a shorter amount of time, as the data could be processed nearly real-time, rather than having a lengthy post-process for data

acquisition. Also, if the probes with different mechanical properties, such as length, cross-sectional dimensions, and stiffness were used in the module, a wider range of materials could be indented.

1. Introduction

As technology has progressed in recent years, it has become more important to be able to manipulate things and make measurements on an increasingly smaller scale. Specifically, indentation technology has progressed from large scale indentation to nano-scale indentation. Nanoindenters can be used for numerous applications, including data storage, nano-hardness testing, and nano-scratching. They can also be used to ascertain properties of a material, including modulus of elasticity and Poisson's Ratio.

Of course, as the scale of indentation technology has become smaller, the cost of this technology has become progressively larger. A basic, commercially available nanoindenter costs a minimum of \$100,000. Additional accessories to increase the versatility of the machine can bring prices to \$400,000 and beyond. As a result, a very limited number of commercial and educational institutions have access to a nanoindenter, and those who don't need to make use of nearby facilities if they wish to perform indentation experiments. Worcester Polytechnic Institute's Center for Holographic Studies and Laser micro-mechaTronics is one facility that has nanoindentation capability, utilizing an indenter manufactured by the Australian firm CSIRO. The machine has a very large footprint for the task that it performs, and it is currently not functioning. Due to the age of the machine, parts are scarce, and the benefit from repairing such a machine is low compared to the cost. The cost of replacing the existing unit with a new indenter is also prohibitive.

This paper describes the development of a nanoindentation module to be used with a laser interferometer. The total cost of the module is less than \$5,000, significantly less than a commercially purchased machine, and the module will allow many more labs to have nanoindentation capabilities. The setup is small, lightweight, and can be easily removed and transported. Because the system was designed to be a module, it can be taken off of the

interferometer when not in use and therefore does not require a large amount of dedicated lab space nor an interferometer devoted entirely to nanoindentation.

The uniqueness to the design and operation of this module comes from the manner in which the force and displacement data are obtained. Whereas the commercial models described previously use force and displacement sensors to ensure the accuracy of these measurements, the approach taken in this exercise was an inverse approach. Through the use of the principles of laser interferometry, the force and displacement data needed to construct a force-displacement curve can be extracted from a series of images showing interferometric fringes. Thus, all of the data gathered using this approach is gotten indirectly, as the images must be processed following the experiment to determine the force and displacement acting on the sample at any particular moment during the indentation event.

To prove the validity of this approach, a theoretical model was created based on the theory of contact mechanics. Then, a series of designs for the module were created, taking into account the resolutions, degrees of freedom, and constraints inherent in both the modules and the workspace in which the module would have to function. Following the design, analysis, and construction of the module, a series of indentation experiments were conducted, and the data was analyzed using the aforementioned inverse method. After extracting of the force and displacement data from the interferometric images obtained during the experiments, this data was compared to the theoretical model of indentation.

2. Computational and Experimental Approach to Nanoindentation

As previously stated, aside from sub-micrometer magnitudes, nanoindentation is characterized by measuring force and displacement simultaneously during the indentation process. Ideally, this data would be measured and processed in real time; however, it would require extensive computing power. Instead, an indirect solution involving post-processing was

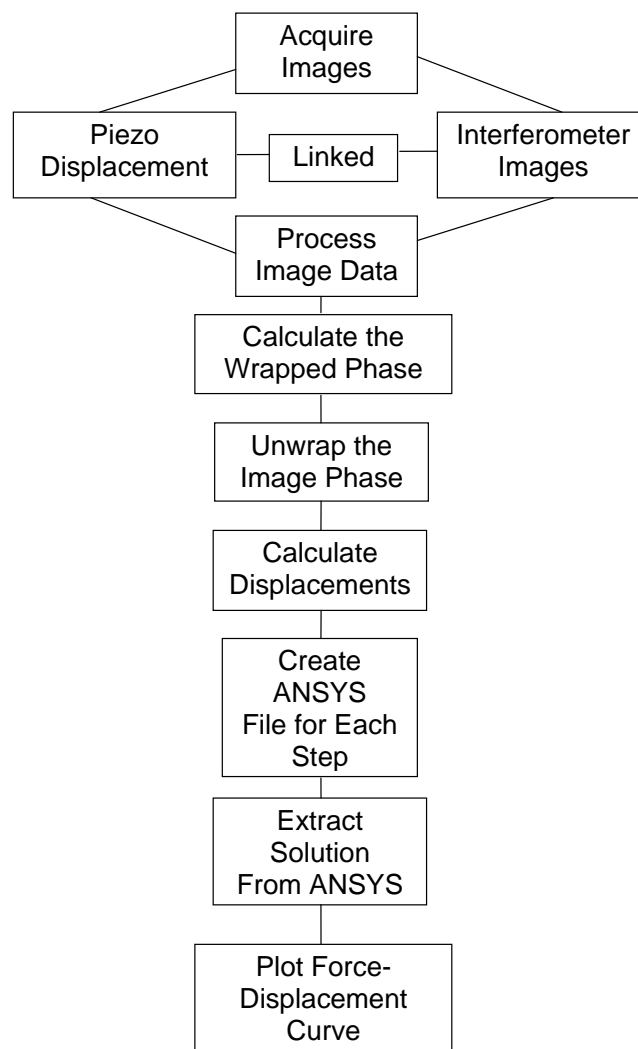


Fig. -. Flow chart of the indentation process.

used to obtain these measurements. A flow chart describing the process used to complete an entire indentation can be seen in Fig. -. A laser interferometer is used to record a series of images of a loading and unloading cycle. Each image is associated with a specific position of the piezoelectric actuator that is used to move the AFM probe into the sample. For every position, the interferometer takes four images which are phase shifted by ninety degrees. Using these images, the wrapped phase is calculated for the given piezo position. When the phase is unwrapped it can be used to acquire information about the relative heights of the surface that has been imaged. In this case, it yields the contour of the deflected AFM cantilever. The contour information, along with the associated position of the piezo actuator, is processed by a program written using Python, a scripting language with syntax similar to C++. The Python program uses the data to create script files that can be read by ANSYS, a robust finite element analysis software package. When the ANSYS script file is read, it solves an inverse problem to determine the force and displacement at the probe tip. This force and displacement become the coordinates of one point on the loading-unloading curve, and the steps are repeated for each height step of the actuator.

2.1. Obtaining Deflection Information with Laser Interferometry

In order to perform true nanoindentation, both force and indentation depth during the loading and unloading of the sample must be measured. If the indentations were to be done on a larger scale, these parameters could be measured with more conventional methods, such as a dial drop gauge and a force gauge. None of the available conventional measuring instruments are capable of measuring forces of the magnitude expected during nano-scale indentation. While

some conventional methods explored were able to measure the indentation distances expected, it was decided to be more beneficial to use a system that could measure both force and displacement together rather than trying to use two separate sensing units. The force and displacement measurements for the nanoindenter constructed are obtained using images taken with a laser interferometer.

2.1.1. Laser Interferometry

Interferometry is a way of making measurements that cannot be measured directly. Interferometry gets its name because it makes use of interfering beams of light in order to obtain measurements. One of the unique qualities of light is that it acts as both a particle and a wave. The light emitted from a laser in an interferometer can be tuned to give off light of a specific wavelength. Wavelengths of light can be over an extremely wide range; however laser light is usually in the visible range of the spectrum between 400 and 700 nanometers. The other important property of light that allows for laser interferometry to be possible is the fact that light travels at a constant speed, roughly 300 million meters per second.

Interferometry makes use of a known reference distance to determine the distance that is being measured. Laser light exits the laser traveling in one coherent beam at the fixed wavelength. The first obstacle the beam encounters is a beam splitter which breaks the laser light into two beams at a right angle with each other. One of the beams passes straight through the splitter to the known reference mirror. The other is bent 90 degrees towards the sample being measured. Both the beams are reflected back towards the beam splitter and recombine to form the light that is detected by the camera.

Making use of the knowledge that light has a finite wavelength and a constant speed, it follows that if the two beams of light have traveled the same distance, they will have traveled the same number of wavelengths. If the distances that the two split beams have traveled are not exactly the same, however, one of the beams will have more waves than the other. When the beams recombine, the waves no longer match up as they did when they were split from the initial coherent beam. Depending on the difference of the sample distance from the reference distance, the waves in the newly combined beam of light will be out of phase with each other and can cause interference. These interferences produce a fringe pattern on the image collected by the digital camera. The fringes change in color between black and white based on how much interference there is between the beams. The fringes are analogous to contour lines on a map that are used to

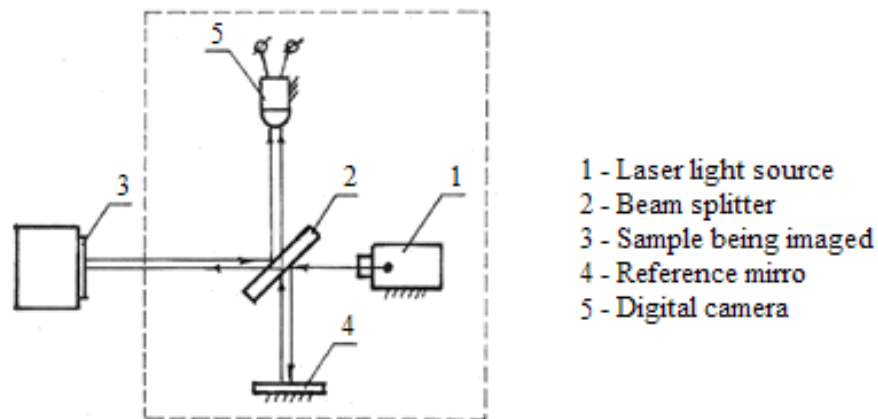


Fig. -. A schematic of how a laser interferometer is configured (National Instruments, 2008).

show differences in elevation. Working with the images captured with the interferometer, the distance from the beam splitter to the sample can be compared to that of the reference distance. Because the image taken by the camera is several pixels wide, the topology of the surface can be obtained using the individual pixel data.

2.1.2. Application of Laser Interferometry for Nanoindentation

For the purposes of this nanoindenter, laser interferometry will be used to measure the deflections in a cantilever beam AFM probe and the force between the probe and the sample being indented. A schematic the configuration of a typical laser interferometer can be seen in . In the laboratory setup where this indenter is installed, the detector is a digital camera with a resolution of 800 pixels by 800 pixels. The physical size of each pixel on the charge-coupled device, the electronic component used to acquire the images, within the camera is 6.25 micrometers. The beam passes through an objective lens that magnifies the light by a factor of 4. Other objective lenses could be used, however it would require the interferometer to be recalibrated. With the 4x lens, each pixel on the images obtained with the camera represents data from a 1.5625 micrometer square area on the sample based on the equation below.

$$Width = \frac{Physical\ Size}{Magnification} = \frac{6.25\ \mu m}{4} = 1.5625\ \mu m \quad (.)$$

With the area described by that each pixel in the image known, the maximum viewing area available with the given setup can be determined. To find this, the following equation can be used.

$$Image\ Width = Pixel\ Width * Pixels = 1.5625\ \mu m * 800 = 1250\ \mu m \quad (.)$$

The cantilevers used for the indenter are only 225 micrometers long, so the entire cantilever can be seen within a single image taken by the interferometer.

Due to the interference in the light reaching the digital camera, the images obtained have a series of fringes that range in color from black to white. The vertical distance from one black fringe to the next is determined by the following equation where the wavelength lambda is 630 nm.

$$FringeDist = \frac{\lambda}{2} = \frac{630nm}{2} = 315nm \quad (.)$$

Once the desired sample to be measured with the interferometer is setup, it must be brought into focus in order to see the fringes. Focusing is done by carefully adjusting the distance from the sample to the beam splitter and the angle of the sample. These are adjusted using micrometer dials on the z stage under the indenter and the two goniometers that allow rotation about the x and y axes.

The interferometer is controlled by a program called LaserView, which was written by Ellery Harrington, a graduate student helping Professor Furlong with his research. This program has several unique features that simplify obtaining images from an interferometer. One feature that is crucial is known as Reference mode. With this mode enabled, a piezo actuator connected to the interferometer's reference mirror cycles through a series of four positions. An image is acquired at each of these positions. The reason it takes four images is that in each image that light is phase-shifted by 90 degrees relative to the previous image. The light intensity data from these four images is combined to determine the relative heights of the surface that was imaged. The equations below are used to obtain the wrapped phase which can be unwrapped in order to determine the contours of the surface.

$$\Phi = \arctan \left[\frac{I_4 - I_2}{I_1 - I_3} \right] \quad (.)$$

$$I_N = I_O \left[1 + \frac{1}{2} \cos \left(\frac{N\pi}{2} \right) \right] \quad (.)$$

In the above equations, I_O is the intensity of a pixel from the image I_N . The second part of the second equation corrects for the phase shift between the images so they can be combined. The wrapped phase is calculated for each of the pixels in the area of interest of the images so the

height information can be extracted. Within Reference mode, LaserView performs these calculations internally and outputs images that represent the wrapped phase.

2.1.2.1. Unwrapping the Phase

The range of the wrapped phase data obtained using the above equations is shown in the equation below.

$$\frac{-\pi}{2} < \text{WrappedPhaseValues} < \frac{\pi}{2} \quad (.)$$

This is the range of the function because it is governed by the range of the arctangent function. For the purposes of this project, the phase does not need to be unwrapped for the entire image, because only a line along the length of the cantilever beam is of interest. The unwrapping of the wrapped phase data is performed using a Python script. The script iterates through the wrapped phase data along the line of interest. Each time the program encounters a peak in the phase data, it increments an offset value, by pi which is equal to one period of the wrapped phase function. Every point in the unwrapped phase is created by adding the wrapped phase value to the current offset value.

2.1.2.2. Extracting Height Information from the Unwrapped Phase

With the unwrapped data calculated, only simple calculation is necessary to obtain the associated displacement data. The equation below allows you to calculate the height term L_z using the unwrapped phase where lambda is the wavelength of the laser light.

$$L_z = \Phi \frac{\lambda}{2\pi} \frac{1}{2} \quad (.)$$

When this equation is applied to all the points along the line of interest, the points can then be plotted to show a profile of the surface along that line.

2.1.3. Determining the Necessary Sampling Frequency

Since the measurements are being done on such a small scale, it is important that it is known that the measurements are being done with an appropriate frequency to completely resolve what is being measured. If there was a large amount of variation in height within each pixel, it would be hard to determine the actual height. To find the minimum frequency that must be used when sampling the data in order to obtain a complete picture of the beam contour, the Nyquist Criterion is used. For this experiment, the frequency is associated with how frequently you select a point from the beam deflection curve to model the beam deflection in ANSYS. The Nyquist Criterion states that the frequency at which data is sampled must be at least twice the maximum frequency in the data being measured as expressed in the equation below.

$$f_s > 2f_{\max} \quad (.)$$

To find out if the images being taken satisfied the Nyquist Criterion, the wrapped phase data gathered from LaserView needed to be analyzed. Since the wrapped phase is a periodic function, frequencies can be determined from the data. To begin with, the distances between all the peaks on a sample wrapped phase curve were measured. Each of these measurements is the period of the function at that given section. The local frequency of the function for each period

was then determined by calculating the reciprocal of each of the periods. The equation for frequency as a function of the period T is

$$f = \frac{1}{T} \quad (.)$$

The maximum frequency seen in the sample wrapped phase data was

$$f_{\max \text{ observed}} = 0.2133 \frac{1}{\mu\text{m}} . \quad (.)$$

Based on the Nyquist Criterion, the sampling frequency must be at least twice that, or

$$f_{\text{sampleminneeded}} = 0.4266 \frac{1}{\mu\text{m}} . \quad (.)$$

The actual sampling frequency of the digital camera at full resolution was found by taking the reciprocal of the distance between pixels, 1.5625 μm . This maximum possible sampling frequency is

$$f_{\text{samplemax}} = \frac{1}{1.5625 \mu\text{m}} = 0.64 \frac{1}{\mu\text{m}} . \quad (.)$$

Since

$$f_{\text{samplemax}} > f_{\text{sampleminneeded}} , \quad (.)$$

the Nyquist Criterion is satisfied and the images from the camera are acceptable for measuring the surface contours seen in this experiment.

Working backwards from the Nyquist Criterion, the maximum possible frequency that can be measured was determined from the sampling rate. The frequency measured must be less than one half the sampling frequency.

$$f_{\text{measured}} < \frac{f_{\text{sampling}}}{2} = \frac{0.64}{2} \frac{1}{\mu\text{m}} = 0.32 \frac{1}{\mu\text{m}} \quad (.)$$

This means that the maximum frequency measurable in the data is

$$f_{\max measurable} = 0.32 \frac{1}{\mu m} . \quad (.)$$

By finding the reciprocal of the maximum measurable frequency, the minimum period measurable is found.

$$T_{\min measurable} = \frac{1}{f_{\max measurable}} = \frac{\mu m}{0.32} = 3.125 \mu m \quad (.)$$

This minimum measurable period corresponds to a function period just two pixels wide. For this to occur, the image would need to change from black to white to every other pixel.

2.1.4. Determining the Optimum Number of Elements

When using a finite element approach to solve a problem, the number of elements used will affect the solution obtained. The finite element method is merely an approximation of the actual solution. If too few elements are used, the solution could be far from the true value. As a general rule, the finite element solution will converge upon the actual solution as the number of elements used increases. Logically, it would then make sense to use as many elements as possible within the limits of the available computing power. However, not all situations will continue to converge with an increasing number of elements. Finite element models with a large number of elements are prone to round-off errors, caused by the computer rounding or truncating values too long or large for it to handle, which can actually make the solution obtained diverge from the desired solution.

To ensure that the data obtained from ANSYS is the most accurate, a sensitivity analysis was performed on the solution based on changing the number of elements. The analysis was run

Table -. Convergence to Expected Force with Increased Number of Elements

Number of Elements	Force (microNewtons)	Percentage Error	Nominal Force
1	0. 0000000	100. 000000%	0. 052799769
2	0. 0791997	50. 000000%	0. 052799769
3	0. 0422398	20. 000000%	0. 052799769
4	0. 0565712	7. 142857%	0. 052799769
6	0. 0532059	0. 769231%	0. 052799769
8	0. 0528386	0. 073638%	0. 052799769
9	0. 0527880	0. 022197%	0. 052799769
12	0. 0528001	0. 000570%	0. 052799769
16	0. 0527998	0. 000004%	0. 052799769
18	0. 0527998	0. 000001%	0. 052799769
24	0. 0527998	0. 000005%	0. 052799769
36	0. 0527998	0. 000018%	0. 052799769
48	0. 0527998	0. 000061%	0. 052799769
72	0. 0527998	0. 000064%	0. 052799769
144	0. 0527995	0. 000532%	0. 052799769

for a deflection of 1.23 nanometers (the minimum measurable deflection), 315 nanometers (the deflection that would add one fringe to the beam), and a force of 100 microNewtons. This way ensured that the determined number of elements would be valid for all possible instances. It was determined that the maximum number of elements that could make up the beam is

$$N_{\max} = \frac{\text{BeamLength}}{\text{PixelWidth}} = \frac{225 \mu\text{m}}{1.5625 \mu\text{m}} = 144 \quad (.)$$

The number of elements for each step of the analysis corresponded to all whole number factors of 144.

For each element configuration, an ANSYS script file was created with the corresponding number of elements. The only difference between the script files is the number of elements. ANSYS was used to solve for the force at the end of the beam for each of these scripts. After all the solutions were obtained from ANSYS for each configuration, they were compared with the theoretical solution for a beam undergoing that load or deflection. The comparison involved finding the percentage error from the actual value and showing how it converges towards zero.

For all three cases, the percentage error quickly approached zero. The equation use to find the percentage error is as follows:

$$\%Error = \frac{(AcceptedValue - ExperimentalValue)}{AcceptedValue} (100\%) \quad (.)$$

An example of this convergence can be seen in Table -. However, around 18 or 24 elements, the percentage error began to increase again. Based on these analyses, it was decided to use 18 elements to model the deflection of the beam in ANSYS. Numbers greater than this, while only a few millionths of a percentage error different, would increase the error too much on a nano scale to be acceptable. One benefit of using fewer elements, aside from increased accuracy of the solution, is that the computer will process it faster with a fewer number of elements.

2.2. Solving an Inverse Beam Deflection Problem

In order to find the force between the AFM probe and the sample being indented, an inverse approach must be used to solve the problem. With this approach, the deflections that exist in the beam after the deflection has occurred are measured, and the forces that caused those deflections are determined. This requires knowledge of how a beam deflects under given loading conditions. Use of finite element methods is also necessary to solve this problem because the equations are complex and would take a large amount of time to solve manually.

2.2.1. Cantilever Beam Deflection Equations

With traditional cantilever beam theory problems, input loading conditions are given, and the object is to find the displacement profile along the length of the beam. For this to hold true,

the boundary conditions of a cantilever are assumed to be that the beam deflection is zero at its fixed end and that the slope of the beam profile is also zero at that point.

Equations called singularity functions are used to specify the loading on a beam and determine the shear force, moment, slope, and deflection along its length. In defining the singularity functions for a beam, a free body diagram of the beam, seen in Fig. -, is needed to identify all the loading conditions acting upon it and their locations. From this free body diagram, the balance of forces needed for the cantilever to be in static equilibrium can be solved. The static equilibrium equations are found by summing the forces in the x and y directions and the moments.

$$\begin{aligned} \Sigma F_x &= 0 \\ \Sigma F_y &= F_{wy} + F_{sy} \\ \Sigma M &= M_w + LF_{sy} \end{aligned} \quad (.)$$

For static equilibrium to exist, each of these equations must be equal to zero. By setting each of them equal to zero, the following simplifications can be made to solve for the force and moment reactions at the base of the beam. For these simplifications and further equations, the force at the tip of the beam will be denoted as F.

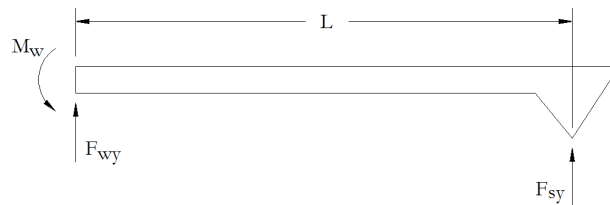


Fig. -. Free body diagram of the cantilever directly after contact with the sample.

$$\begin{aligned}
F_{sy} &= F \\
F_{wy} + F_{sy} &= 0 \\
F_{wy} &= -F \\
M_w + LF_{sy} &= 0 \\
M_w &= -LF
\end{aligned}
\tag{.}$$

Using the information from the free body diagram and the force balance equations, the loading singularity function can be developed, as seen in the equation below.

$$q(x) = M_w \langle x-0 \rangle^{-2} + F_{wy} \langle x-0 \rangle^{-1} + F_{sy} \langle x-l_e \rangle^{-1} \tag{.}$$

By integrating this singularity function twice, the functions for the shear force and moment along the beam can be obtained. Further integration, with the material and geometric properties of the beam E , elastic modulus, and I , cross sectional area moment of inertia, taken into consideration, yields the singularity functions for the slope and deflection along the beam. Once all of the singularity functions are determined, there were four constants of integration that can be solved for using the boundary conditions mentioned above. The final singularity functions after solving for the constants of integration solved are as follows:

$$\begin{aligned}
q(x) &= M_w \langle x-0 \rangle^{-2} + F_{wy} \langle x-0 \rangle^{-1} + F_{sy} \langle x-l_e \rangle^{-1} \\
V(x) &= M_w \langle x-0 \rangle^{-1} + F_{wy} \langle x-0 \rangle^0 + F_{sy} \langle x-l_e \rangle^0 \\
M(x) &= M_w \langle x-0 \rangle^0 + F_{wy} \langle x-0 \rangle^1 + F_{sy} \langle x-l_e \rangle^1 - M_w - F_{wy}(l_e) \\
\theta(x) &= \frac{1}{EI} \left(M_w \langle x-0 \rangle^1 + \frac{F_{wy}}{2} \langle x-0 \rangle^2 + \frac{F_{sy}}{2} \langle x-l_e \rangle^2 - M_w x - F_{wy}(l_e)x \right) \\
y(x) &= \frac{1}{EI} \left(\frac{M_w}{2} \langle x-0 \rangle^2 + \frac{F_{wy}}{6} \langle x-0 \rangle^3 + \frac{F_{sy}}{6} \langle x-l_e \rangle^3 - \frac{M_w}{2} x^2 - \frac{F_{wy}}{2} (l_e)x^2 \right)
\end{aligned}
\tag{.}$$

Based on the boundary conditions, the singularity functions that define the loading of the beam, and the static equilibrium equations, the singularity function for beam deflection can be simplified to the following form.

$$y(x) = \frac{-Fx^2(3L-x)}{6EI} \quad (.)$$

In this equation F is the force between the probe and the sample, L is the length of the beam, E is the beam's elastic modulus, I is the cross sectional area moment of inertia, and x is the position along the beam. For the purposes of extraction the indentation results, an equation for the deflection at the end of the beam is necessary. To find one, the length L is substituted into the above equation for x to yield the equation for the end deflection (Norton, 2006).

$$y(x) = \frac{-FL^3}{2EI} \quad (.)$$

2.2.2. A Finite Element Approach to Solving an Inverse Problem

In order to solve this problem an inverse approach which makes use of finite element methods was used. Instead of supplying the finite element software with given loading conditions to produce the deflection curve of the beam, the opposite was done; the loading conditions were solved with a given displacement curve. Discrete points along the deflected beam can be extracted from the relative height data obtained from the laser interferometer images. Together these points make up the shape of the deflected beam that is input to ANSYS to determine the force. Each image of the beam is coupled with the corresponding position of the piezo actuator when it was taken, and the height data is offset by this value so both the force and indentation depth can be extracted from ANSYS in one step.

In ANSYS, each point is modeled as a node with elastic beam elements forming the connection between each node. Elastic beam elements are acceptable to use because the cantilever beam will only be subjected to small deformations within its elastic region. Initially,

each node is created along the x axis with only their lateral position as a coordinate and the elements are created to connect them. Next, the height information for each node is entered as a displacement constraint. The only constraints that can be placed on the nodes are the displacements in the y direction. Any other constraints such as a zero slope at the first node cannot be confirmed and must be omitted (Moaveni, 2003).

Once all of the information has been entered for each node, the inverse problem is ready to be solved using ANSYS. Rather than specifying an applied force, the displacements are specified, and the force that caused them is solved for. ANSYS does this using a method similar to that used for solving traditional beam problems. Normally, it uses stiffness and load matrices, in order to find a displacement matrix that describes the deflections of the beam. With an inverse approach, ANSYS begins with the stiffness and displacement matrices and instead solves for the load matrix.

2.2.3. Mathematics behind the ANSYS Inverse Solution

ANSYS begins its solution by determining the shape functions for the cantilever. Because the beam is modeled using ANSYS's predefined beam elements, it uses the following cubic equation to show the general displacement field along the length of the beam.

$$v = c_1 + c_2x + c_3x^2 + c_4x^3 \quad (.)$$

This model, the input conditions about the element, and the shape functions are used in combination to define the shape of the element. The shape functions are defined as follows.

$$\begin{aligned}
S_{i1} &= 1 - \frac{3x^2}{L^2} + \frac{2x^3}{L^3} \\
S_{i2} &= x - \frac{2x^2}{L} + \frac{x^3}{L^2} \\
S_{j1} &= \frac{3x^2}{L^2} - \frac{2x^3}{L^3} \\
S_{j2} &= -\frac{x^2}{L} + \frac{x^3}{L^2}
\end{aligned} \tag{.}$$

The next step is to calculate the strain energy for the beam element in order to determine the stiffness matrix which will eventually be used to determine the reaction forces. The formula for the strain energy of a beam element is:

$$\Lambda = \frac{EI}{2} \int_0^L \left(\frac{d^2v}{dx^2} \right)^2 dx \tag{.}$$

By substituting in the beam displacement model and simplifying it into a matrix form, the modified strain energy formula is:

$$\Lambda = \frac{EI}{2} \int_0^L \{U\}^T [D]^T [D] \{U\} dx \tag{.}$$

Finally, ANSYS uses the fact that the total potential energy of the beam element is equal to the strain energy less the work done by the forces acting upon it. It is this fact that allows ANSYS to be able to solve for the reaction forces when given displacement inputs. The principle of the minimum total potential energy states that the total potential energy must be a minimum for a stable system. This leads to the following equation where the strain energy is differentiated with respect to the element end conditions.

$$\frac{\partial \Lambda}{\partial U_k} = EI \int_0^L [D]^T [D] dx \{U\} = \frac{EI}{L^3} \begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \begin{Bmatrix} U_{i1} \\ U_{i2} \\ U_{j1} \\ U_{j2} \end{Bmatrix} \tag{.}$$

The 4 by 4 matrix is known as the stiffness matrix and the 4 by 1 matrix is known as the displacement matrix. When multiplied together, these two matrices yield the reaction forces or the load matrix (Moaveni, 2003).

The individual stiffness matrix for each element is found by ANSYS and then combined to create the global stiffness matrix. The matrices can be combined because the reaction forces at the nodes where elements intersect one another are equal. At this point, ANSYS would normally find the inverse of this matrix and multiply it by the load matrix to determine the displacement matrix corresponding to the applied loads. Instead, ANSYS multiplies the global stiffness matrix by the global displacement matrix which is made up of the displacement data that is input to ANSYS. This multiplication gives a load matrix that contains the reaction forces on the beam being modeled (Moaveni, 2003).

2.2.4. Extracting Force and Displacement Data from ANSYS Solution

To extract the solution for the force on the end of the beam from ANSYS, the Reaction Solution feature of ANSYS's post-processing menu is used. The indentation depth, if any indentation has occurred, is found using the Nodal DOF Solution to find the displacement of the end of the beam in the y direction. These two pieces of information can later be used to create a single point on a force displacement curve for the indentation process (Moaveni, 2003).

```
# Assign beam properties
beamprops = "R,1," + str(area) + "," + str(I) + "," + str(beamheight) + ", , , ,\n"
commands. append(beamprops)
# Assign material properties
commands. append("MPTEMP,,,,,,,,\n")
commands. append("MPTEMP,1,0\n")
commands. append("MPDATA,EX,1,,150000\n")
commands. append("MPDATA,PRXY,1,,0.17\n")
```

Fig. -. Python code to add beam and material properties.

2.3. Validation of Algorithm with Comparison to Theoretical Results

Once it was shown that ANSYS was capable of solving the inverse problem using the algorithm with sufficient accuracy, verification was needed to prove that the algorithm could produce the theoretically expected data for a simulated indentation process. To do this, the analytical solution of an AFM cantilever probe being used to indent a sample of aluminum to a maximum penetration depth of 100 nanometers was used.

2.3.1. Creation of Simulated Indentation Data

For the validation of the algorithm, a set of data similar to one that would be obtained during an indentation was required. Since the algorithm was validated before the construction of the indenter, an example set of data that could be seen during an indentation needed to be analyzed. To make this data, an iterative model that created a new data point for each step of the piezoactuator was used. The model created ANSYS script files for each indentation step to solve for the force and displacement at that step. To begin, all the material properties and physical characteristics of the beam were input to the program using the code in Fig. -.

To make this sample data, the material properties of aluminum were used as the simulated sample. For the finished indenter, the material properties of the sample are not needed before indentation because they can be extracted from the indentation data. The material properties are needed for the validation because no physical indentation has taken place. Each time the piezo position was increased in the simulation, the beam deflection was recalculated, and the subsequent penetration depth was determined. The penetration depth during the indentation is equal to the absolute difference between the position of the piezo actuator and the deflection of the cantilever beam. For each step in the indentation, a different ANSYS script file was created to replicate the contour of the beam at that position. The code in Fig. - adds the nodes and elements to the ANSYS script and applies the constraints on the nodes to create the contour.

The completed Python script was used to make all the ANSYS script files to simulate each step in the actuation of the piezo and find the expected of force and penetration for the given extension. The program began with the simulated piezo initially not extended and the

```

# Create nodes
i = 0
while i <= numelements:
    command = "N, ," + str(beamlength / numelements * i) + "0,0,,,,,\n"
    commands. append(command)
    i = i + 1
# Create elements
j = 0
while j < numelements:
    command = "FLST,2,2,1\nFITEM,2," + str(j+1) + "\nFITEM,2," + str(j+2) + "\nE,P51X\n"
    commands. append(command)
    j = j + 1
# Apply y constraints to the nodes
i = 0
while i <= numelements:
    ydisp = -piezoposition + disp(beamlength / numelements * i, force, E, I, beamlength)
    command = "FLST,2,1,1,ORDE,1\nFITEM,2," + str(i + 1) + "\n!* \n/GO\nD,P51X, ," +
    str(ydisp) + ", , , ,UY, , , ,\n"
    commands. append(command)
    i = i + 1

```

Fig. -. Python code to create the nodes and elements for the ANSYS script

simulated cantilever just in contact with the sample. In five nanometer increments, the position of the simulated piezo was increased. The extension of the piezo causes the beam to deflect and a force at the tip of the beam to act on the sample. If the force is high enough, the tip will penetrate the sample a distance proportional to the force. Once the penetration depth reached the depth of 100 nanometers, the set of data to simulate the indentation was complete. The process was then reversed to obtain the data for the tip being retracted from the sample.

The data created was not just a force and a displacement. Instead, the data synthesized for each step was a series of points representing the contour of the cantilever at the given instant. This is analogous to the beam contour data that can be obtained from an image taken with a laser interferometer. In this way, the simulated dataset can be used to validate that the program can take input from laser interferometer images.

2.3.2. Extraction of Force and Displacement Information from Dataset

Once the simulated dataset was created, the validity of the program could be tested by feeding this dataset as input to the program. For each simulated image the program takes the information and creates an ANSYS script file. The ANSYS script file is comprised of 19 nodes and 18 elements that describe the specific contour of the beam for that image. Each of the script files are opened in ANSYS and solved to obtain the individual force and indentation data.

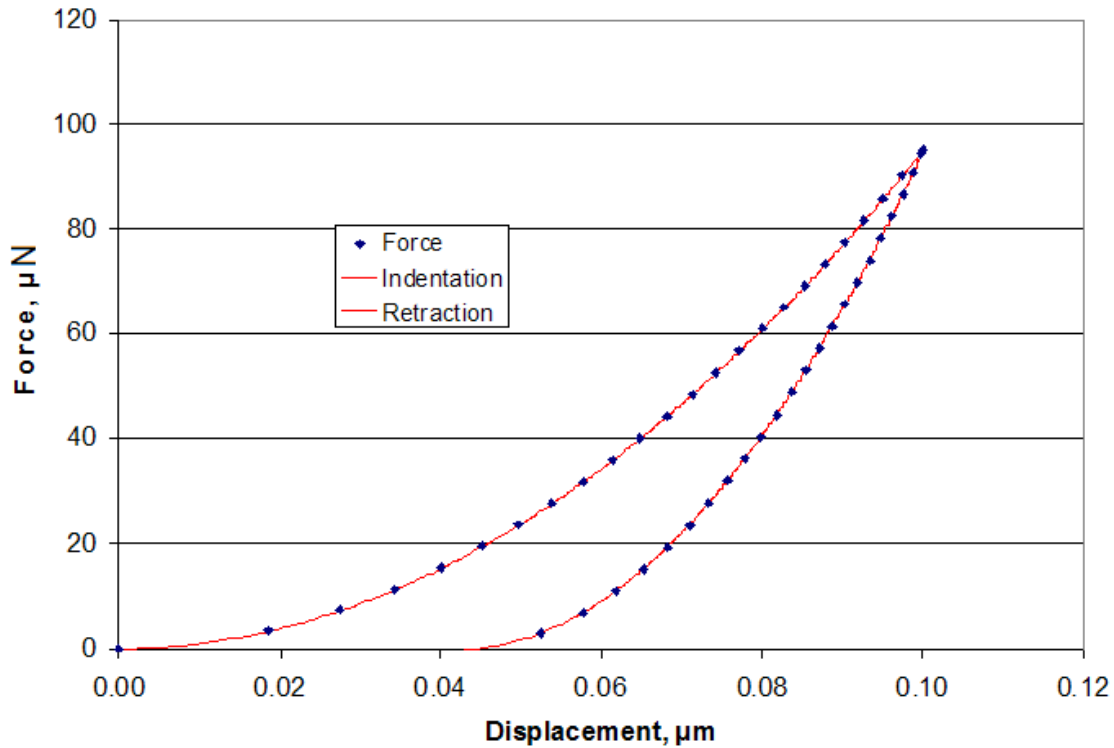


Fig. -. Comparison to theoretical force displacement data.

2.3.3. Comparison of Algorithm Data to Theoretical Data

Using another script that opens, solves, and extracts data from all the ANSYS script files, all the force and indentation depths were automatically extracted and stored the data in a

spreadsheet file. A graph of the theoretical force versus displacement curve was made using Excel and the data obtained from the algorithm was overlaid on top of the curve. This curve can be seen in Fig. 2-6 and the curve created by the data points obtained from ANSYS falls exactly on the theoretical solution curve.

3. Indentation Approaches

Before deciding upon the final design of the nanoindenter, several different design configurations were considered. Research was also done into alternate methods of directly measuring force and displacement; these results can be seen in Appendix B, along with research done on commercially available nanoindenters and nanoindenter tips that was used to compare designs. Finally, the potential nanoindenter configurations are explained and analyzed to determine the design for the nanoindenter.

3.1. Previous Designs

Once research into commercially available components and systems had been completed, work was done to produce several viable concepts for a low-cost indentation module. The goal was to produce a design that could function as a stand-alone module for a laser interferometer, provide a quality of measurement near that of the commercially available systems, and remain below the \$5,000 budget for the project.

3.1.1. Macroscale Cantilever Concept

One of the first designs considered was a macroscale cantilever, which held a commercially-obtained indenter tip at the end. Strain gauges were used to measure the deflection in the cantilever, which, through the use of beam deflection equations, could be used to derive the force applied to the cantilever. To measure the deflection at the end of the cantilever, a capacitive or inductive displacement sensor would be mounted with one segment on the cantilever, and the other mounted on the fixturing that held the piezoactuator. The sample would be held on a stage which allowed coarse vertical motion, which would allow the sample to be brought into contact with the indenter tip. Confirmation of contact with the sample would be provided by the displacement sensor, as a small deviation would be observed from the previous displacement reading.

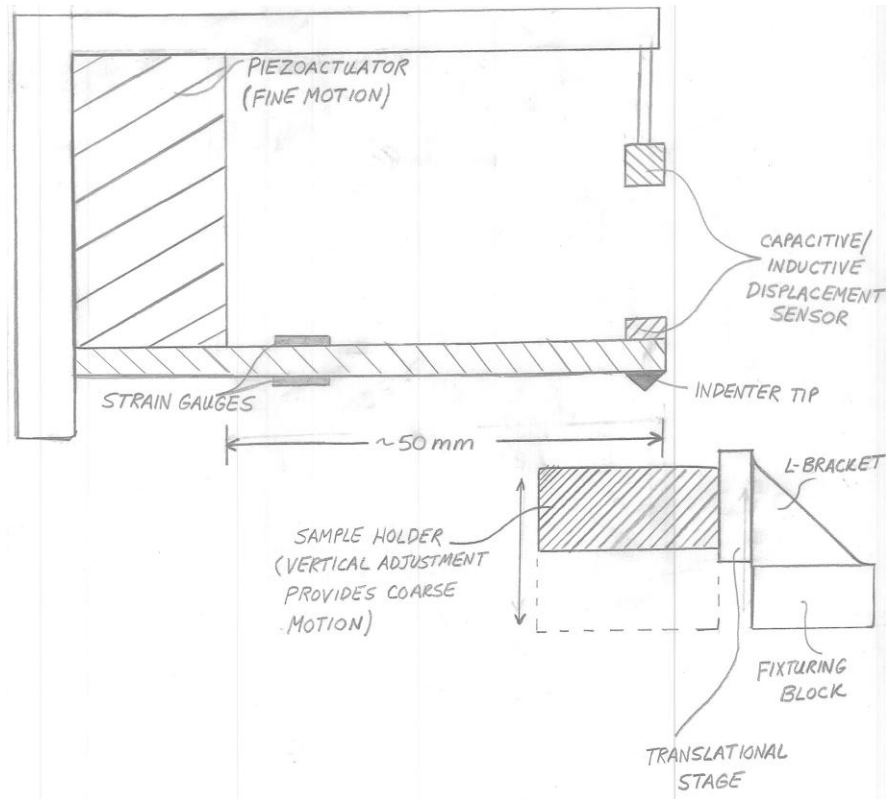


Fig. -. Macroscale Cantilever Concept

Once the tip was brought into contact with the sample, the piezo would then be actuated in small increments, and the readings from the strain gauges and displacement sensor would be used to construct the force-displacement curve.

In addition to the components shown in Fig. 3. 2, an analog-to-digital converter would be needed, as well as amplifiers for the strain gauges and the displacement sensors. This concept was also designed to be a stand-alone module, and, because of the force and displacement sensors, did not require the use of the laser interferometer to obtain readings of force and displacement. Theoretically, this configuration could have provided resolutions of 0. 6 micrometers for the displacement, using a commercial inductive displacement sensor, and 162 micro-Newtons, if platinum-tungsten strain gauges were used in a full-bridge configuration.

3.1.2. Voice Coil Actuation Concept

Another design taken under consideration was a design that was comprised of three separate mounting plates connected by folded, leaf-type springs. The indenter tip is a magnetic rod rigidly mounted on the center plate, and its motion is controlled by the voice coil on the third plate. Thus, when the voice coil is energized, the leaf spring between the plates holding the indenter tip and the sample is compressed. The sample itself is mounted to a stage that allows coarse motion in the lateral direction, allowing the sample to be positioned in contact with the tip of the indenter before the voice coil is energized.

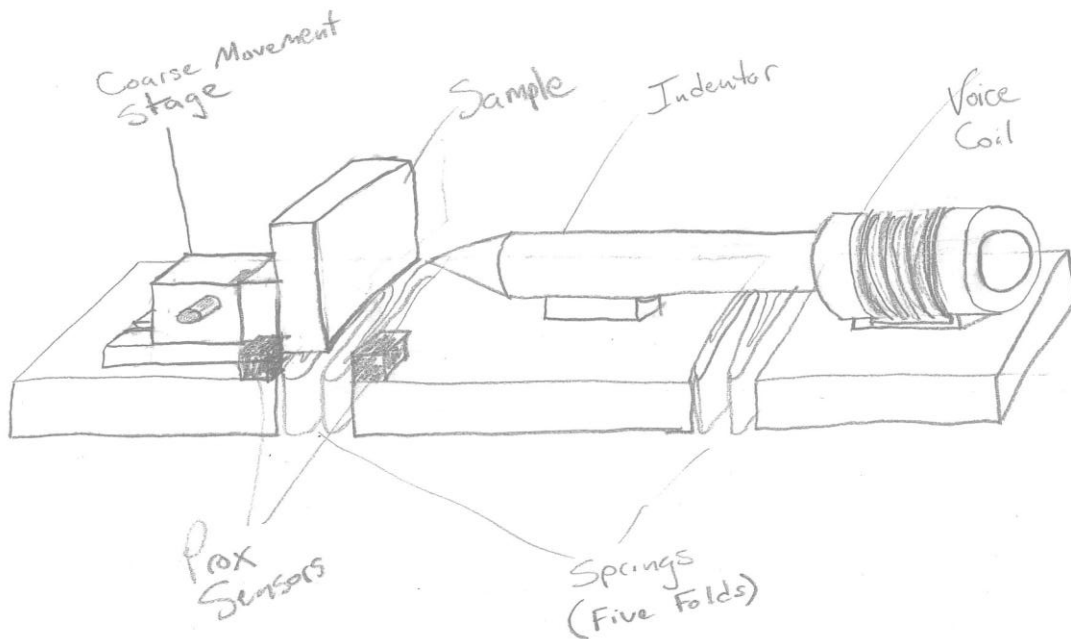


Fig. -. Voice Coil Actuation Concept

For this design, force measurements would come from the specified input voltage to the voice coil. Position measurements are obtained using a capacitive or inductive sensor, as with

the macroscale cantilever design; one side of the sensor is mounted to the plate with the sample, and the other side is mounted to the plate that holds the indenter tip. Thus, as the voice coil repelled the magnetic rod with the indenter tip, the spring between the plates with the sample and the indenter tip would compress, and indentation would occur. This configuration uses the inductive displacement sensor described above, resulting in the same 0.6 micrometer resolution for displacement; based on a minimum of 8 microamperes to actuate the voice coil, the minimum force resolution is 0.018 micro-Newtons.

3.1.3. Hydraulic Column and Pressure Transducer Concept

A third potential design was synthesized using a column of incompressible fluid and a pressure transducer. This concept was created in an attempt to exploit the force-multiplying relationship between the surface area of piston and the pressure applied to it. In this design, a closed-loop piezoactuator was used, providing the displacement the indenter tip had traveled. The resulting force would then be transmitted to the column of fluid, which is monitored by a pressure transducer. By using pistons of different known sizes and the readings from the pressure transducer, the resulting force can be determined, and the force-displacement curve can be constructed. One of the driving forces behind this design was the desire to magnify the forces, expected to be on the order of micro-Newtons, to a range at which they could be read by commercial devices, whether they be pressure transducers (as in this application), force transducers, or displacement sensors.

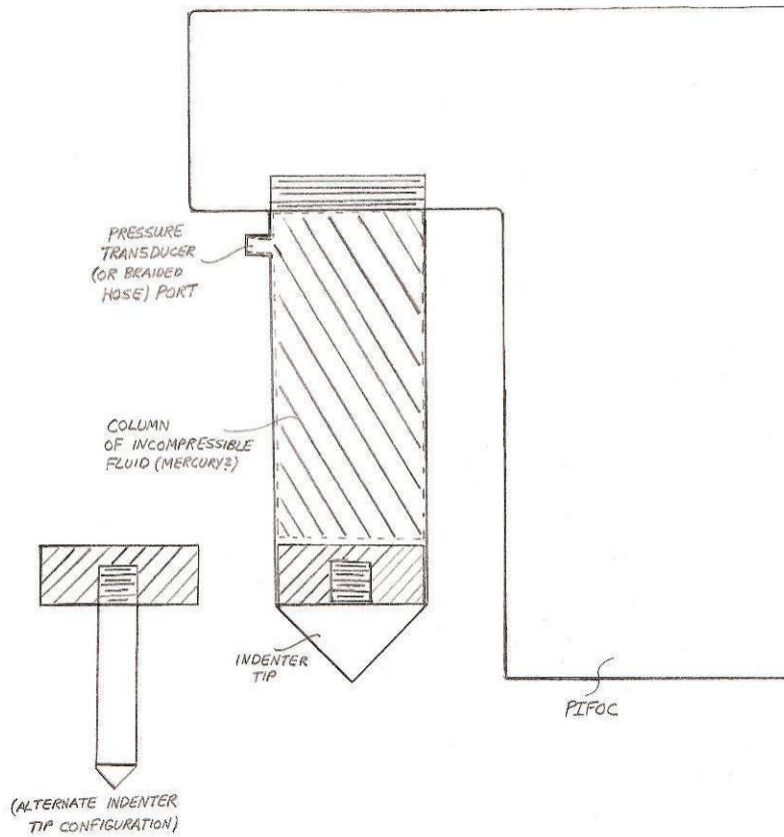


Fig. -. Hydraulic Column and Pressure Transducer Concept

Amongst the drawbacks to this design were the problems of encapsulating of the fluid, to prevent leaks and erroneous readings, as well as obtaining a pressure capable of being read by a commercially available pressure transducer. Also, this design suffered a setback when the piezoactuator shown in the diagram was found to be inoperable, and repair was determined to be cost prohibitive.

3.1.4. AFM Cantilever and Laser Interferometer

As discussed previously, the distinctive aspect of this approach to nanoindentation is the inverse approach, where the force applied to and the displacement experienced by the cantilever probe are not determined through a direct manner, but through the analysis of images obtained using laser interferometry. The concepts presented previously, however, did not use this inverse method; rather, the force and displacement data was obtained from direct methods. Although direct methods will give the data necessary to construct the force-displacement curve, the inverse method was selected because it represents a lower-cost approach to nanoindentation that requires fewer components in the indenter assembly, as well as more precise measurement of both force and displacement data.

To achieve these goals, a fourth and ultimately final design was that which involved the use of a piezoactuator, cantilever probes used in atomic force microscopy, and a laser interferometer. This design is similar in concept to the macroscale cantilever presented in section , but has several noticeable differences. Since the cantilevers intended for atomic force microscopy are much too small to bond strain gauges to, the force and displacement are

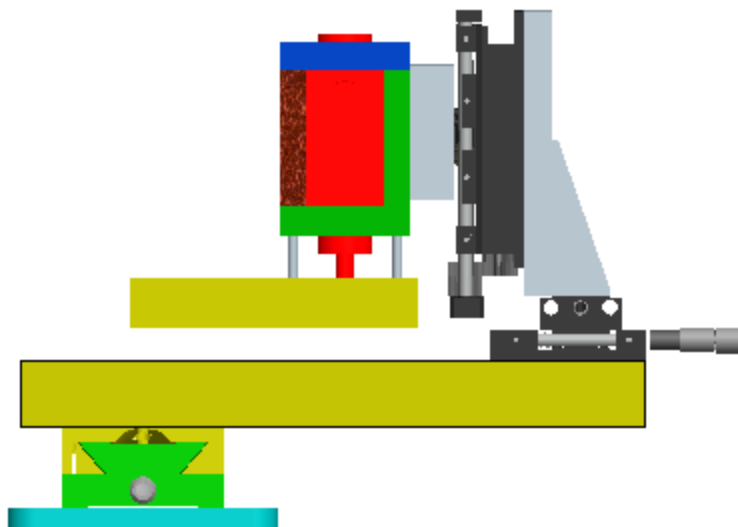


Fig. -. Three dimensional model of the piezo actuated AFM cantilever concept.

found using images taken with a laser interferometer. Laser interferometry is a principle by which a laser beam is emitted, then split, with a portion of the light reflecting off the object placed under the microscope, and the remainder of the light reflecting off a mirror at a known distance. The two reflected light beams' wavelengths are then compared, and the distance to the object can be determined (see section for further explanation). Through the use of laser interferometry, the distance to the top side of the cantilever probe can be determined. As the indentation process occurs, the beam deflects, thus changing the height of the beam relative to the interferometer optics. This change, detected as a series of black-to-white fringes, can then provide information about how much the probe has deflected, and the forces that caused the deflection.

To provide coarse motion, this design uses a system of stages in the x-, y-, and z- directions. For the x- and y-directions, manual staging was used to enable a series of indentations to be performed without having to change the position of the sample. For the z-direction, a motorized stage was used to provide a means of moving the entire indentation apparatus away from the sample for sample insertion and removal. Once the sample has been inserted, the assembly is brought close to the surface of the sample using the coarse adjustment provided by the motorized stage. Then fine adjustment of the probe can be achieved using the piezoactuator. Once it is seen that the probe is in contact with the sample, indentation would occur through further incremental motion of the piezoactuator. A series of images would then be captured at each indentation step, and post-processed using an inverse method of finite analysis to determine the force acting on and the displacement into the sample at that instant. From this data, a force-displacement curve can be obtained, yielding the same results as the macroscale approaches to indentation presented above.

This design, while utilizing the inverse approach, also provides the best resolution for the measurement of force and displacement; displacement can be measured at a minimum resolution of 1.23 nanometers, while force can be measured at a minimum resolution of 0.25 micro-Newtons.

Once this design was chosen as the candidate on which further design work and analysis would be completed, a motion analysis was done to determine the degrees of freedom necessary to implement this design. To position the entire module properly beneath the interferometer, stages that allow motion in the x-, y-, and z-directions are needed, as are two goniometer stages that control the rotational position in two orthogonal directions. In addition to positioning the module beneath the microscope objective, they are also needed to focus the interferometer beam on the cantilever probe, resulting in the fringes needed to obtain force and displacement data. Within the module itself, stages for controlling both coarse and fine motion in the z-direction are needed; the coarse motion stage is needed for the insertion of the sample, so as to not damage the probe, whereas the fine motion stage is required for the indentation process itself, and must be capable of the precision needed to perform the indentation. Additionally, x- and y-stages are needed to position the probe relative to the sample, permitting repeated indentations to be performed without having to manually move the sample or the probe.

The challenge, therefore, was to fit the hardware necessary to achieve these motions in the workspace provided beneath the interferometer module. Early in the concept development stage, it was seen that the entire module assembly could not be placed directly beneath the objective lens of the interferometer, as there was not enough available travel in the interferometer to prevent interference between the interferometer head and the module. Thus, attention was paid to ways in which several of the components could be moved from directly

beneath the interferometer beam while preserving the control each component had over the motion of the assembly. Due to these space constraints, it was decided that the entire module would be mounted on a cantilever beam, with the probe on a separate cantilever beam to place it under the objective. This was an acceptable solution, as there was more space available in the lateral direction.

Lastly, attention was paid to which each of the motions specified above could be controlled. It became apparent that commercial software and control hardware would be needed for the piezoactuator, due to the delicate nature of the device. A driver would also be needed for any of the other stages that were motorized; it was later determined that this driver could be controlled using simple text commands. Lastly, software would be needed to unwrap the phase of the images of the interferometric fringes, as well as to analyze the resulting data and construct the force-displacement curve.

4. Implementation

While the analytical part of the nanoindenter was being tested, the mechanical design of the device was being developed in parallel. The device had to be designed around the size constraints of the laser interferometer in the lab, which are non-negotiable. The design process included several iterations, first on the conceptual stage and later with more detailed three-dimensional solid models. Once the basic design concept was determined, it was then possible to conduct a finite element analysis in order to find the deformation of the system under the expected loads. The device was also put through a simple kinematic test to confirm that inducing motion would not cause the device to not oscillate beyond control. During manufacturing, extra precautions were taken to make each part meet its necessary specifications and to ensure that all pieces had exactly orthogonal faces. Any faces that were not perfectly flat or perpendicular to adjoining sides could have caused large errors in the final indenter setup. The full setup can be seen in Fig. - and the side view is shown in Fig. -.



Fig. -. Picture of completed module, mounted and ready to use

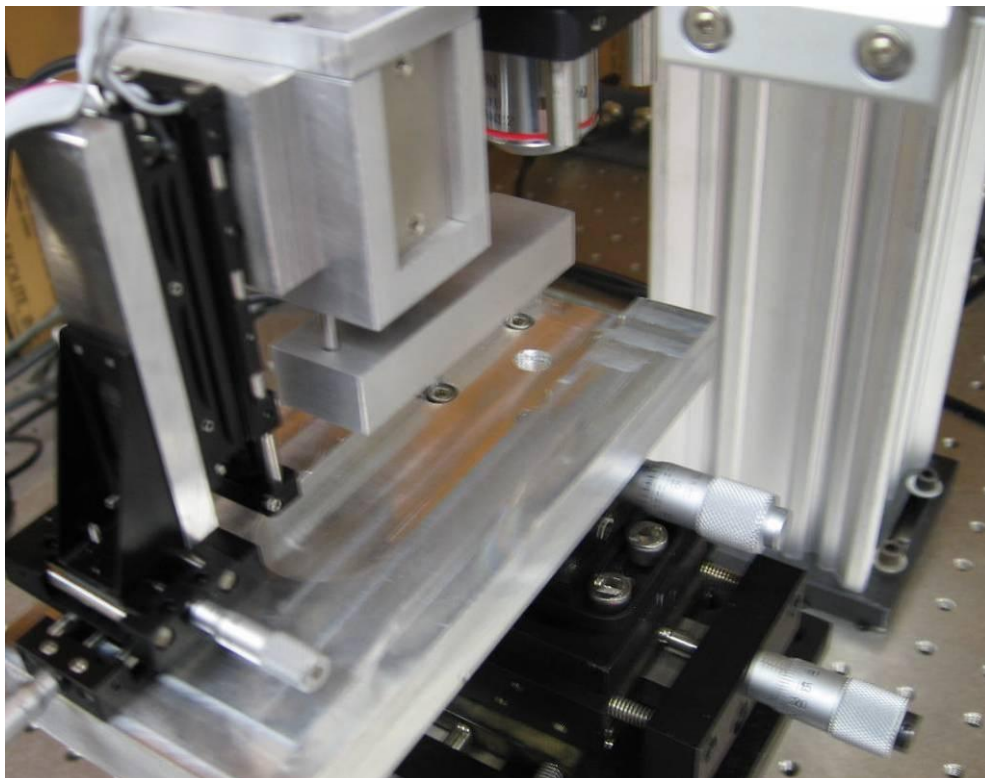


Fig. -. A side view of the module that is mounted underneath the laser interferometer.

4.1. Final Design of the Nanoindenter

After reviewing the potential design concepts outlined in section , a final design

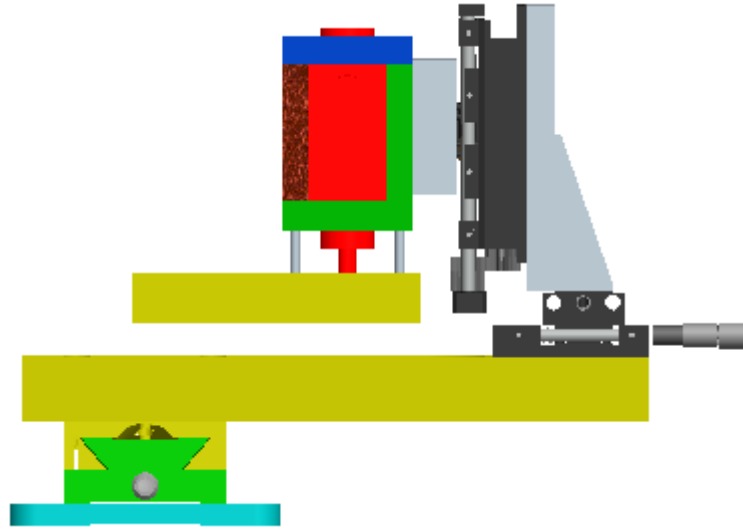


Fig. -. Final configuration of the nanoindenter.

comprised of using an AFM probe actuated by a piezo electric actuator was decided upon as shown in Fig. -. From this design, an initial three dimensional model was created to see if the

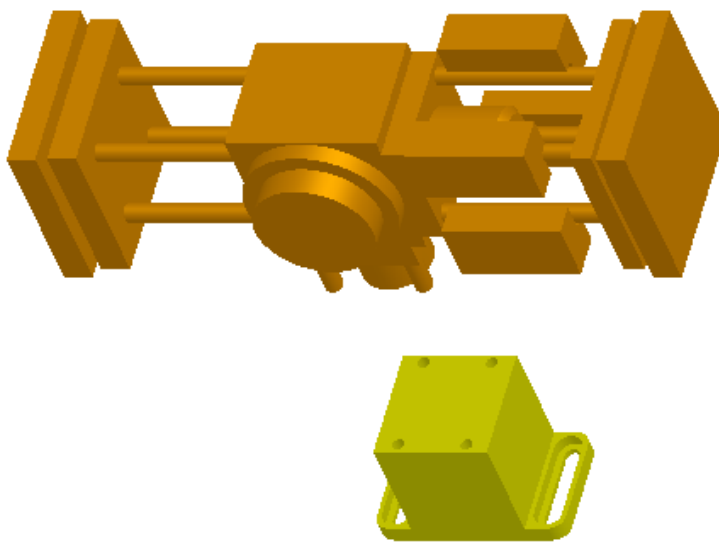


Fig. -. Model of the interferometer and the goniometer.

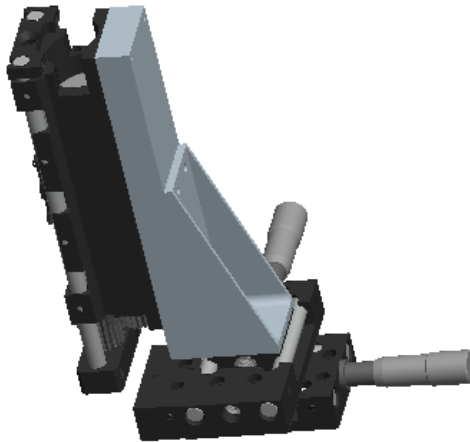


Fig. -. Three axis translation stage component of the nanoindenter.

basic idea made sense and the mechanism would fit within the space available under the microscope objective. When modeling the parts, the interferometer and the translation stages underneath it were modeled to provide a three dimensional workspace to develop the parts for the indenter. The model of these parts can be seen in Fig. -.

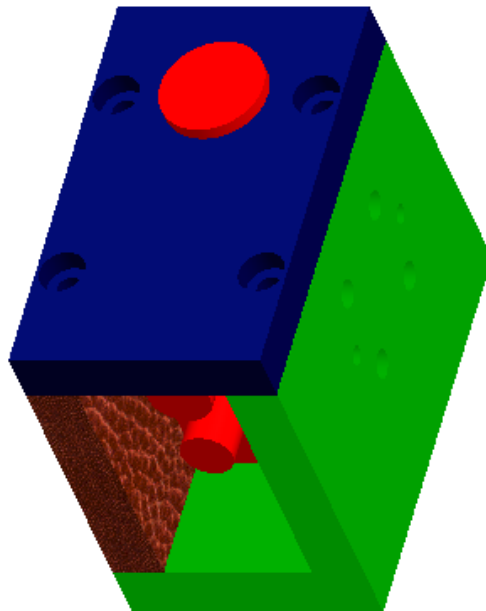


Fig. -. Piezo enclosure model.

From this model, the structural components for the indenter were designed. The final configuration of the design consists of a base plate that has the indenter mounted on one end and is connected to the translation stage on the opposite end. A three axis translation stage forms the base of the indenter and is connected directly to the base plate. The two axes in this stage parallel to the base plate are moved using manual micrometer drives and each have a travel of 0.200 inches. The third axis is a motorized stage with 1.000 inch of travel that is controlled through serial data input. A three dimensional model of this stage is found in Fig. -.

Attached to the vertical axis of the translation stage enclosure were machined with the tightest possible tolerances to ensure that no stress is placed on the piezo when the enclosure is assembled. The tolerances were also important because the pieces of the assembly needed to fit together perfectly. A model of the piezo enclosure assembly with the piezo installed can be seen in Fig. -.

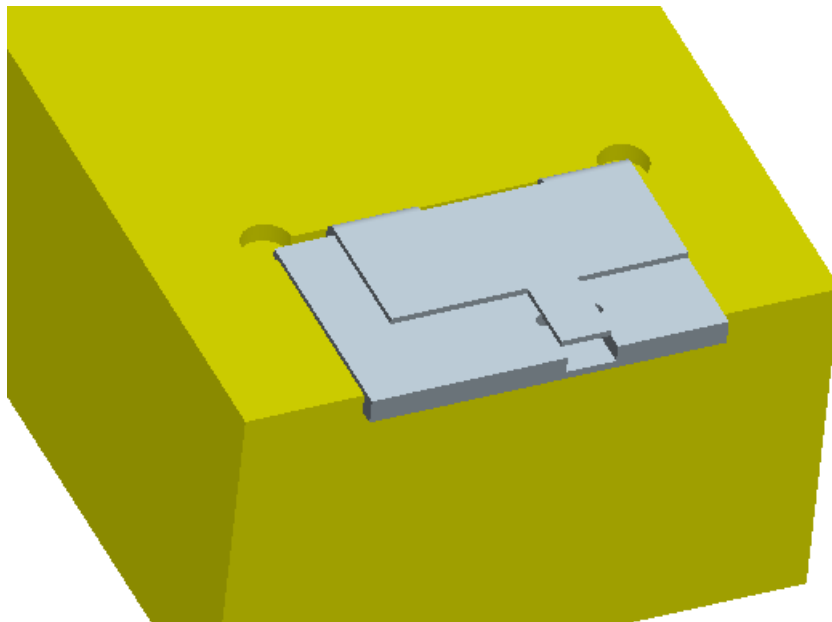


Fig. -. Recessed area in the cantilever with the probe holder.

The final component of the nanoindenter, a cantilever beam used to position the AFM underneath the microscope objective, is connected to the bottom of the piezoelectric actuator by a screw. Two alignment pins are used to keep the beam in the proper orientation. At the end of the cantilever, there is a recessed area with a magnet that holds the AFM probe holder as shown in the figure above.

A summary of the components discussed and the other parts purchased to construct the nanoindenter can be found in Table -.

Table -. Summary of nanoindenter components.

<u>Commercially Available Components</u>		
<u>Quantity</u>	<u>Supplier</u>	<u>Parts List</u>
1	Thorlabs	PAZ-020 Closed-Loop Piezoactuator (20 μm travel)
1	Thorlabs	TSG001 USB Strain Gauge Reader
1	Thorlabs	TPZ001 USB Piezo Controller
1	National Aperture	MM-3M-F Motorized Stage
1	National Aperture	NAI MVP-1 Motion Controller
1	National Aperture	MM-2-CR-XY Stage
1	Pacific Nanotechnologies	Tip Holder P-000-0006-0
N/A	Nano and More	AFM Probes (DT-NCLR, CDT-NCLR)
<u>Manufactured Components</u>		
<u>Quantity</u>	<u>Supplier</u>	<u>Parts List</u>
1		Base plate
1		L-bracket
1		Side plate
1		Top Plate
1		Stage-to-piezo adapter
1		Inter-stage adapter
1		Cantilever

4.2. Construction and Setup of Nanoindenter



Fig. -. Laser interferometer in the CHSLT.

The laboratory setup used to control the nanoindenter is housed in the Center for Holographic Studies and Laser micro-mechaTronics at Worcester Polytechnic Institute. The main component of the device is the laser interferometer, which is used to measure the forces and displacements. An image of the laser interferometer in the CHSLT can be seen in Fig. -.

The lab setup, implementation, and design of the nanoindenter are all extremely crucial to the project. While the group can analytically prove everything about nanoindentation, if the group could not produce results from the device it does not matter what is possible. The group formulated many designs using many different methods of measurement of both distance and force along with actuation methods, but what is important is that a design that could perform with high resolution and accuracy was created. The design which was created had many components some of which were purchased, some of which that were manufactured by the

group, and some that were already available to the group in the lab. The next few sections describe the components and such complications or successes that were experienced with such components.

4.2.1. Laser Interferometer

The laser interferometer was the initial piece of lab equipment that the design was based around. This was the piece of lab equipment that was available and could be utilized to its full extent in the testing portion of the project. With the ability to measure a deflection of 1.23 nanometers on the cantilever beam, this device supplied the group with more than necessary resolution. Having the high resolution was a big advantage and also allows for further expansion of the nanoindenting device. One of the biggest disadvantages to using the interferometer was that there was a small amount of space underneath the lens. This disadvantage required modifying the design many times to be able to use the piezo with the interferometer. Since the piezo was just about the size of the available space underneath the interferometer lens, the group had to design something that was able to sit as close to the interferometer as possible without hit the interferometer and then extend an arm out underneath the lens of the interferometer.

4.2.2. Piezo: Open Loop vs. Closed Loop

The piezo is one of the components that the group knew had to be used somewhere in the design of the nanoindenter. Research was conducted in other methods of translation, such as voice coils with spring stages and MEMS devices, but none the other designs could achieve the right resolution and could be designed and manufactured in the time allotted for the project.

Where those methods of translation failed, the piezo succeeded and also provided more than required translation which could be used to take study further by upgrading the cantilever beam.

Since a piezo was necessary for the group's design to function, the group then had to determine whether to use a closed loop piezo or an open loop piezo. At first, the group was willing to use an open loop piezo and characterize the piezo to use in the nanoindenter since there was access to one in the lab already, but our advisor informed the group that there would not be enough time to characterize it and that is another research project in itself. In fact, the group had a chance to take a look at a project that had been completed in past years of a student who had characterized an open loop piezo, but his data was unable to be used for this project.

Now that an open-loop piezo was pretty much out of the question, more research had to be done in closed loop piezos. A closed loop piezo provided many advantages such that the group would not have to characterize it nearly as much since the strain gage feedback sensor would provide a position at all times during the test trials. Also the closed loop piezo provides easier mounting capabilities due to the strain gage feedback sensors attached to the piezo and that a power supply to drive the piezo was already available in the lab. A couple of the disadvantages to the closed loop piezo were that there weren't any in the lab and the group would have to buy one and also that a controller for the strain gage feedback sensor would have to be bought as well. Our advisor helped the group purchase the piezo and was able to overcome most of the disadvantages of the closed loop piezo. Since we had a piezo and a controller, along with the interferometer, the group was now ready to design some of the other components that were necessary to hold the piezo and position the cantilever underneath the interferometer.

4.2.3. Cantilever Beam

In the proposed design, there was not enough room for the piezo to fit directly underneath the laser interferometer so we had to produce a solution in which the translation of the piezo is moved directly underneath the laser interferometer. The solution the group created was a cantilever beam that is attached at the end of the piezo and the other end extends out so that this end is positioned directly in the line of the laser interferometer's sight.

4.2.4. Control Strategy

To control the indentation process, this design has both coarse and fine motion within the module. The coarse motion is a linear stage with one inch of travel between limit stops. This stage is used to raise the entire assembly towards and away from the sample. To provide inputs to the stage, the stage is connected to a controller which receives signals through an RS-485 port. This connection is made to the computer using a null modem cable, which allows the controller to communicate and handshake with the computer, and an RS-232 port on the computer. Commands to control this device are input through the use of HyperTerminal, using the knowledge provided by the manufacturer of the stage that one encoder pulse is 0.49609 micrometers (National Aperture, 2008). Thus, commands can be given in terms of the number of encoder counts, and the distance the stage has traveled will be known.

Once the indentation assembly has been placed close to the sample by the coarse motion motorized z-stage, the closed-loop piezoactuator is used to provide fine motion, both until the probe makes contact with the sample, and during the indentation and retraction processes. To control the piezoactuator, two Thorlabs T-cubes are used, one to control the motion of the piezoactuator, and one to read the feedback signal from the strain gauge module on the piezo.

Both of the cubes are connected to the computer via USB cables, and are interconnected using an SMA-to-SMA cable (Thorlabs, Inc. , 2008). Connections are made from the piezo controller cube using a 7-pin LEMO connector to provide the drive signal and voltage, and using an SMC-to-SMC cable to connect the strain gauge feedback module to the strain gauge reader cube (Thorlabs, Inc., 2008). Using the provided software, the piezo can be zeroed; that is, the output displacement can be calibrated to the output voltage. Commands can be input to this system of controllers using either the digital potentiometer knobs on the cubes themselves or the APT User software that is provided.

4.2.5. Sample Preparation

During the early indentation tests, a proper sample had not been created, so a piece of aluminum with a surface roughness that was obviously too large for the application. These tests were valuable because they provided images that could be used to work on the post-processing algorithm, but it was impossible to see any indentations on the surface.

For the final indentation tests, an aluminum disc sample was prepared with special care to minimize its surface roughness. When turning the piece in the lathe, the spindle was run at the fastest speed possible and with the slowest possible feed rate. This produced the best surface finish the lathe could possibly make. From there the piece was brought to a buffing wheel and was polished for several hours using burnishing compound. Once all of visible surface imperfections were removed with the buffing wheel, the sample was brought to the Sample Preparation Laboratory in Washburn Shops. There, the sample was further polished with three progressively finer polishing wheels down to a surface roughness of 50 nanometers. Since the

predicted indentations with the AFM tip are greater than 50 nanometers, any indentations could be distinguished from the background and this was an acceptable surface roughness.

4.3. Indentation Procedure

1. Using the motorized z-axis, move the assembly to the top of its range of travel.

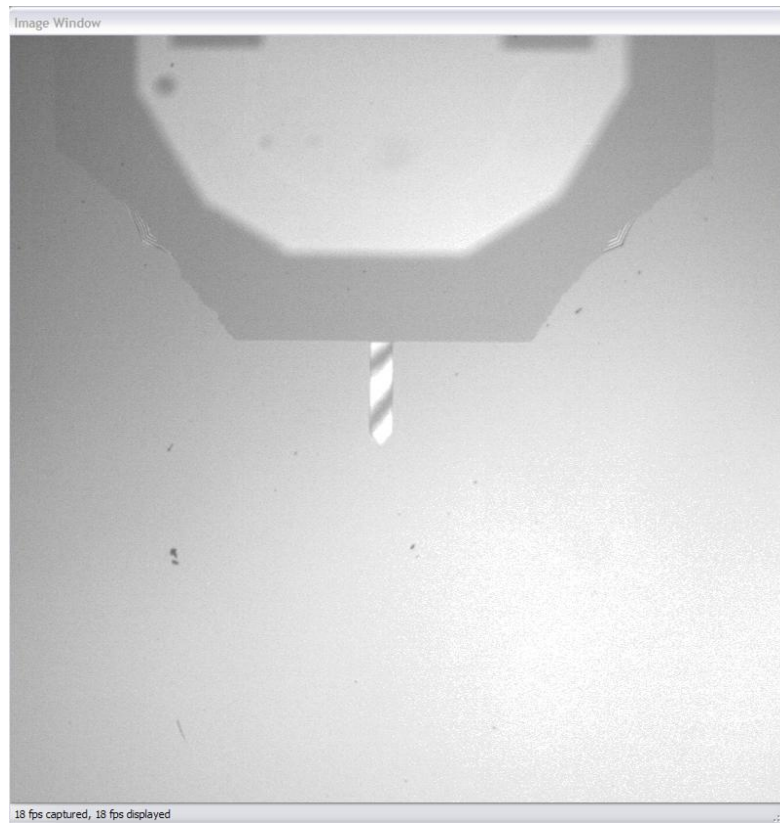


Fig. -. Live view of LaserView with the cantilever brought to the top of the motorized z-stage.

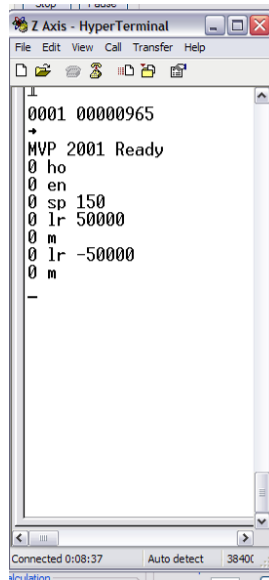


Fig. -. The HyperTerminal showing the z-stage retracted to top of the z-axis.

2. In the APT User software, use the self-calibrate command to zero the piezoactuator displacement with respect to the piezoactuator voltage.
 - a. Using either the APT User software provided by Thorlabs, or the button on the strain gauge reader module, select “position” mode from the list of options.

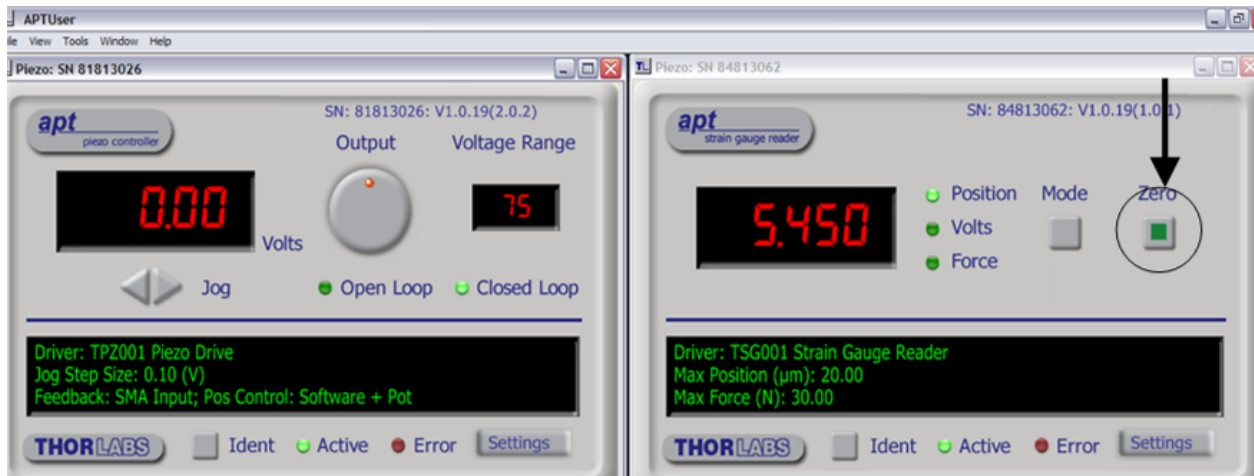


Fig. -. APT software used to zero the strain gauge reader for the closed loop piezo (Thorlabs).

- b. Using the APT User software, select “closed loop” on the panel that controls the piezoactuator.

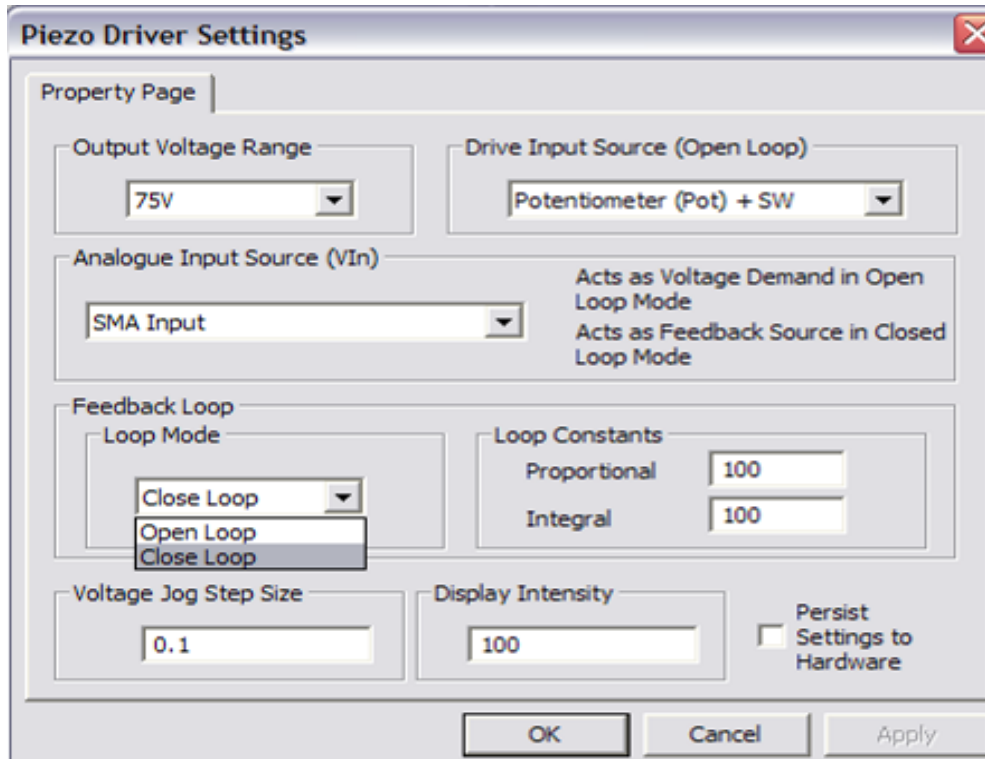


Fig. -. The settings menu which shows the closed loop tab selected to put the piezo into closed loop mode. Note: The SMA Cable must be connected from the strain gauge reader to the piezo driver.

- c. Zero the piezoactuator using the “zero” command in the APT User software.

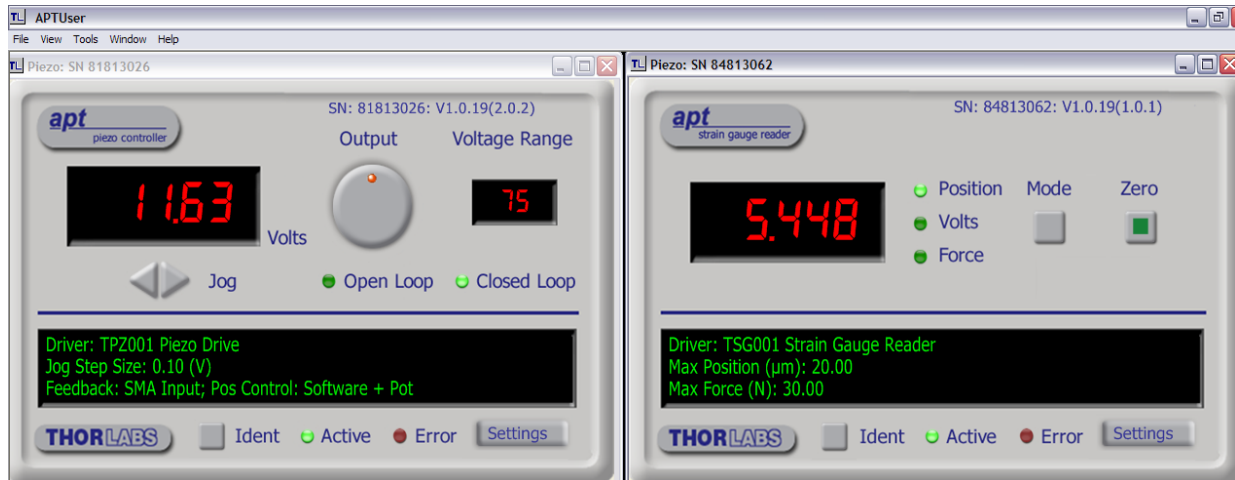


Fig. -. The Piezo is now zeroed in closed loop mode and also the two APT modules are active as shown in the software.

- d. While looking at the interface for the APT User software, make sure the “active” lights are green for the panels controlling both the piezoactuator and the strain gauge reader module. Insert the cantilever probe into the probe holder

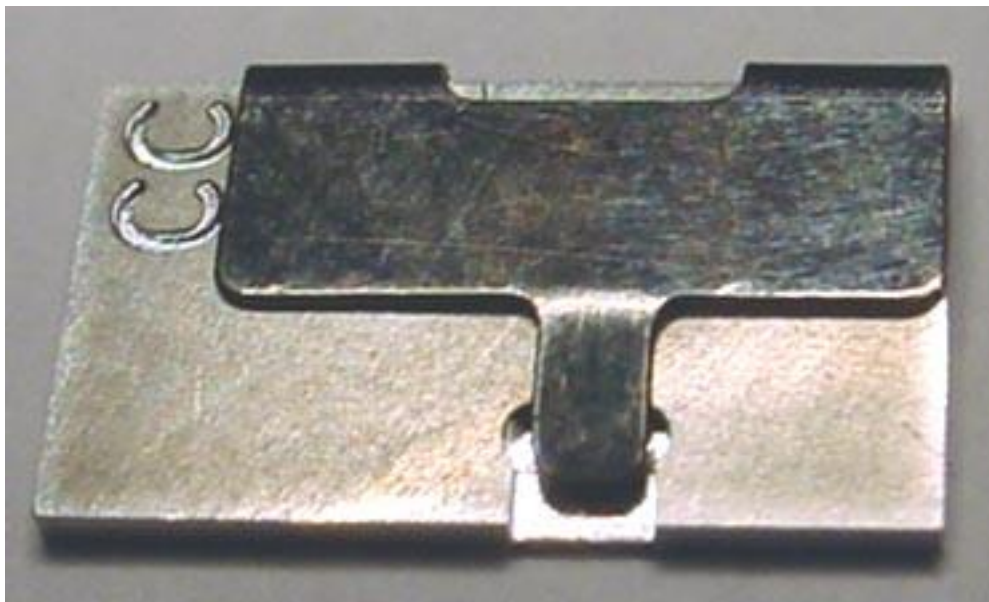


Fig. -. Close contact mode probe clip holder (Pacific Nanotechnologies, 2008).

- e. Use the SM108 tweezers and the tip holder opening fixture to position the cantilever probe in the probe holder. Then release the force on the tip holder opening fixture, and the probe should be secure in the holder.

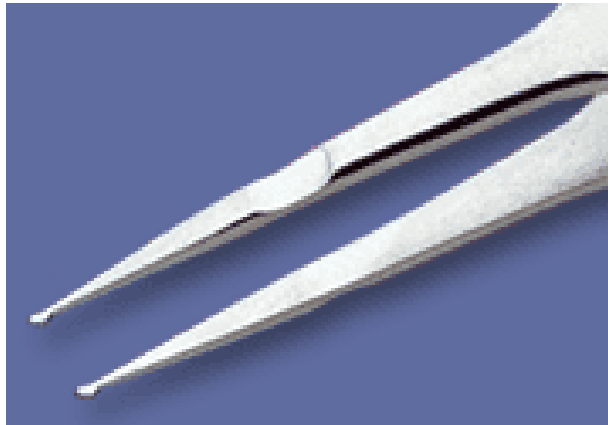


Fig. -. Special tweezers used to insert AFM probes in to Probe Clip Holder (TDI International, Inc. , 2008).

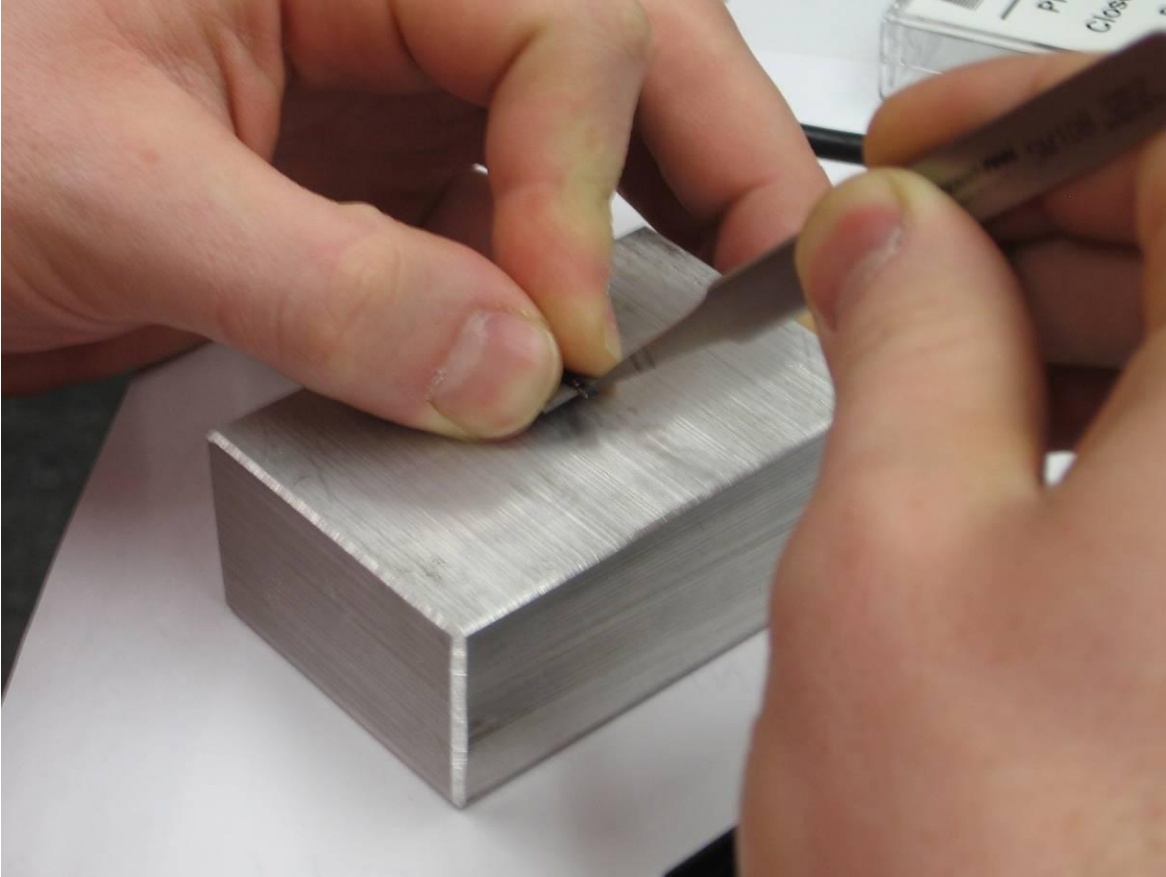


Fig. -. Placing the AFM probe into the Probe Clip Holder and using the group's own Probe Clip Holder opening device.

3. Very carefully insert the probe holder into the groove in the bottom of the cantilever.
Caution – any contact with other components or mishandling of the probe holder will result in the cantilever separating from the probe.
4. Place the sample that is to be indented into the sample holder.



Fig. -. The sample in which we are going to indent which has a surface roughness of 0.050 microns.

5. Insert the sample holder into the groove cut into the plate that supports the entire assembly. The magnet in the base plate should attract the magnet in the sample holder and hold it firmly in place during the indentation event.

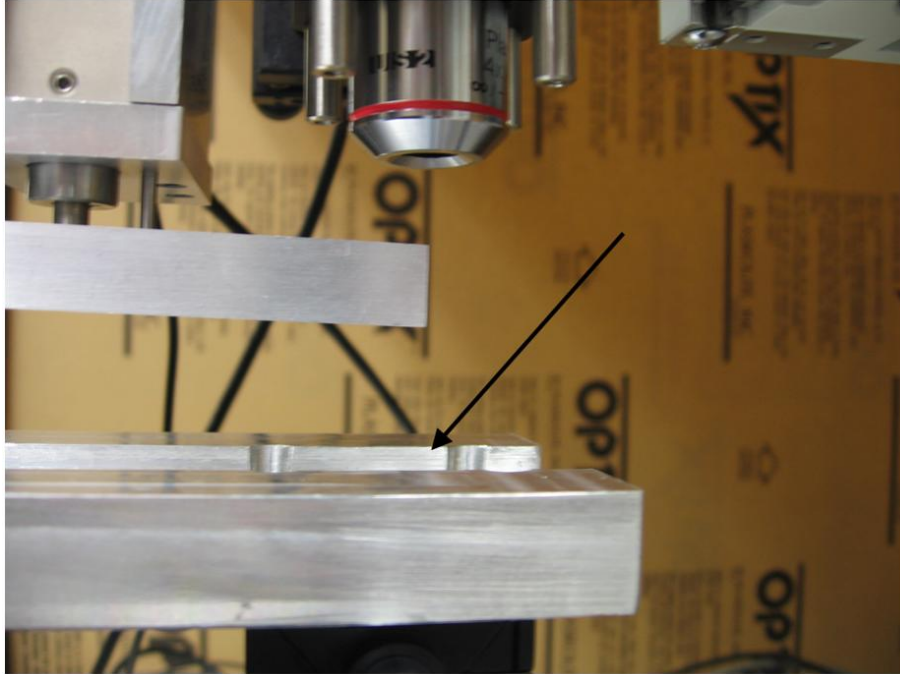


Fig. -. The module ready for the insertion of the sample as indicated by the arrow.

6. Open the LaserView program. Using the stages provided and the adjustment capabilities of the laser interferometry apparatus focus the interferometer on the probe such that a moderate number of interferometric fringes can be seen. Now, use the goniometer and manual z-axis stages to bring the fringes on the probe into focus. Ideally, the fringes should be perpendicular to the long axis of the probe, and there should be as few of them as possible (zero is the best-case scenario).

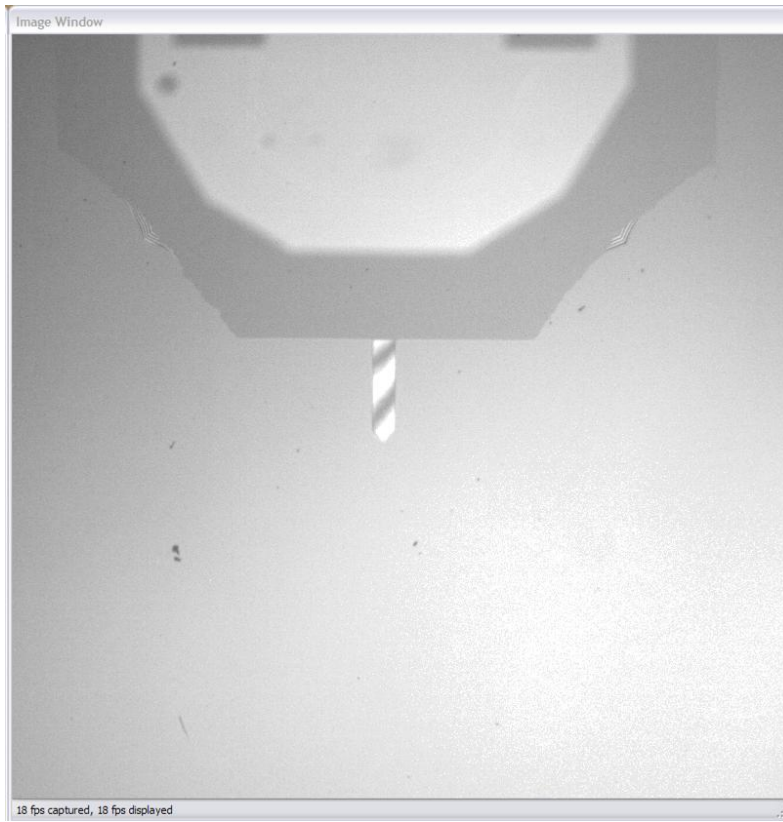


Fig. -. Probe brought into focus and before adjustments were conducted with goniometer.

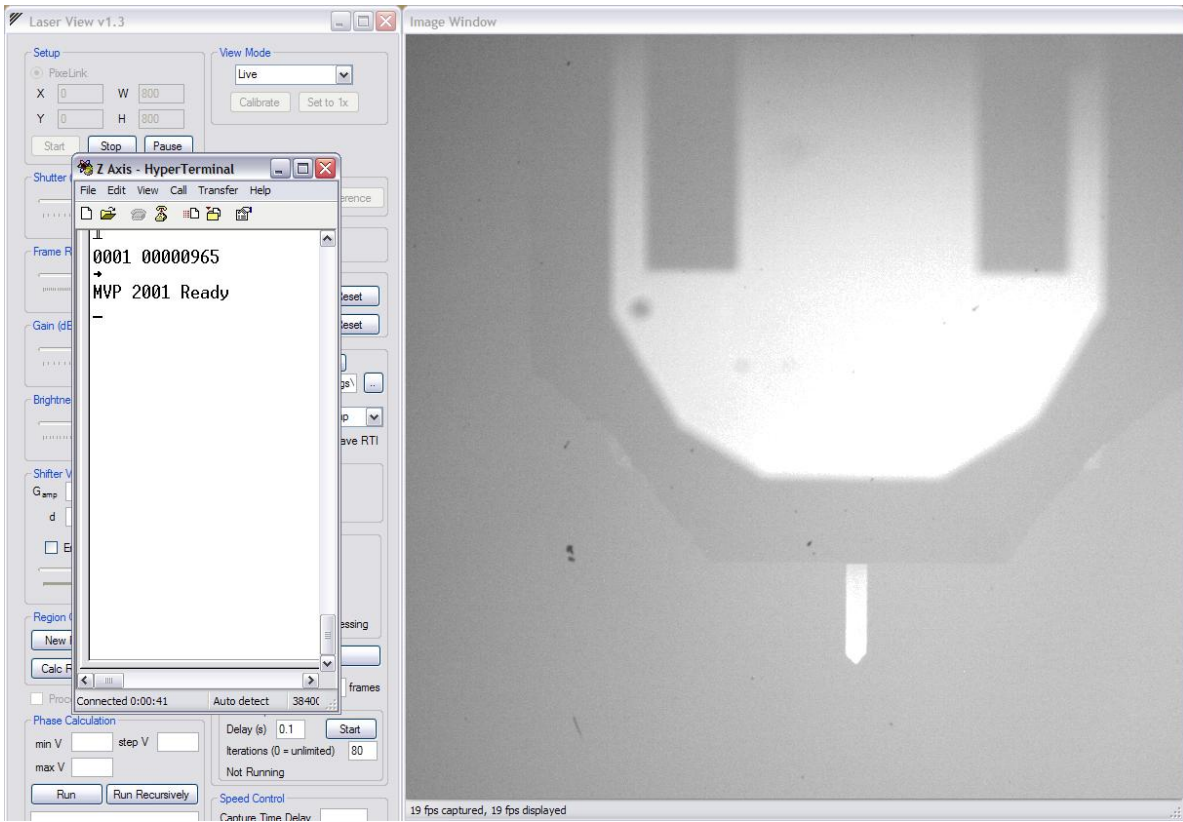


Fig. -. Probe now in focus and goniometers adjusted to have only one fringe, the ideal situation before indenting.

7. Using HyperTerminal on one side of the monitor with the live feed from LaserView visible on the other side of the monitor, move the motorized z-axis downward towards the sample.

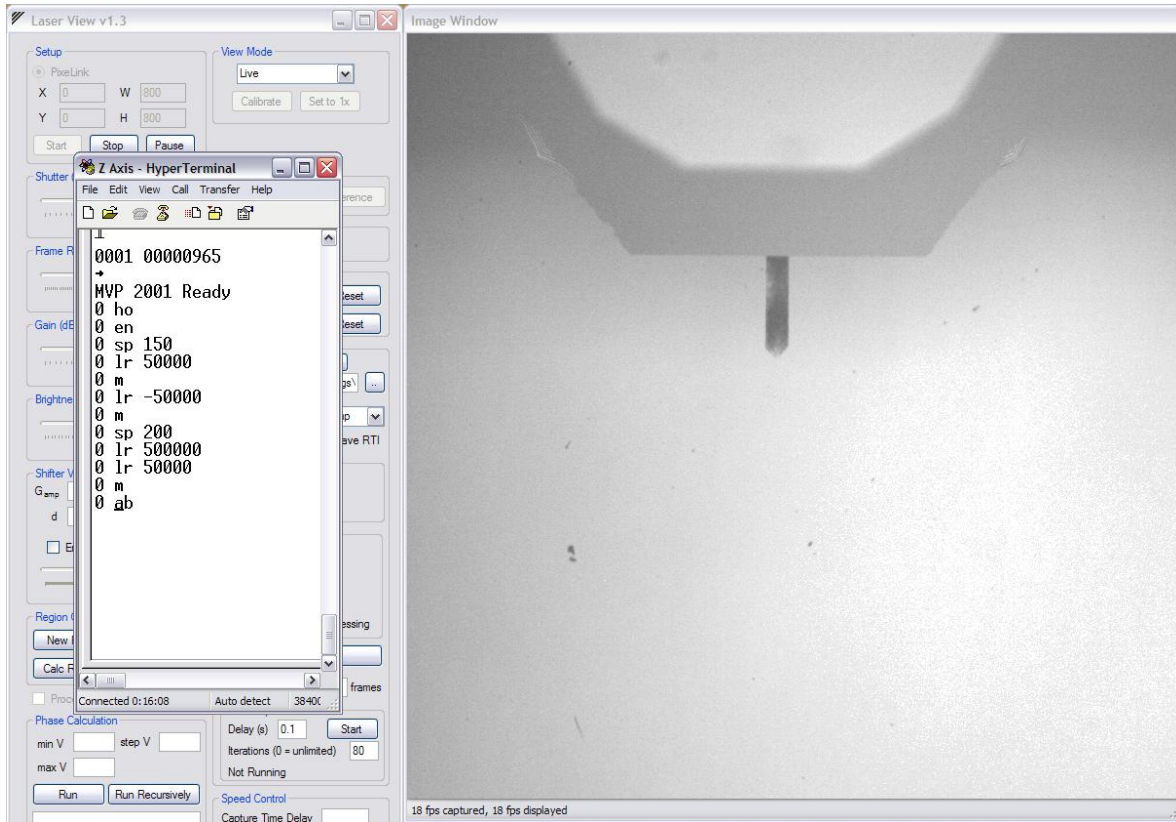


Fig. -. The probe actuated down from the limit stop closer to the sample using the motorized z-axis.

When the assembly is not close to making contact with the sample, the z-axis can be moved in large increments with moderate speed. As the probe gets closer to the sample, interferometric fringes will appear on the probe cantilever. At this point, decrease the step and

Table -. Commands for controlling the motorized Z axis.

Format = Motor# Command Parameter		
Note: Down = Positive and Up = Negative		
Command	Example	Explanation
ho	0 ho	Sets home position of the specified motor.
en	0 en	Enables the specified motor.
lr	0 lr 40000	Move the motor 40000 encoder counts down from current position.
sp	0 sp 100	Sets the speed of the specified motor.
m	0 m	Executes the current motion commands.
ab	0 ab	Aborts current movement (Must be re-enabled afterwards).

speed at which the motorized stage is adjusted. After each increment of motion, refocus the image on the cantilever probe. The commands used for moving the motorized z axis can be found in Table -.

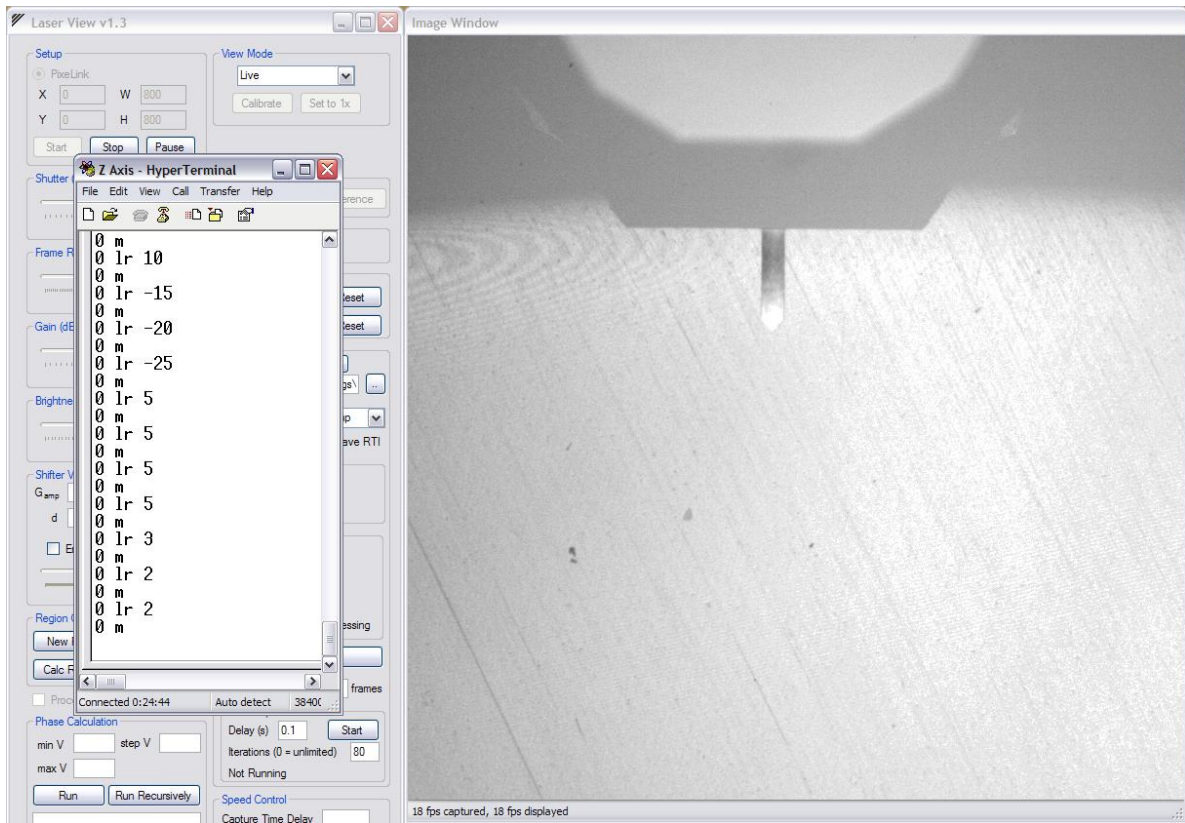


Fig. -. The probe brought into contact with the sample using the motorized z-stage. Smaller steps with the motorized stage were used as the probe got closer to the sample.

Contact with the sample will be shown by an increase in the number of fringes that appear on the probe.

- a. In HyperTerminal, a positive displacement value brings the probe closer to the sample, whereas a negative displacement value moves the probe further away from the sample.

8. After contact has been made with the motorized z-stage, and the stage has been moved upwards until the probe is no longer in contact with the sample. Using the piezoactuator, increase the piezo voltage in steps. For this experiment, the piezo was actuated in increments of 200 nanometers.
9. In LaserView, turn on reference mode. The shifter voltage should be set to 5.52V.

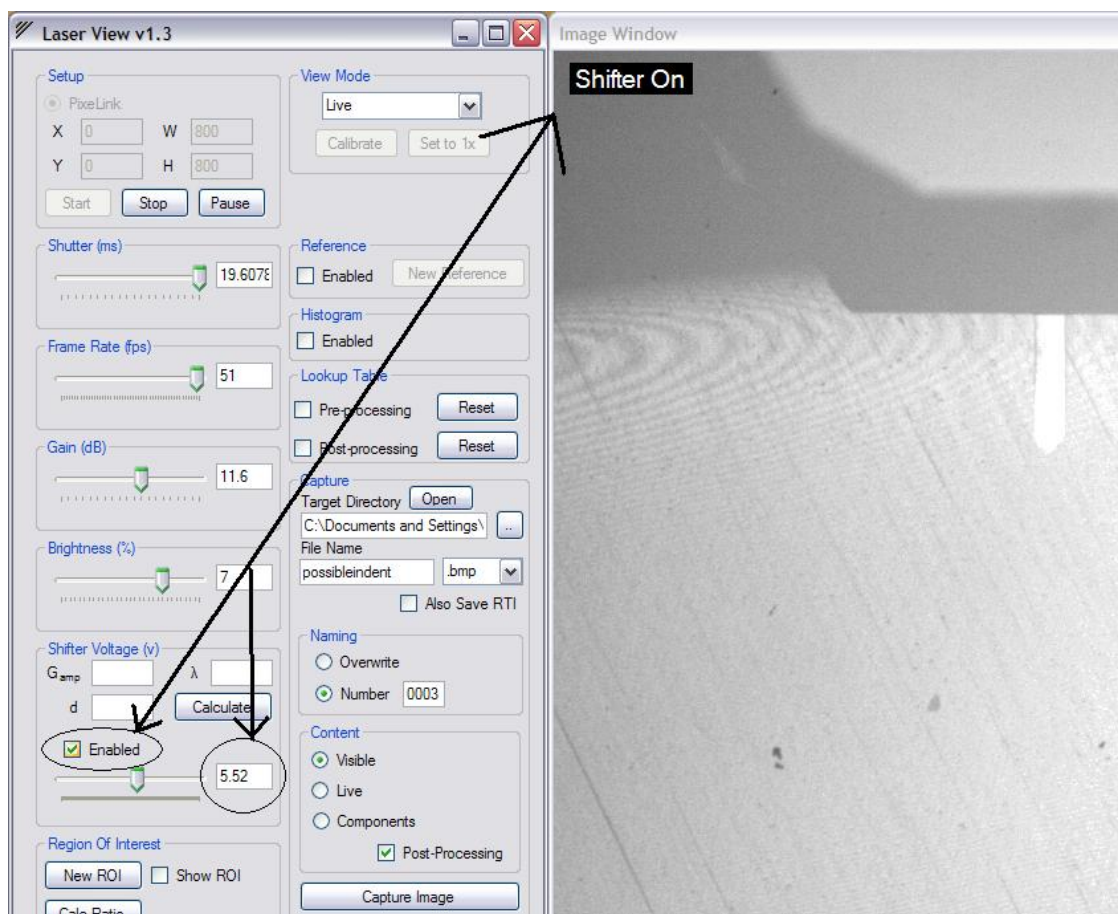


Fig. -. Shifter voltage enabled and set to 5.52 volts; Also indicated in the image window.

From the drop-down menu, select the “optical phase plus 180” mode.

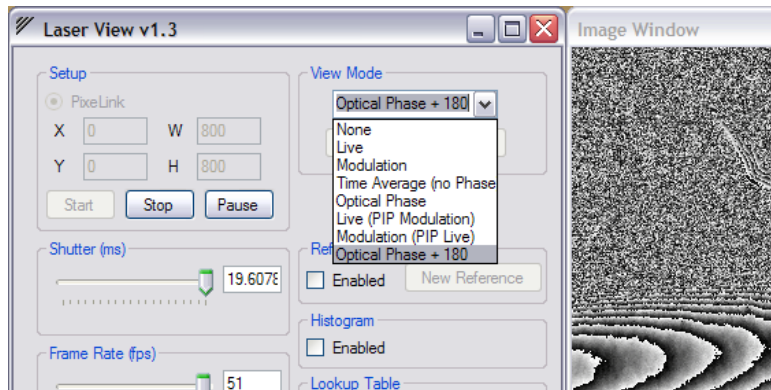


Fig. -. In the View Mode, the optical phase + 180 is select to see the wrapped phase.

Once in reference mode, select the “new reference” button. This will provide the image upon which all future measurements will be compared. Note: once a reference has been selected, do not refocus the interferometer.

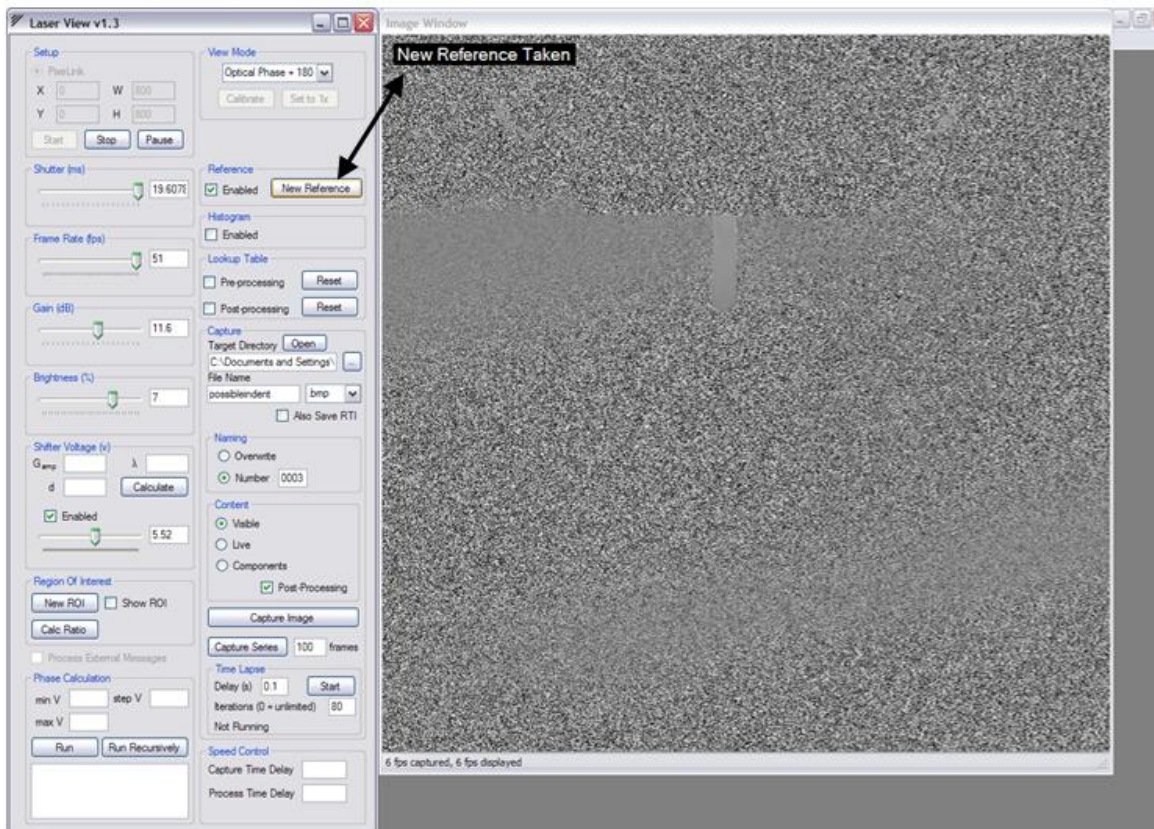


Fig. -. Enable the reference and take a new reference. The image window will also indicate that a new reference was taken.

10. Set the filename of each image and the location to which the file will be saved. Also, be sure to click the number bubble in the naming section so the files will be added in sequential order.



Fig. -. Before starting indentation and capturing pictures, indicate the directory to store the images and file name.

11. Now that Laser View is setup and the probe tip is in contact with the sample, the piezo will be actuated to perform the indenting. A capture of the probe with only one solid fringe should be taken now before indenting to have a starting reference and to be used as the first image for the analysis. If the probe is not in contact with the sample as of yet, but is extremely close, the piezo can be used to step the probe closer until contact. The

piezo can be actuated until more fringes appear, but at that point it is important to record the strain gauge reading and capture the second picture in the series.

12. Once contact occurs and the picture is captured, the piezo will be stepped in 200 nm and the strain gauge measurement will be recorded (used later in the analysis) and the next picture of the first step will be captured. This process of stepping 200 nm, recording the strain gauge value, and capturing a picture of each step will continue for 2.2 microns. This distance of 2.2 microns is based on the model shown in Section .

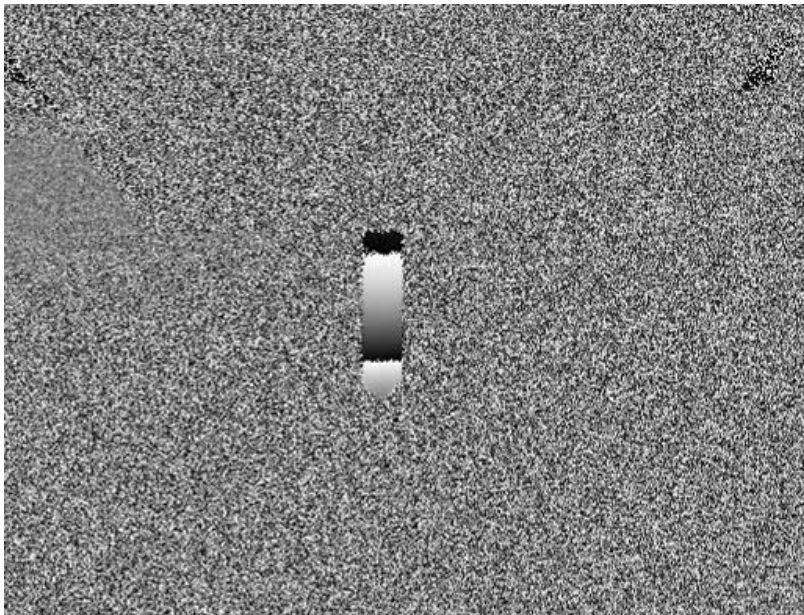


Fig. -. The cantilever is in contact and actuation has begun.

13. After actuating the piezo 2.2 microns, the same process of actuating 200 nm, recording the strain gauge reading, and capturing a picture will continue, but now in reverse to retract the probe from the sample. Before retracting the probe, it is important to change the name of the capture file name. Once the name is changed retract the piezo in 200 nm steps and perform the process backwards.

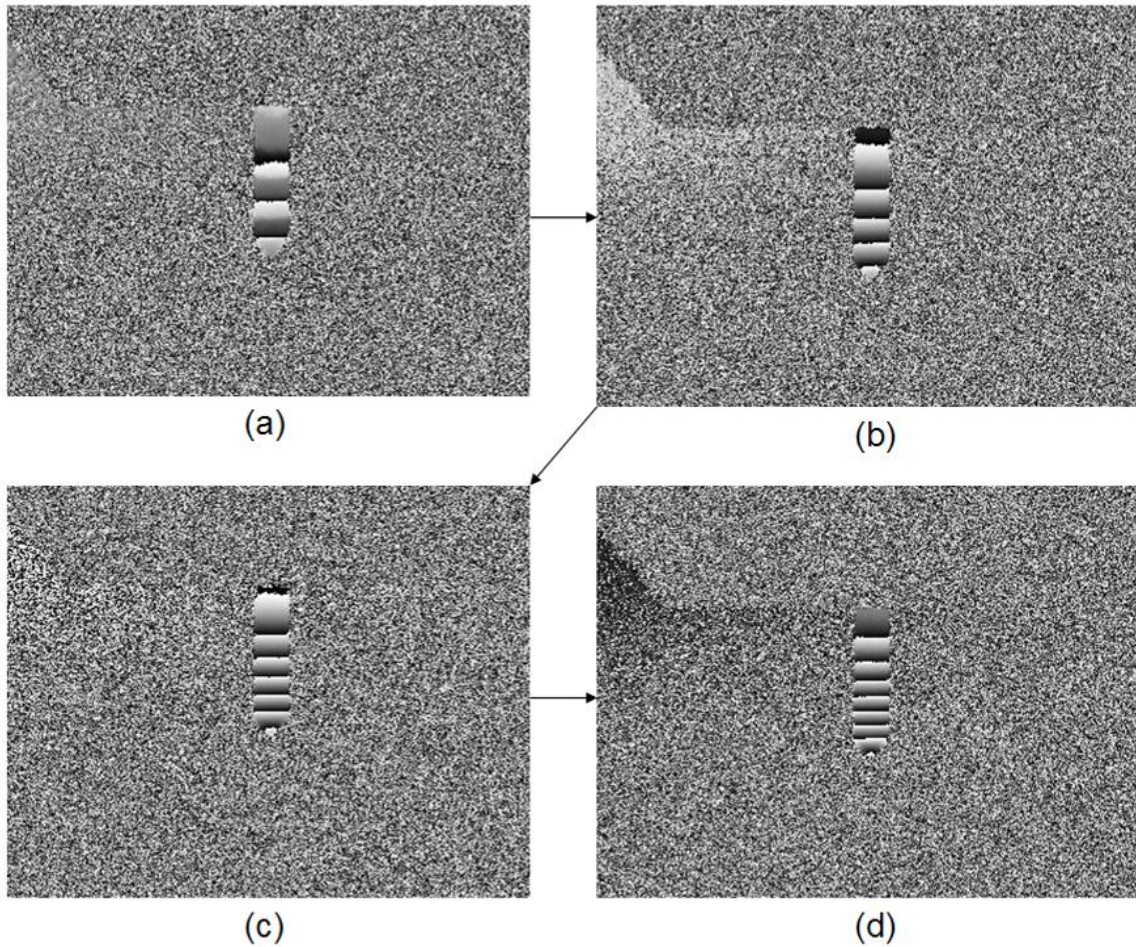


Fig. -. This picture shows the indentation process. As the piezo is actuated the fringes increase.

14. When the piezo is retracted from the sample, use the motorized z-stage to back the piezo, cantilever, and probe out away from the sample. Once it is backed away from the sample, the micrometers on the x and y-stages can be moved to set up a new position to indent or the module could be back out and the sample can be removed to be inspected by a microscope or a Scanning Electron Microscope (SEM).

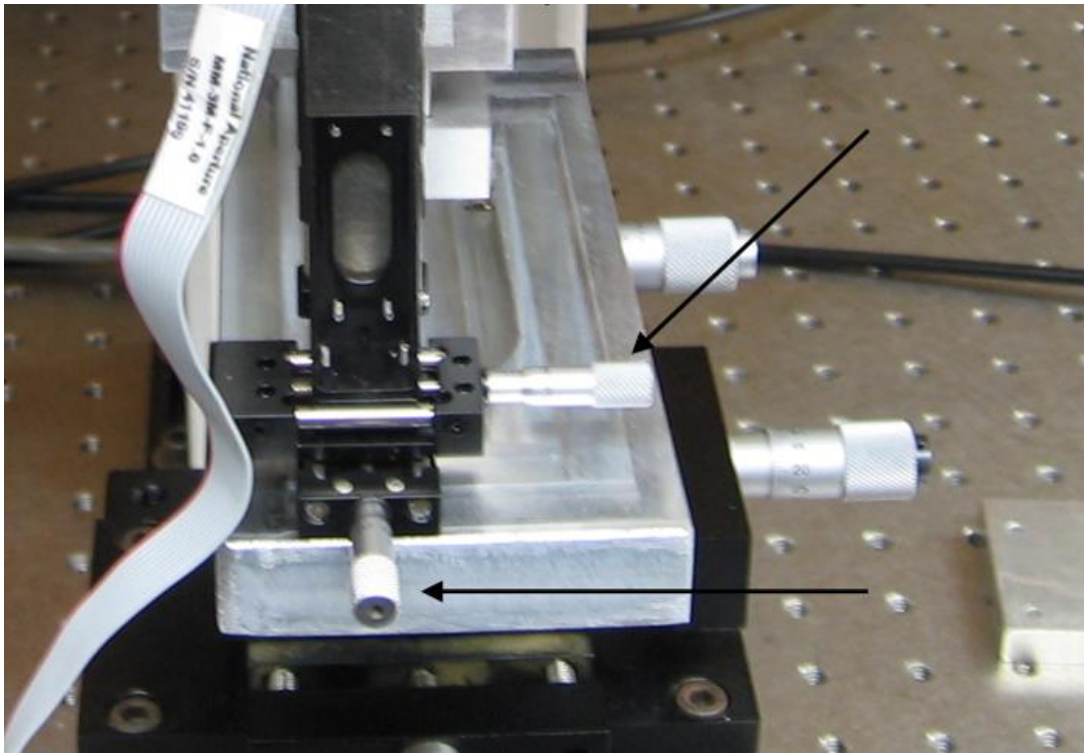


Fig. -. Micrometers used to move the piezo, cantilever, and probe to different positions to indent.

5. Procedure for Data Analysis

Once the indentation has been performed and all the images have been acquired, they must be processed in order to yield the force displacement curve. This chapter outlines the steps necessary to process the image data acquired and extract the results.

1. To perform the data analysis, Python version 2.5 should be installed on the computer running the analysis. This software can be obtained for free at the following address:

<http://www.python.org/download/releases/2.5/>

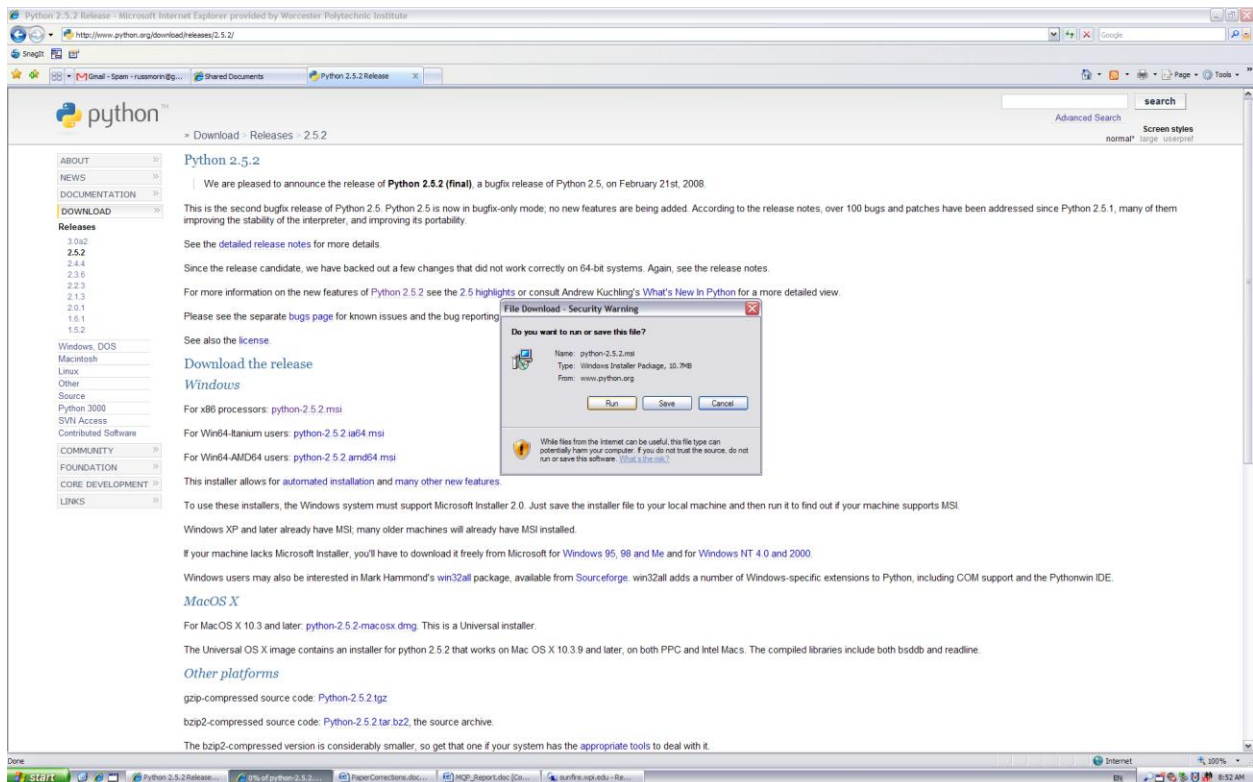


Fig. -. Downloading Python version 2.5 (Python).

2. Along with Python, there are addons that must be installed for the analysis program to function properly. The packages Matplotlib and NumPy can be obtained for free

at the following websites respectively: <http://sourceforge.net/projects/matplotlib> and http://sourceforge.net/project/showfiles.php?group_id=1369&package_id=175103.

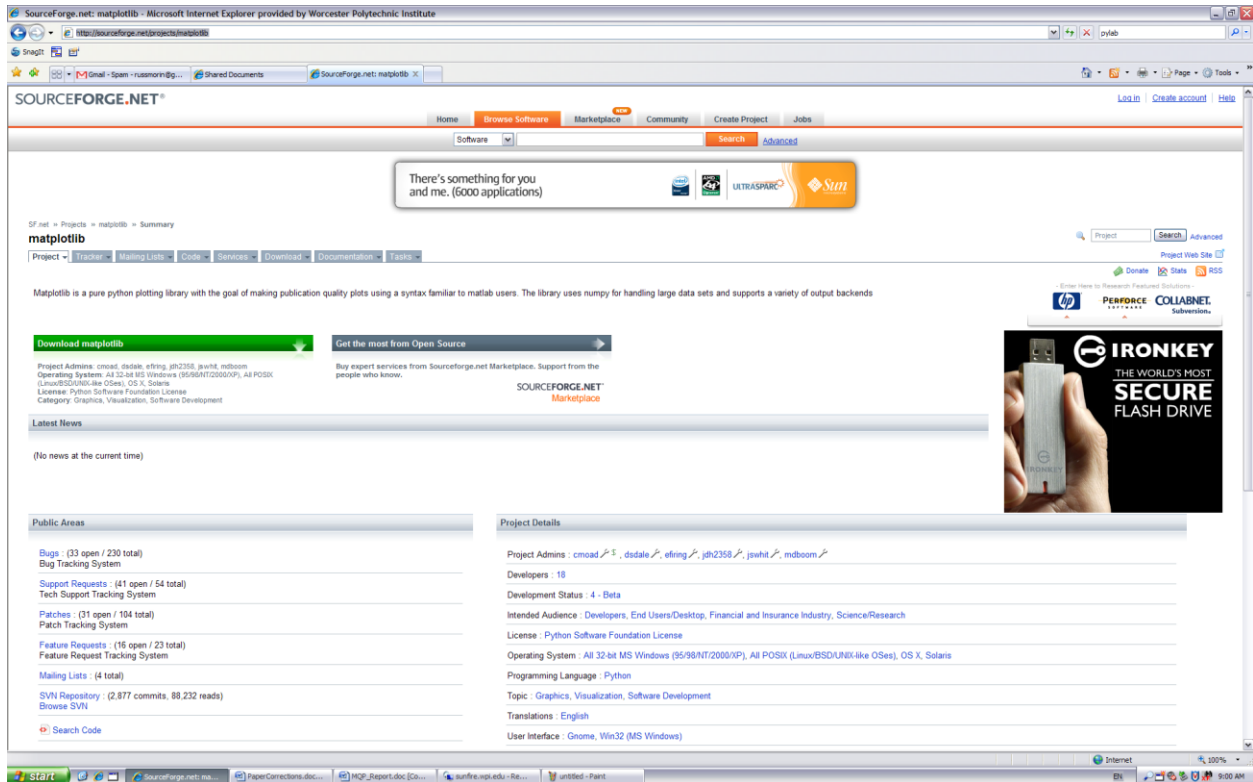


Fig. -. Download screen for MatPlotLib (Matplotlib, 2008).

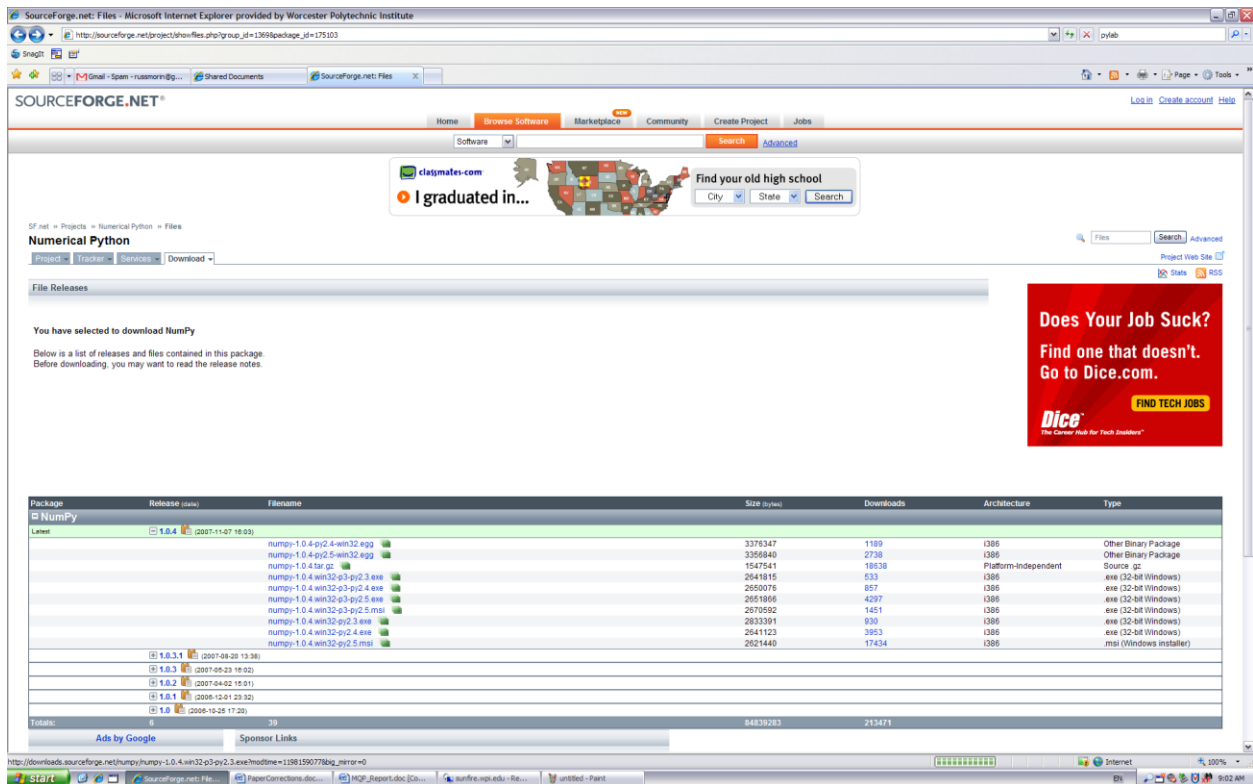


Fig. -. Download page for NumPy (NumPy, 2008).

3. The last software application that must be installed or be accessible to perform the analysis is ANSYS. At WPI, ANSYS can be accessed by remotely connecting to sunfire.wpi.edu. ANSYS is a powerful finite element analysis program that is used to solve the inverse problem.

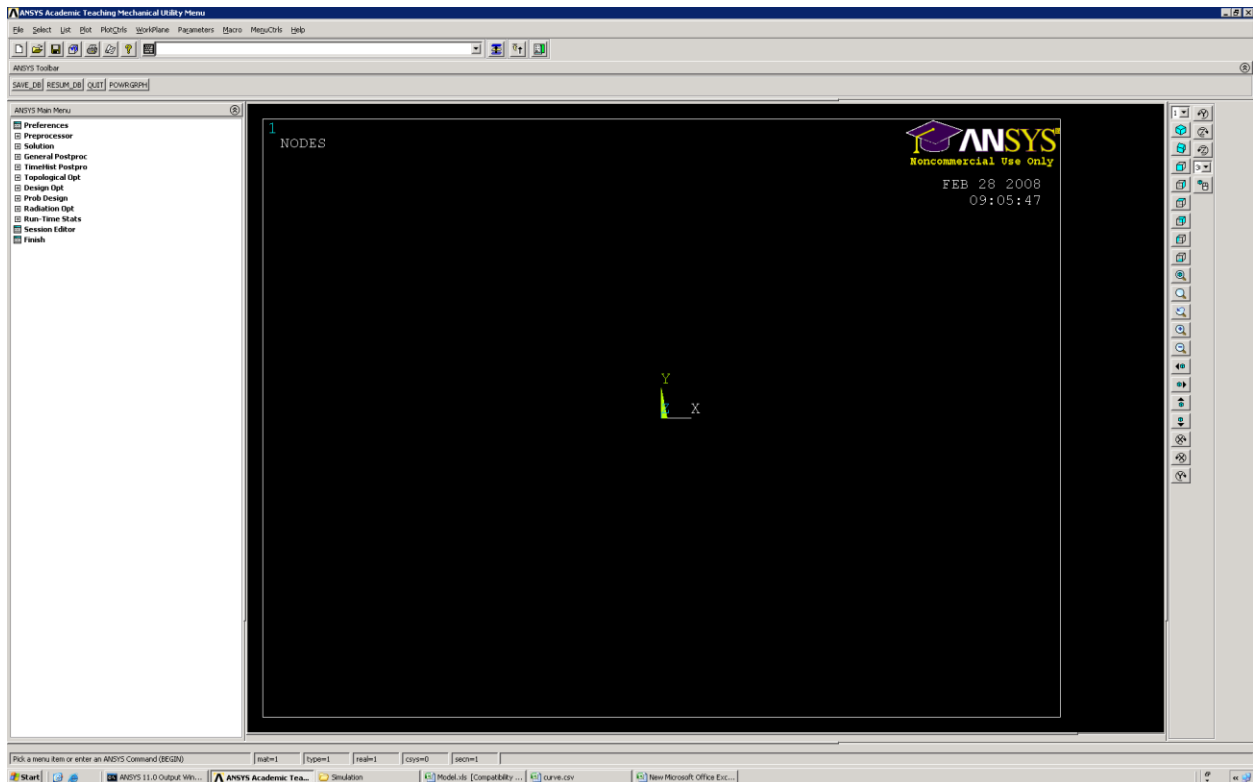


Fig. -. Finite element program, ANSYS, used for analysis of image data, (ANSYS, 2008).

4. To begin the analysis, copy the analysis Python scripts into the directory containing the images acquired with the interferometer. The files required are:
IndentationImageProcessor.py, ANSYSMacroCreator.py, ConverttoCSV.py, and
RPolyFit.py. The source code for all of these files is located in Appendix C.

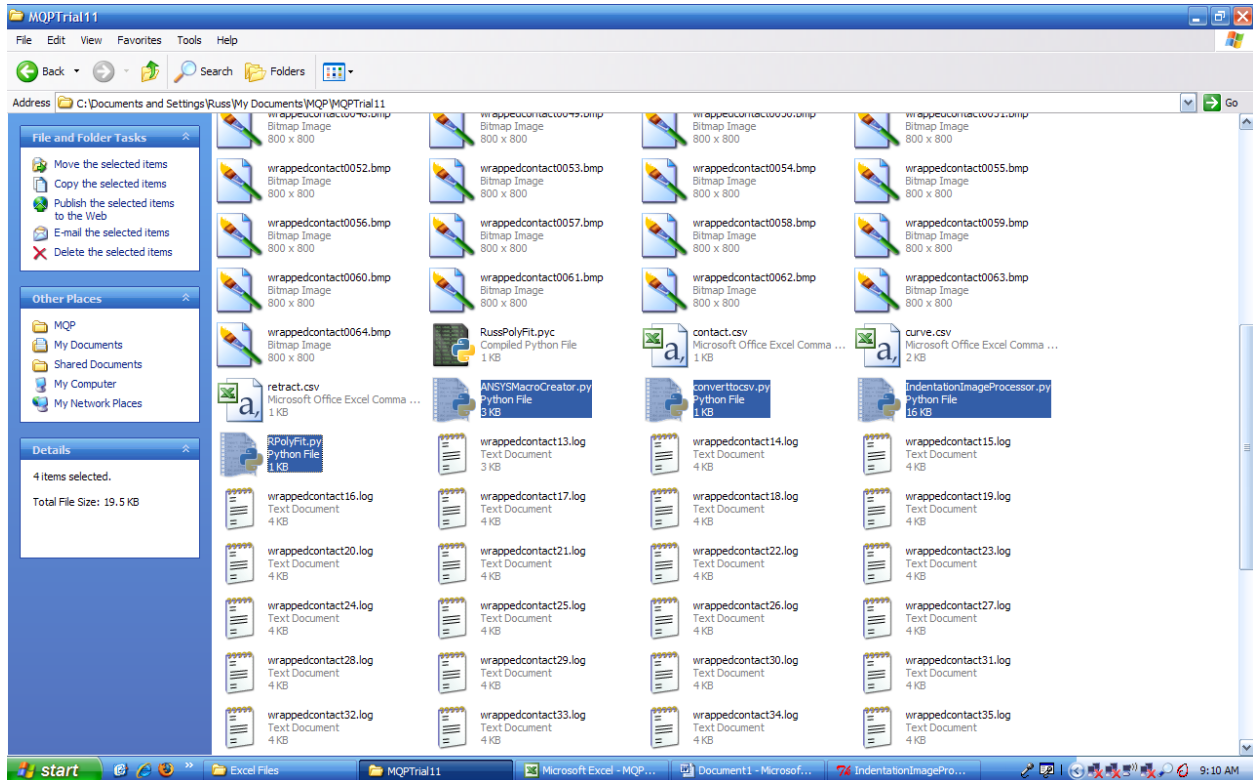


Fig. -. Pasting the analysis files into the image directory.

5. Before beginning the analysis, the piezo displacement data that was written down during the indentation must be input to the computer so it can be read by the program. Two files should be created, one for indentation and one for retraction. The files should look like the following figure and be saved with the file extension, “*.csv”, which denotes “comma separated values”.

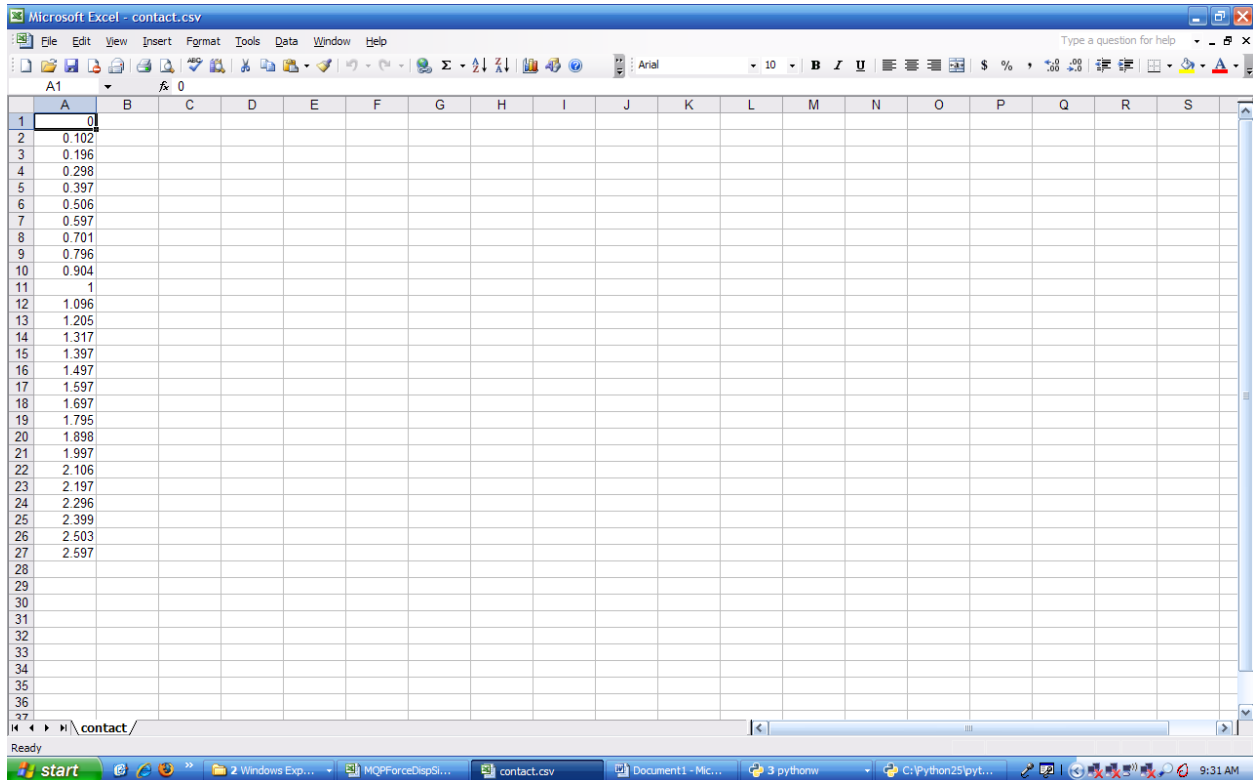


Fig. -. Creation of CSV file with piezo displacement data.

- Open the first image being processed with Microsoft Paint. In the lower right hand corner, the image coordinates of the current mouse position are displayed. Using this coordinate display, determine the coordinates of a rectangle that covers the center of the beam area as is shown in the following figure. The rectangle should only be about 3 pixels wide and should cover the entire length of the beam.



Fig. -. Obtaining the beam rectangle from MS Paint using cursor position.

7. Now that the image rectangle coordinates have been obtained, they must be entered into the program. Right click on the file `IndentationImageProcessor.py` and click “Edit with IDLE”. Search for the line `box = (“`. When you have found the line, edit the coordinates that are in the program to reflect the coordinates of the beam in your images as is shown in the following figure. Save the program and exit.


```

numelements = 10

lightwavelength = 0.630 # micrometers

halfwavelength = lightwavelength / 2.0 # n

heightstep = halfwavelength / 256.0 # micr

# Beam rectangle box
box = (370, 230, 372, 333)

```

Fig. -. Editing the beam rectangle in the program.

8. To begin the image analysis, double click on the program

IndentationImageProcessor.py. A screen like the following figure will appear.

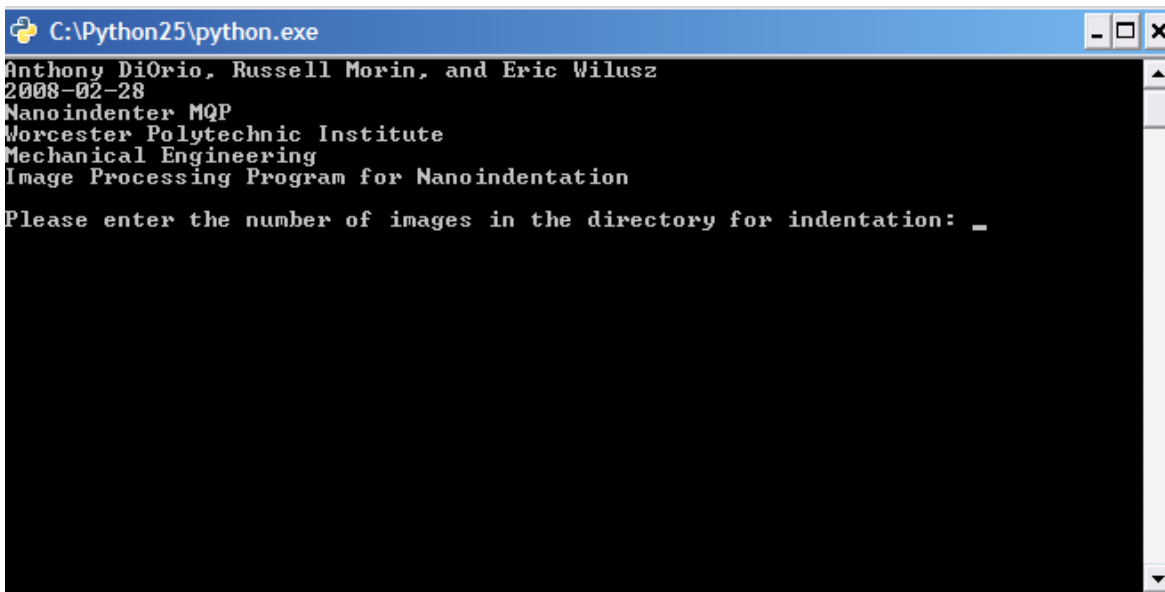
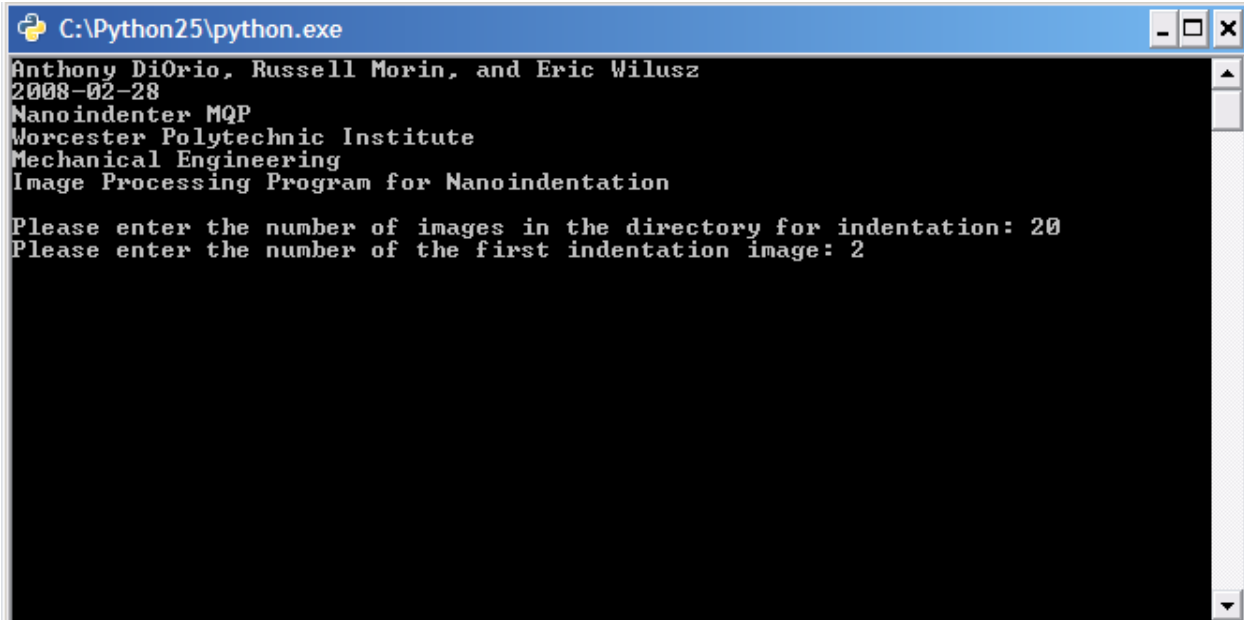


Fig. -. Initial screen for image processing program.

9. At this screen you are prompted to enter the number of images for the indentation.

This is only for the indentation part of the processing. The program will later ask for information about the retraction portion. The next screen will look like the following

figure and will ask for the first image number. The images numbers do not need to start at 1, as long as they are sequential.

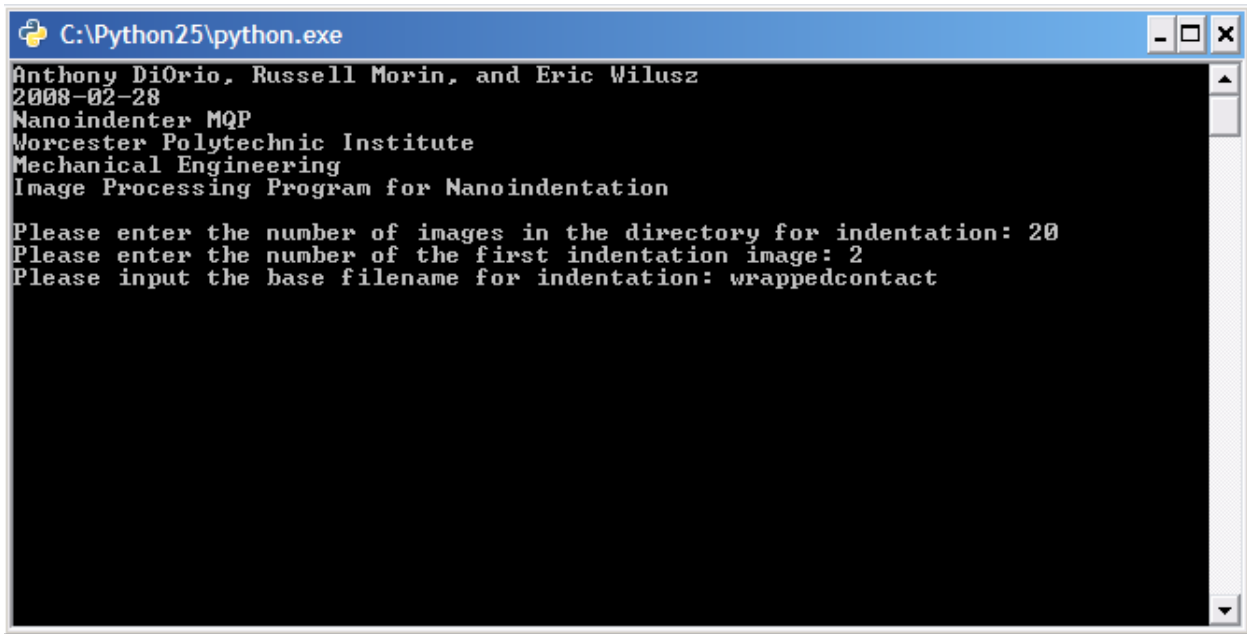


```
C:\Python25\python.exe
Anthony DiOrio, Russell Morin, and Eric Wilusz
2008-02-28
Nanoindenter MQP
Worcester Polytechnic Institute
Mechanical Engineering
Image Processing Program for Nanoindentation

Please enter the number of images in the directory for indentation: 20
Please enter the number of the first indentation image: 2
```

Fig. -. Entering initial image number to the processor.

10. The next prompt will ask for the base filename of the indentation images. This is the filename without any numbers or file extensions on the end. For the purposes of this program, all the images are assumed to be bitmap or “*.bmp” format. All of the indentation experiments performed so far have used the filenames “wrappedcontact” and “wrappedretract” to denote the image types.

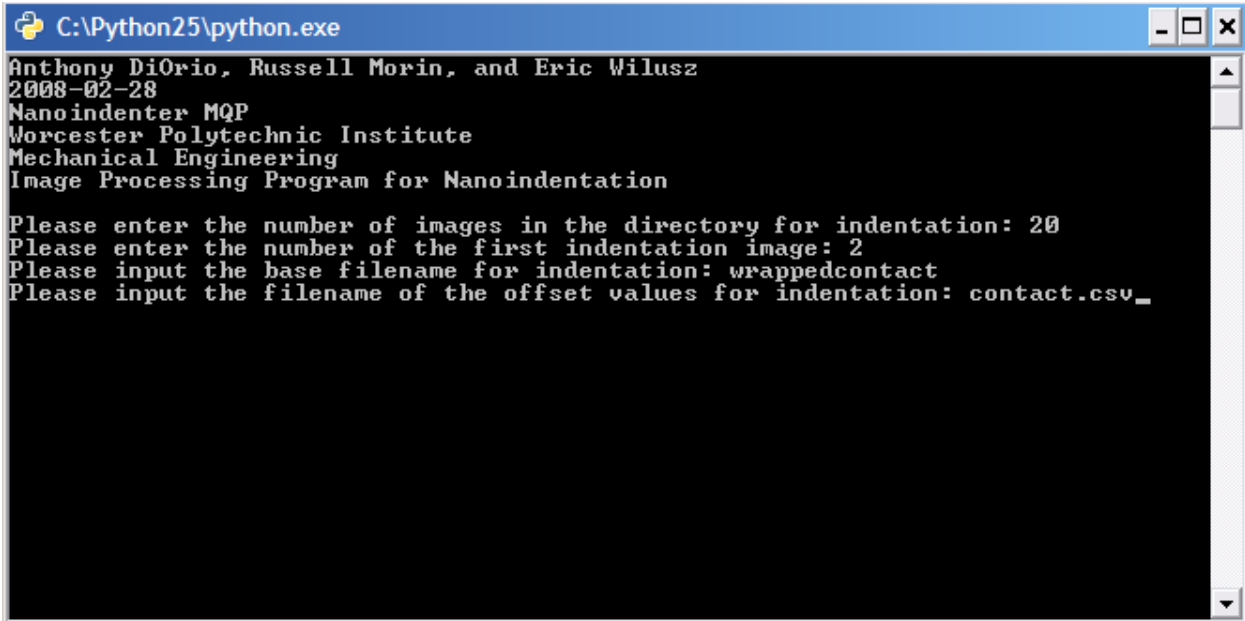


```
C:\Python25\python.exe
Anthony DiOrio, Russell Morin, and Eric Wilusz
2008-02-28
Nanoindenter MQP
Worcester Polytechnic Institute
Mechanical Engineering
Image Processing Program for Nanoindentation

Please enter the number of images in the directory for indentation: 20
Please enter the number of the first indentation image: 2
Please input the base filename for indentation: wrappedcontact
```

Fig. -. Prompt to enter the base filename into the processor.

11. The final prompt for the indentation will ask for the filename of the file where all the piezo displacement data is stored. This file was created in step 5. For this prompt, it is important that the entire filename is entered to the program.

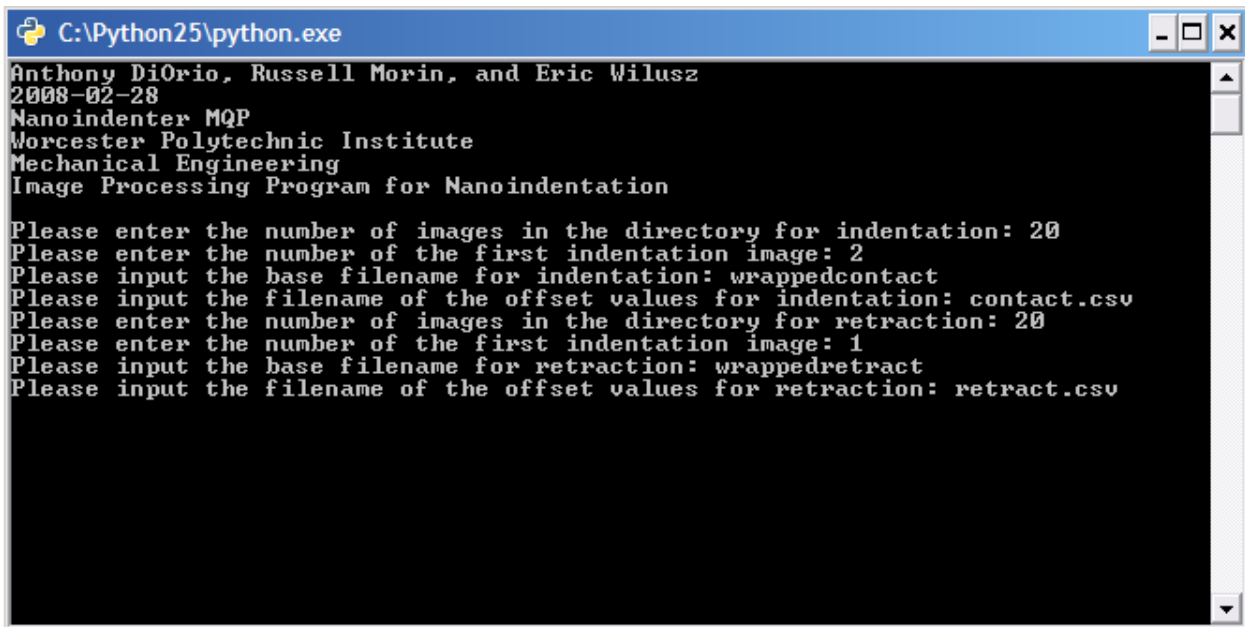


```
C:\Python25\python.exe
Anthony DiOrio, Russell Morin, and Eric Wilusz
2008-02-28
Nanoindenter MQP
Worcester Polytechnic Institute
Mechanical Engineering
Image Processing Program for Nanoindentation

Please enter the number of images in the directory for indentation: 20
Please enter the number of the first indentation image: 2
Please input the base filename for indentation: wrappedcontact
Please input the filename of the offset values for indentation: contact.csv_
```

Fig. -. Prompt for the piezo displacement data filename.

12. The next four prompts will be similar to the first four, but they will deal with the information about the images for retraction. When all the prompts have been answered, the screen should look something like the following figure.



```
C:\Python25\python.exe
Anthony DiOrio, Russell Morin, and Eric Wilusz
2008-02-28
Nanoindenter MQP
Worcester Polytechnic Institute
Mechanical Engineering
Image Processing Program for Nanoindentation

Please enter the number of images in the directory for indentation: 20
Please enter the number of the first indentation image: 2
Please input the base filename for indentation: wrappedcontact
Please input the filename of the offset values for indentation: contact.csv
Please enter the number of images in the directory for retraction: 20
Please enter the number of the first indentation image: 1
Please input the base filename for retraction: wrappedretract
Please input the filename of the offset values for retraction: retract.csv
```

Fig. -. All prompts answered before beginning image processing.

13. When the final prompt has been answered, the program will open each image and process the data within them. First, it unwraps the phase along the image beam length. Then it offsets the data so it is aligned with the origin and rotates it so it has a slope of zero at the origin. A best fit cubic curve through all the points is created using the least squares curve fitting method. Finally, the data is offset by the piezo displacement data to produce the beam profile for that image.
14. After the program has all of the deflection information for the beam processed, it creates an ANSYS script file that is used to process the data. These script files have the file extension “*.log” and have the same filename as the image that they were created from. An example of these log files can be seen in the figure below.

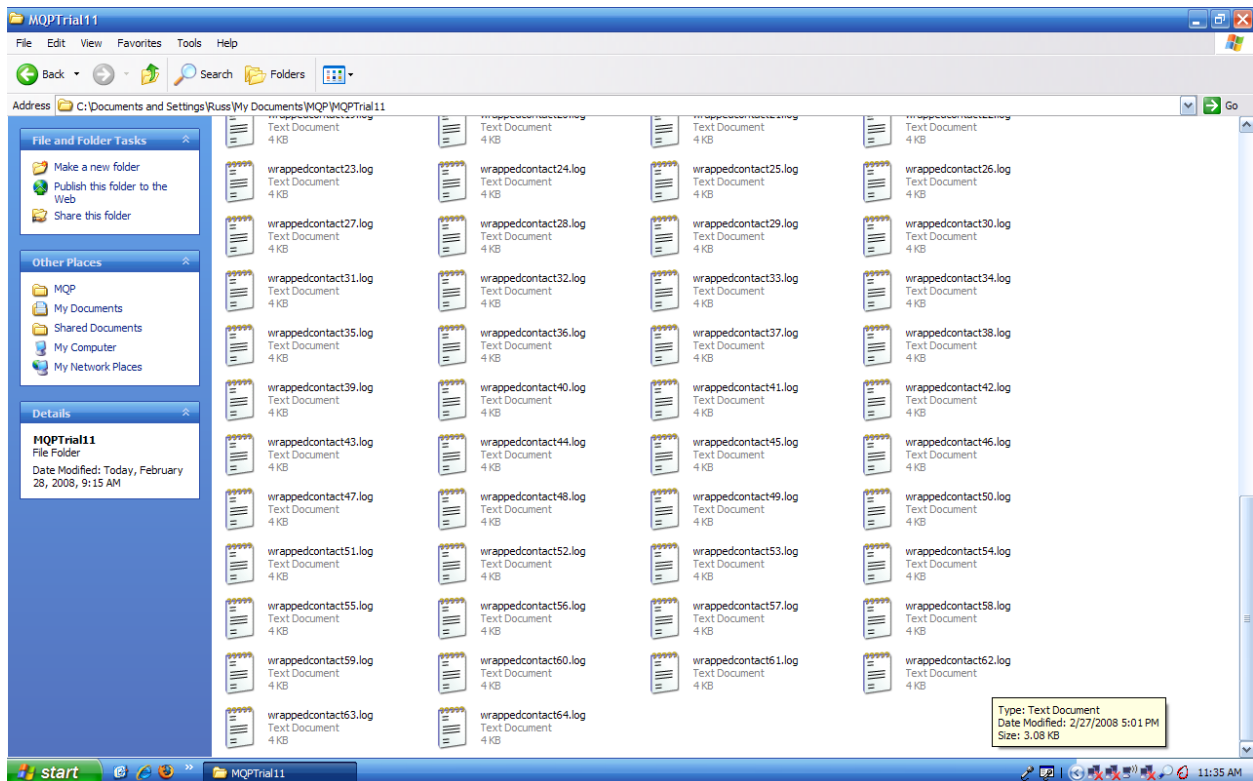


Fig. -. ANSYS log files created by the image processor.

15. The next step involves creating another ANSYS script file that will automatically open all the files just created, solve them, and extract the data. This is done by double clicking the program ANSYSMacroCreator.py. A screen will appear like the one in the following figure.

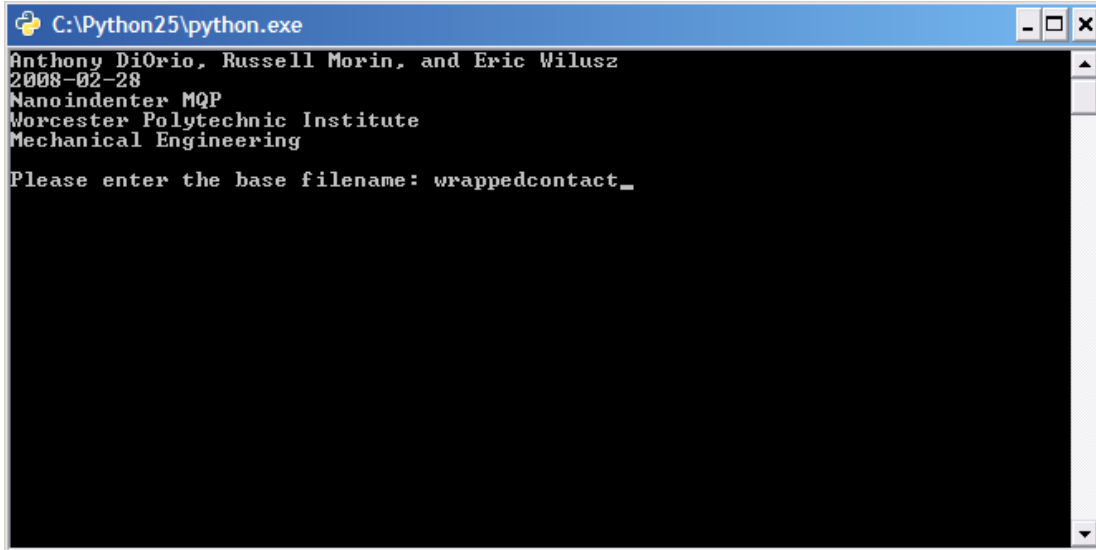
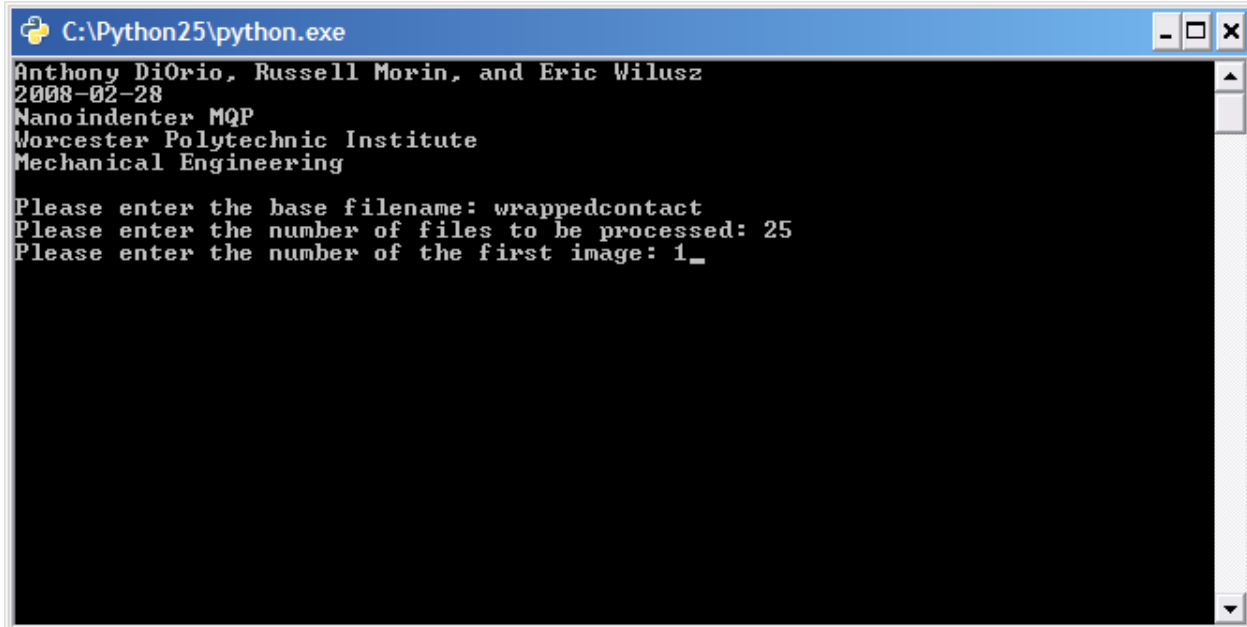


Fig. -. Prompt for base filename in ANSYSMacroCreator. py.

16. The program will ask for the base filename, the number of files to be processed, and the image number of the first image as shown in the figure below. It will then create an ANSYS script file with the name "basefilename.log". The program should be run once for the indentation images and once for the retraction images.



```
C:\Python25\python.exe
Anthony DiOrio, Russell Morin, and Eric Wilusz
2008-02-28
Nanoindenter MQP
Worcester Polytechnic Institute
Mechanical Engineering

Please enter the base filename: wrappedcontact
Please enter the number of files to be processed: 25
Please enter the number of the first image: 1_
```

Fig. -. Prompts for number of files to be processed by ANSYS.

17. Copy all of the “*.log” files to a computer with ANSYS installed. Open ANSYS and click “File->Read Input From...” as seen below. Make sure that the writer.mac file is in the same directory as the “*.log” files.

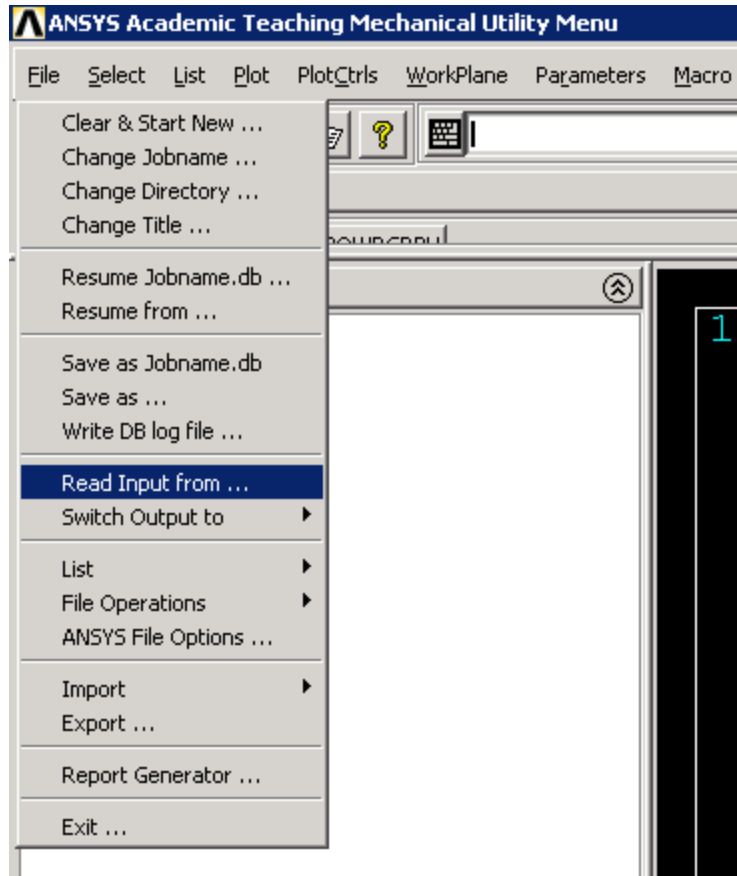


Fig. -. Click File->Read Input From. . .

18. This will bring up a dialog box like the one in the following figure. Navigate through the dialog to find the location of the “*.log” files. Click on the log file that has no numbers at the end to begin running the processing.

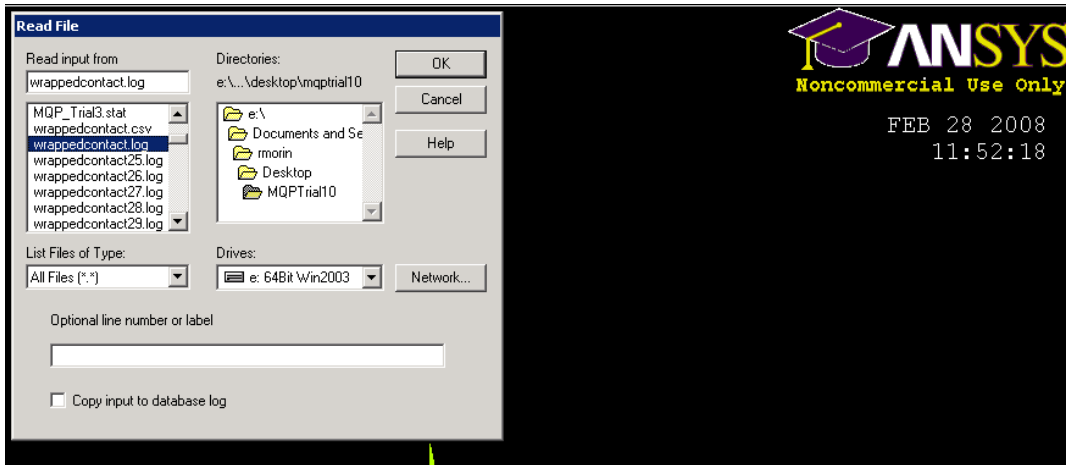


Fig. -. Open the log files to begin the ANSYS processing.

19. ANSYS will automatically open and solve all of the “*.log” files that were created by the image processor. It will output a file called “basefilename.csv” with the force and displacement data obtained from ANSYS. Once ANSYS is completed, run the analysis again for the retraction files.
20. The files output by ANSYS need to be processed one final time to make them usable. Double click on the program ConverttoCSV.py. A screen like the one in the following figure will appear. Enter the base filename and the program will process the data appropriately. Run this for both “*.csv” files.

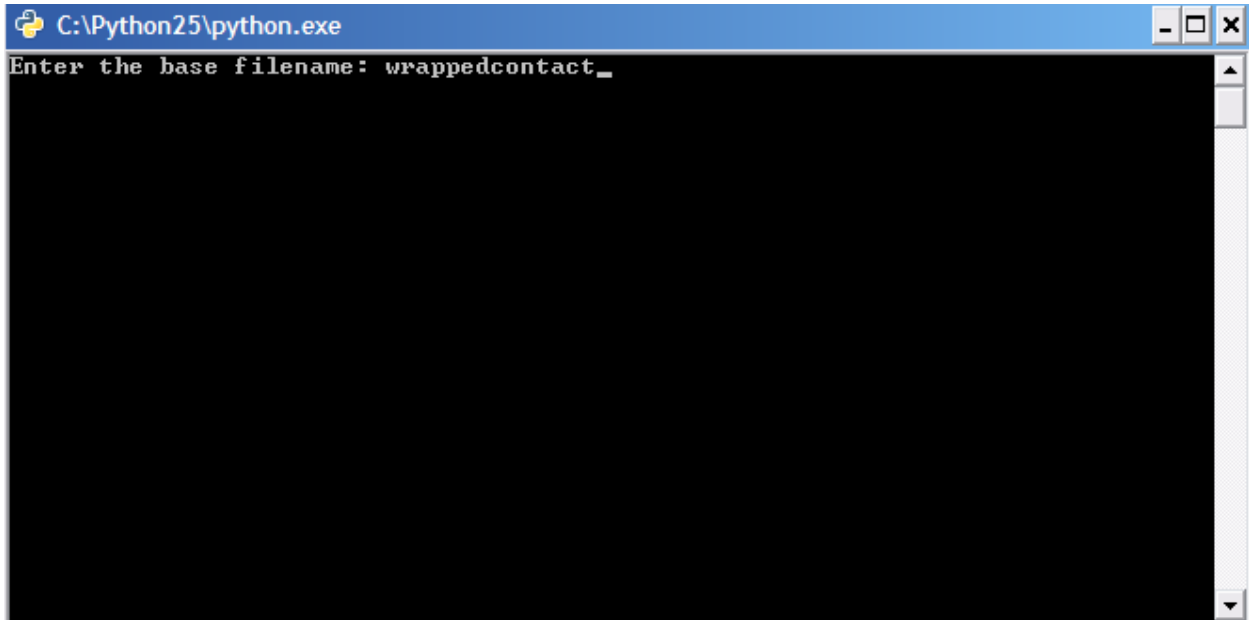


Fig. -. Program to convert to "*.csv".

21. Now the files are ready to be plotted. If everything worked properly with ANSYS, they should trace curves that look similar to those shown in the model of indentation.

6. Mechanics of Indentation

In the field of modern materials science, hardness tests are one of the most common tests performed on samples or components. As told by Norton (2006), hardness tests employ the technique of macroscale indentation:

Hardness is most often measured on one of three scales: Brinell, Rockwell, or Vickers. These hardness tests all involve the forced impression of a small probe into the surface of the material being tested. The Brinell test uses a 10-mm hardened steel or tungsten-carbide ball impressed with either a 500- or 3000-kg load depending on the range of hardness of the material. The diameter of the resulting indent is measured under a microscope and used to calculate the Brinell hardness number, which has the units of kg/mm^2 . The Vickers test uses a diamond-pyramid indenter and measures the width of the indent under the microscope. The Rockwell test uses a 1/16-in ball or a 120° cone-shaped diamond indenter and measures the depth of penetration.

All forms of macroscale indentation are characterized by applying a specified load and measuring an aspect of the resulting indentation once the test has been completed.

Nanoindentation, by comparison, is a process characterized by continuous monitoring of force and displacement throughout the entire indentation event. Thus, the advantage of nanoindentation becomes apparent; by recording force and displacement continually, several mechanical properties of the sample can be determined, including elastic modulus, Poisson's ratio, and hardness. Macroscale indentation, however, provides only hardness data, as there is only a single data point.

Nanoindentation is a process by which two rigid bodies are brought into contact, and deformation occurs at the point of contact. An indenter tip of known geometrical properties is used to physically deform a sample in a predictable manner while the force applied to the indenter tip and the displacement the indenter tip has traveled into the sample are recorded continuously. As the indenter moves further into the sample, the area of the indentation increases, as does the force.

In the strictest sense of the term, “indentation” refers only to the action during which the tip starts entirely outside the sample, and is then brought into contact with the surface of the sample before the indenter moves to its force or displacement limit, depending on design. While the indenter tip is being retracted from its maximum depth into the sample, restorative elastic forces in the sample work to return the sample to its original crystalline formation. For this reason, indentation data is often collected as the tip is retracted, to determine the qualities of the elastic restoration forces of the sample.

Once the indentation has been performed, the force and displacement data are plotted on a curve of force versus displacement. These plots often exhibit hysteresis, however, as the materials from which the samples are made rarely return to their exact form after indentation. A plot of the stress gradient throughout the entire sample can also be determined from the force-displacement data. From the force-displacement plots, the elastic modulus, Poisson’s ratio, and hardness of the sample can be determined.

The deformation in the sample during the indentation can be separated into three distinct regimes: elastic deformation, yield, and plastic deformation. In the elastic regime, the contact mechanics are governed by the Hertzian contact equations; at yield and in the plastic regime, the contact mechanics are governed by the work of Johnson, and Field, Bell, and Swain, whose work was also based on that of Tresca and von Mises.

At the beginning of the indentation process, the indenter tip is brought into contact with the sample. As the indentation process begins, the deformations in the sample in the elastic regime are characterized by the Hertzian contact stress theory.

[Hertz] first made the hypothesis that the contact area is, in general, elliptical. . . He then introduced the simplification that, for the purpose of calculating the local deformations, each body can be regarded as an elastic half-space loaded over a small elliptical region of its plane surface. By this simplification, generally

followed in contact stress theory, the highly concentrated contact stresses are treated separately from the general distribution of stress in the two bodies which arises from their shape and the way they are supported. In addition, the well developed methods for solving boundary-value problems for the elastic half-space are available for the solution of contact problems (Johnson, 1985).

6.1. Elastic Regime

Before a full understanding of Hertz's work can be obtained, one must fully understand Hooke's Law, as the work of Hertz is based on the work done by Hooke two hundred years earlier. As can be determined experimentally, "... the stress-strain diagrams for most engineering materials exhibit a *linear relationship (sic)* between stress and strain within the elastic region. Consequently, an increase in stress causes a proportionate increase in stress. This fact was discovered by Robert Hooke in 1676 using springs and is known as *Hooke's Law (sic)*" (Hibbeler, 2005). Expressed mathematically, Hooke's Law can be written

$$\sigma = E\varepsilon \quad (.)$$

where σ is defined as the stress, E is Young's modulus, and ε is the strain.

Although Hertz (1896) begins the explanation of his contact stress theory with the definitions of Kirchhoff's elastic coefficients, the final form is an analogue to Hooke's Law. To begin the derivation of the governing equations of the Hertzian contact stress theory, one must first make an assumption about the geometry at the interface between the indenter and the specimen. Assuming the indenter is of spherical geometry and the specimen is flat, the perimeter of contact between the tip and the sample will be circular. As mentioned above, Hertz (1896) begins with the definitions of K and θ , which are the elastic coefficients defined by Kirchhoff.

The expression for θ is derived from the following expression (a partial derivation follows; the full derivation can be found in Appendix A):

$$\nu = \frac{\theta}{1+2\theta} \quad (.)$$

where ν is Poisson's ratio; algebraic manipulation of the equation yields:

$$\theta = \frac{\nu}{1-2\nu} \quad (.)$$

To derive the expression for K , begin with

$$E = 2K \left(\frac{1+3\theta}{1+2\theta} \right) \quad (.)$$

and arrive at the expression

$$K = \frac{E}{2(1+\nu)} \quad (.)$$

Hertz then uses a contraction to simplify the mathematics of future steps, defining

$$\ell = \frac{2(1+\theta)}{K(1+2\theta)} \quad (.)$$

If K and θ are substituted and the expression is simplified, the expression for ℓ becomes

$$\ell = \frac{4(1-\nu^2)}{E} \quad (.)$$

Hertz then introduces a function which equates electric potential and pressure, which is defined as the derivative of P with respect to z :

$$\frac{dP}{dz} = \frac{-3p}{8\pi} \cdot z \cdot \int_u^{\infty} \frac{1}{\lambda \cdot \sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} d\lambda \quad (.)$$

Simplifying the equation above yields

$$\frac{dP}{dz} = \frac{-3p}{8\pi \cdot a \cdot b} \cdot \frac{z^2 \left(\frac{du}{dz}\right)}{u\sqrt{u}}, \quad (.)$$

where p is defined as force. Substituting dP/dz into the equation of Z_z yields

$$Z_z = \frac{3p}{2\pi \cdot a \cdot b} \cdot \sqrt{1 - \frac{x^2}{a^2} - \frac{y^2}{b^2}}, \quad (.)$$

for the $z = 0$ plane. To satisfy the conditions that exist inside of the contact surface, a, b, and h must be defined. To accomplish this, a and b are defined as follows:

$$a = \frac{A}{\ell_1 + \ell_2} \cdot \frac{16\pi}{3p} \quad (.)$$

$$b = \frac{B}{\ell_1 + \ell_2} \cdot \frac{16\pi}{3p} \quad (.)$$

Hertz then gives four principal curvatures, denoted by ρ_{11} , ρ_{12} , ρ_{21} , and ρ_{22} . If it is assumed that

$$\rho_{11} = \rho_{12} = \frac{1}{R_1} = \rho_1, \quad \text{and} \quad (.)$$

$$\rho_{21} = \rho_{22} = \frac{1}{R_2} = \rho_2 \quad (.)$$

where R_1 and R_2 are the radii of the spheres in contact, then the area of contact will be circular, meaning that

$$a = b = \sqrt[3]{\frac{3p(\ell_1 + \ell_2)}{16(\rho_1 + \rho_2)}}. \quad (.)$$

From the equations above, the equation for h can be derived:

$$h = \frac{3p(\ell_1 + \ell_2)}{16a} \quad (.)$$

Hertz then introduces the concepts of a composite radius, R , and a composite elastic modulus, E^* . Both derivations are similar to that of the derivation of springs in series through the use of Hooke's Law:

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} = \rho_1 + \rho_2 \quad (.)$$

and

$$\frac{1}{E} = \frac{(1-\nu_1^2)}{E_1} + \frac{(1-\nu_2^2)}{E_2} \quad (.)$$

Recall that

$$\ell_1 = \frac{4(1-\nu_1^2)}{E_1} \quad (.)$$

$$\ell_2 = \frac{4(1-\nu_2^2)}{E_2} \quad (.)$$

Therefore,

$$\ell_1 + \ell_2 = \frac{4}{E^*} \quad (.)$$

This result is then substituted into the integral of the equation Hertz uses to equate pressure and electrical potential, resulting in the following:

$$P = \frac{3p}{16\pi} \int_u^\infty \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} - \frac{3p}{16\pi} \int_u^\infty \frac{x^2}{a^2 + \lambda} \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} - \frac{3p}{16\pi} \int_a^\infty \frac{y^2}{b^2 + \lambda} \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} \quad (.)$$

which can be simplified to

$$P = L - Mx^2 - Ny^2, \quad (.)$$

where

$$L = \frac{h}{l_1 + l_2}, \quad (.)$$

$$M = \frac{A}{l_1 + l_2}, \text{ and} \quad (.)$$

$$N = \frac{B}{l_1 + l_2}. \quad (.)$$

Manipulation of the expression for L results in:

$$\frac{h}{l_1 + l_2} = \frac{3p}{16\pi} \int_u^\infty \frac{d\lambda}{\sqrt{(\lambda^3 + \lambda^2(a^2 + b^2) + \lambda(a^2b^2))}} = \frac{3p}{16\pi} \int_u^\infty \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}}. \quad (.)$$

Rewriting a portion of the equation for P and evaluating the integral term yields:

$$\int_u^\infty \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} = \int_u^\infty \frac{d\lambda}{\sqrt{a^2(1 + \frac{\lambda}{a^2})(b^2 + \lambda)\lambda}} = \frac{1}{a} \int_u^\infty \frac{d\lambda}{\sqrt{(1 + \frac{\lambda}{a^2})(b^2 + \lambda)\lambda}} = \frac{1}{a}. \quad (.)$$

Further simplification yields:

$$\frac{h}{l_1 + l_2} = \frac{3p}{16\pi} \int_u^\infty \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} = \frac{3p}{16\pi} \cdot \frac{1}{a} \quad (.)$$

Therefore,

$$h = \frac{3p(l_1 + l_2)}{16\pi \cdot a}. \quad (.)$$

Similarly, manipulation of the expression for M given above results in:

$$\frac{A}{l_1 + l_2} = \frac{3p}{16\pi} \int_0^\infty \frac{x^2 d\lambda}{\sqrt{(a^2 + \lambda)^3(b^2 + \lambda)\lambda}} = \frac{3p}{16\pi} \int_0^\infty \frac{x^2 d\lambda}{\sqrt{(a^2)^3(1 + \frac{\lambda}{a^2})^3(b^2 + \lambda)\lambda}}. \quad (.)$$

Continuing to expand,

$$\frac{A}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \frac{1}{a^3} \int_0^\infty \frac{x^2 d\lambda}{\sqrt{(1 + \frac{\lambda}{a^2})^3 (b^2 + \lambda)\lambda}} = \frac{3p}{16\pi} \frac{1}{a^3}, \quad (.)$$

which leads to

$$a^3 = \frac{3p}{16\pi} \frac{(\ell_1 + \ell_2)}{A} = \frac{3p(\ell_1 + \ell_2)}{16\pi(\rho_1 + \rho_2)}. \quad (.)$$

Solving for a yields

$$a = \sqrt[3]{\frac{3p(\ell_1 + \ell_2)}{16\pi(\rho_1 + \rho_2)}}. \quad (.)$$

Since the manipulation of the expressions for N follows exactly as that of the expressions for M, it will not be presented in its entirety. The end result of both expressions, after substitution for ℓ_1 , ℓ_2 , ρ_1 , and ρ_2 is

$$a = b = \sqrt[3]{\frac{3p \cdot 4E^*}{16 \frac{1}{R}}} \quad (.)$$

Through re-arrangement of the equation above,

$$a^3 = \frac{3pR}{4E^*} \quad (.)$$

Similarly, the expression for h can be written

$$h = \frac{3p(4 \frac{1}{E^*})}{16a} \quad (.)$$

Simplifying the equation above yields the following expression:

$$h = \frac{3p}{4E^* a} \quad (.)$$

This result is significant, as h is the depth that will result from an applied force P at the axis between the centers of the surfaces in contact. From this result, a function for the depth of indentation as a function of the radius of the indenter tip can be derived.

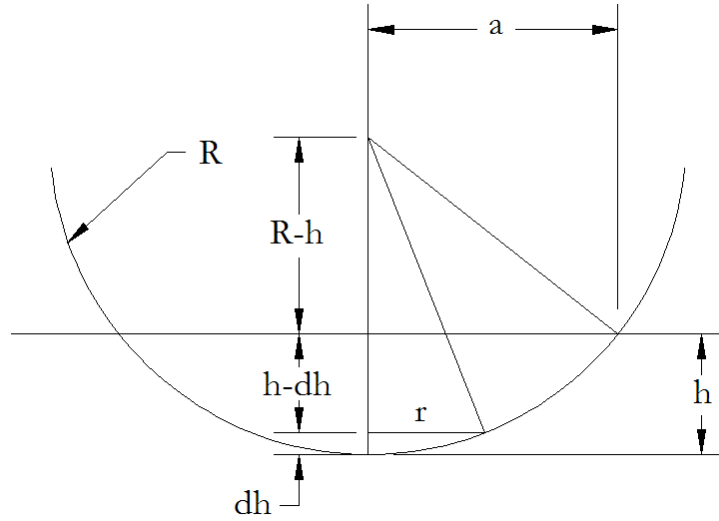


Fig. -. Cross section of an indentation tip.

If it is assumed that

$$x^2 + y^2 = R^2, \text{ then} \quad (.)$$

$$r^2 + (R - (dh))^2 = R^2. \quad (.)$$

Simplification yields

$$r^2 = 2(dh)R - (dh)^2. \quad (.)$$

Through the use of similar triangles,

$$a^2 + (R - h)^2 = R^2, \quad (.)$$

which simplifies to

$$a^2 = 2hR - h^2. \quad (.)$$

If it is assumed that the h^2 and $(dh)^2$ are negligible,

$$r^2 = 2(dh)R, \text{ and} \quad (.)$$

$$a^2 = 2hR. \quad (.)$$

If we divide the two equations, the result is

$$\frac{r^2}{a^2} = \frac{dh}{h}. \quad (.)$$

If it is also assumed that

$$h(r) = h \text{ at } r = 0, \text{ and} \quad (.)$$

$$h(r) = 0 \text{ at } r = a, \quad (.)$$

then

$$h(r) = h - dh. \quad (.)$$

From the previous derivation,

$$dh = h \frac{r^2}{a^2}. \quad (.)$$

Substituting into the expression for $h(r)$,

$$h(r) = h - h \frac{r^2}{a^2} = h \left(1 - \frac{r^2}{a^2}\right). \quad (.)$$

Substituting the expression for h into the expression for $h(r)$,

$$h(r) = \frac{3p}{4E^*} \left(1 - \frac{r^2}{2a^2}\right). \quad (.)$$

Recall that

$$h = \frac{3p}{4E^* a}, \text{ and} \quad (.)$$

$$a^3 = \frac{3pR}{4E^*} \quad (.)$$

Substituting the expression for a into that for h yields

$$h^3 = \left(\frac{3p}{4E^*}\right) \frac{1}{R}. \quad (.)$$

From the previous two equations, it can be seen that

$$h = \frac{a^2}{R}, \quad (.)$$

or, equivalently,

$$a = \sqrt{hR} \quad (.)$$

To find the mean pressure p_m , at the contact surface, recall that the force p is given by the expression:

$$p = \frac{4}{3} hE^* a. \quad (.)$$

Dividing p by the contact area yields

$$p_m = \frac{p}{A} = \frac{\frac{4}{3} hE^* a}{\pi \cdot a^2}, \quad (.)$$

and simplifying yields

$$p_m = \frac{4E^* h}{3a\pi} = \frac{4E^* a}{3\pi R} \quad (.)$$

The derivation for the mean pressure can then be applied to determine the stresses in each of the three mutually perpendicular directions. At the center, where x and y are both equal to zero, and a is equal to b ,

$$Z_z = \frac{3p}{2ab\pi} \sqrt{1 - \frac{x^2}{a^2} - \frac{y^2}{b^2}}, \quad (.)$$

which simplifies to

$$Z_z = \frac{3p}{2a^2\pi}. \quad (.)$$

If the expression for a is substituted, and the resulting equation is simplified,

$$Z_z = \frac{1}{\pi} \sqrt[3]{\frac{6p(E^*)^2}{R^2}}. \quad (.)$$

As demonstrated earlier, since

$$Z_z = -2 \frac{dp}{dz} = \frac{3p}{2ab\pi} \sqrt{1 - \frac{x^2}{a^2} - \frac{y^2}{b^2}}, \text{ then} \quad (.)$$

$$\frac{dp}{dz} = \frac{-3p}{4ab\pi} \sqrt{1 - \frac{x^2}{a^2} - \frac{y^2}{b^2}}; \quad (.)$$

as with the expression for Z_z , this relationship is true only at the surfaces where the pressure is being applied, as the pressure decreases away from the origin. At the origin, however, a is equal to b, and x and y are both equal to zero, resulting in

$$\frac{dp}{dz} = \frac{-3p}{4a^2\pi}. \quad (.)$$

For the stresses in the x- and y-directions, the following expressions are given:

$$X_x = \frac{-E}{(1+\nu)} \frac{de}{dx} + \left(\frac{3p\nu}{a^2\pi}\right), \text{ and} \quad (.)$$

$$Y_y = \frac{-E}{(1+\nu)} \frac{df}{dy} + \left(\frac{3p\nu}{a^2\pi}\right). \quad (.)$$

Having determined the values of each of the principal stresses, the idea of an equivalent stress can be explored. If

$$X_x = \sigma_x, \quad (.)$$

$$Y_y = \sigma_y, \text{ and} \quad (.)$$

$$Z_z = \sigma_z, \text{ and} \quad (.)$$

$$X_y = \tau_{xy}, \quad (.)$$

$$X_z = \tau_{zx}, \text{ and} \quad (.)$$

$$Z_y = \tau_{yz}, \quad (.)$$

the intermediate quantities I_1 , I_2 , and I_3 can be determined:

$$I_1 = \sigma_x + \sigma_y + \sigma_z \quad (.)$$

$$I_2 = \sigma_x \sigma_y + \sigma_y \sigma_z + \sigma_z \sigma_x - \tau_{xy}^2 - \tau_{yz}^2 - \tau_{zx}^2 \quad (.)$$

$$I_3 = \sigma_x \sigma_y \sigma_z + 2\tau_{xy} \tau_{yz} \tau_{zx} - \sigma_x \tau_{yz}^2 - \sigma_y \tau_{zx}^2 - \sigma_z \tau_{xy}^2. \quad (.)$$

It should be noted that the exact values of the shear stresses are unknown, but can be determined through numerical methods. If it is then defined that

$$\sigma^3 - I_1 \sigma^2 + I_2 \sigma - I_3 = 0, \quad (.)$$

and the roots of the equation above, denoted σ_1 , σ_2 , and σ_3 , are the principal normal stresses, the von Mises equivalent stress can be defined as

$$\sigma' = \sqrt{\frac{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2}{2}}. \quad (.)$$

Using the von Mises and yield stresses, the elastic and plastic regimes can be defined. In the elastic regime,

$$\sigma' \leq \sigma_{yield}, \text{ and} \quad (.)$$

$$\sigma' > \sigma_{yield} \quad (.)$$

for the plastic regime. Since the point selected as the origin experiences the highest forces, and, consequently, the highest pressures, the only component of stress at the origin is Z_z ; Hertz uses

this component of stress to define the elasto-plasticity. If hardness, h , is defined as the maximum stress achieved before exceeding the elastic limit, the deformation that occurs can also be characterized into the elastic and plastic regimes using the following criteria:

$$Z_z \leq h, \quad (.)$$

for the elastic region, and

$$Z_z > h \quad (.)$$

for the plastic region.

6.2. Elastic-Plastic Regime

As the mean pressure applied to the area of contact continues to increase, plastic deformation begins to occur in the sample. The first region within the sample that experiences plastic deformation is the point of maximum shear stress, which lies directly beneath the axis of symmetry of a spherical indenter. (Fischer-Cripps, 2005). Through the use of an extension of the Hertzian theory used to describe the elastic regime, and the maximum shear stress failure criterion developed by Tresca, as well as the von Mises failure criterion, a series of formulae can be constructed to describe the deformation that occurs in the transitional elastic-plastic regime.

If an indenter tip with spherical geometry is used, then, in the elastic-plastic regime, the surface of the sample which is in direct contact with the indenter tip will undergo elastic deformation, while the material of the sample below the surface will undergo plastic deformation while remaining surrounded by material deforming elastically. In Johnson's text on contact mechanics, he describes the "expanding cavity model," in which a plastic flow is contained by

elastic material, and where the deformation is expansion happening in a radial manner (Johnson, 1985).

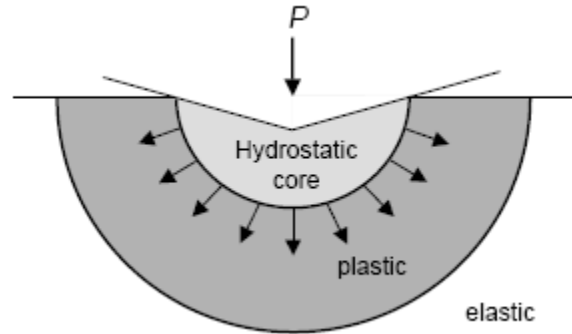


Fig. -. The expanding cavity model (Fischer-Cripps, 2005)

Thus, the formulation of equations that describe the stresses and deformations that occur in the elastic-plastic regime are based on both the Hertzian formulae for the purely elastic region, and the Tresca and von Mises formulae for the perfectly plastic region. The Hertzian formulae and their derivation were presented above; the following discussion will briefly touch upon the pertinent aspects of the Tresca and von Mises theories.

The Tresca theory states that the greatest stress within the sample is given by:

$$\max[|\sigma_1 - \sigma_2|, |\sigma_2 - \sigma_3|, |\sigma_3 - \sigma_1|] = 2k = Y \quad (.)$$

where k and Y are the yield stresses of the sample in shear and tension, respectively (Johnson, 1985). Alternatively, describing the same state of stress using the von Mises theory yields a slightly different value numerically, obtained from the following expression:

$$J_2 = \frac{1}{6}[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2] = k^2 = \frac{Y^2}{3} \quad (.)$$

If we reintroduce the concept of the mean pressure, defined earlier as

$$p_m = \left(\frac{4E^*}{3\pi}\right) \frac{a}{R}, \quad (.)$$

it can be used to find the pressure needed to initiate yield, denoted p_y . First, the intermediate quantity p_o must be determined, using either the Tresca or von Mises criteria. Using the Tresca criterion for axi-symmetric contact (in this case, a sphere) results in the following formulation:

$$p_o = \frac{3}{2} p_m = 3.2k = 1.60Y; \quad (.)$$

using the von Mises criterion results in a similar expression:

$$p_o = 2.8k = 1.60Y. \quad (.)$$

The value of p_o can then be used to determine p_y , the pressure needed to initiate yield in the elastic-plastic regime (Johnson, 1985):

$$p_y = \frac{\pi^3 R^2}{6E^{*2}} (p_o) Y^3 \quad (.)$$

It should be noted that these formulae can be used for indenter tips of spherical geometry only; tips of other geometries follow a similar procedure, but the coefficients in equations 54 and 55 are changed.

6.3. Plastic Regime

Lastly, as the contact pressure is increased even further, fully plastic deformation of the sample occurs. A sample has entered the plastic regime when the plastic region within the sample reaches the surface of the sample, and the plastic flow continues in a manner such that the rate of change of the contact area beneath the indenter tip results in a nearly constant mean pressure p_m . (Fischer-Cripps, 2005). In the plastic regime, as in the preceding elastic-plastic

regime, “subsurface displacements produced by any blunt indenter are approximately radial from the point of first contact, with roughly hemispherical contours of equal strain” (Johnson, 1985).

The contours within the material can be described by the relationship given for J_2 (equation 52).

Johnson (1985) uses a simple relationship between mean pressure p_m and yield stress Y to describe the entire indentation process in all three regimes:

$$p_m = CY \quad (.)$$

In the equation above, C is a constant that is determined by the geometry of the indenter and the material that is being indented, as well as the friction that occurs at the interface between the indenter and the sample. Using a spherical indenter, and assuming a sample made of metal, C has a maximum value of approximately 2.8 to 3. (Johnson, 1985; Fischer-Cripps, 2005).

Thus, in the elastic regime, C has a value less than one, and no plastic deformation occurs. At the onset of yield in the elastic-plastic regime, C has a value of 1 to 1.6, depending on the geometry of the indenter and the sample. As C increases to a value of approximately 3, plastic deformation occurs at a nearly constant mean pressure, regardless of any increase in the load applied to the indenter (Fischer-Cripps, 2005).

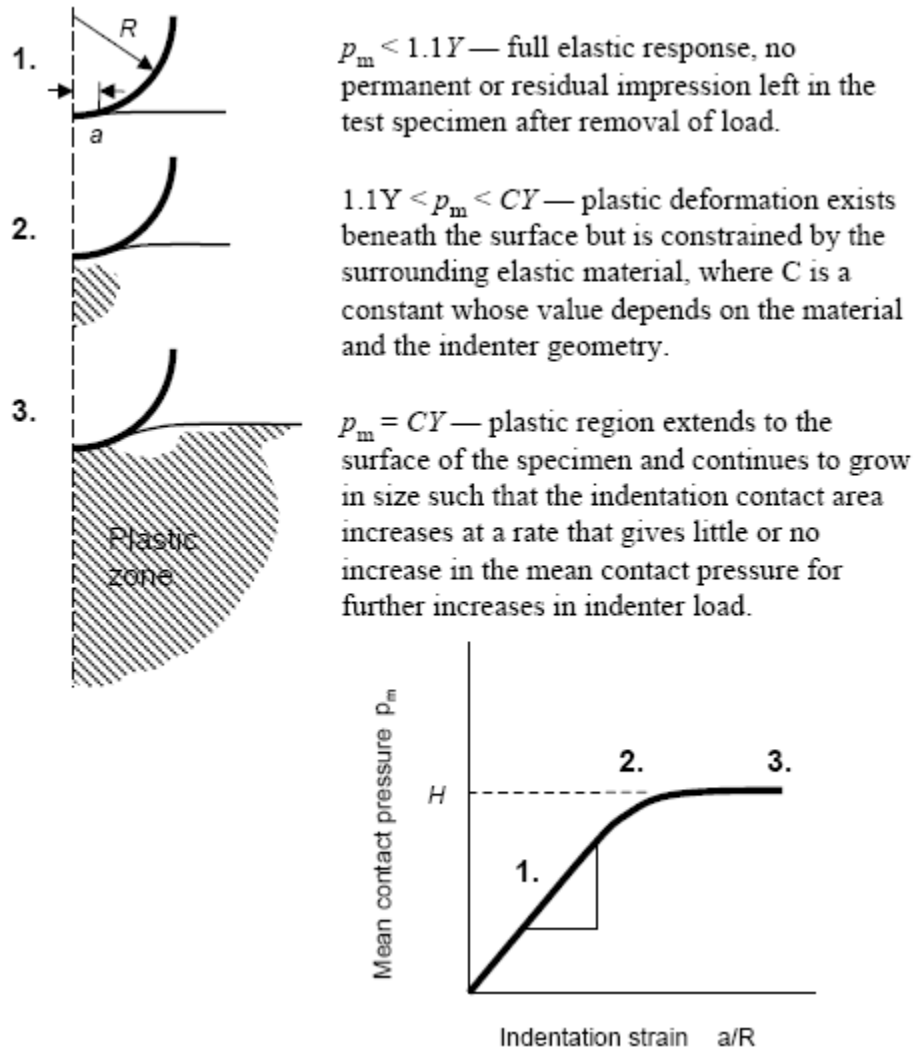


Fig. -. Summary of the Elastic, Elastic-Plastic, and Plastic Regimes (Fischer-Cripps, 2005)

6.4. Non Spherical Indenter Geometries

To this point, the theories and formulae provided have been for spherical indenters only. As different experimental methods were created, however, indenters required for specialized testing were developed, requiring either additions to or modifications of existing theories and equations. Pyramidal indenters, for example, were developed for hardness testing with brittle materials, as “a residual impression is readily obtained at low values of indenter load” (Fischer-

Cripps, 1997). Given the popularity and practical uses of conical and wedge, or pyramidal, indenters, it was deemed appropriate to include a brief discussion of these variances from the theory presented for spherical indenters.

In his work on contact mechanics, Johnson (1985) states that a “rigid punch having sharp square corners has infinite pressure at the edges of contact,” resulting in apparent plastic deformation with no elastic deformation. Thus, the included angle on the indenter, denoted α , must be very close to 90° for the resulting deformation to fall within the linear theory of elasticity. For pyramidal indenters whose sides are not all oriented such that the included angle is constant between all orthogonal faces, current practice within the field is to approximate a pyramidal indenter to an equivalent (based on the area-to-depth ratio) conical indenter included angle α , which simultaneously simplifies calculation and preserves the characteristics of the original pyramidal indenter (Fischer-Cripps, 2005).

Another important modification results from the fact that the conical and pyramidal indenters are not of constant radius over the entire indenter surface, whereas a spherical indenter is. Thus, from the basic definition of pressure, it becomes obvious that the pressure on a sharp apex (which can also be described by an infinitely small radius) is infinite. Thus, Johnson (1985) states, “For compressible materials, infinite elastic pressure at the apex gives rise to theoretically infinite differences in the principal stresses which will cause plastic flow independent of α .” An extension of this theory, used when α is taken into consideration, states that by either the Tresca or von Mises criterion, yield may still occur at the apex if α is such that

$$\cot\alpha \geq \frac{\pi k}{E^*} \quad (.)$$

As the indenter is used, however, the infinitely sharp apex will dull, resulting in a measurable contact area at the tip; correction factors are often used in calculations so that the results with a

slightly worn indenter can be compared with the results obtained using a previously unused indenter.

6.5. Mathematical Modeling of Indentation and Retraction

Using the equations derived in Sections , , and , a model of the theoretical force for a given indentation depth was created. The equations were then manipulated to produce a series of expressions that provide force in terms of indentation depth. The resulting equation accounts for a non-spherical indenter tip as outlined in section with the indenter tip angle denoted α , as this type of tip will be used for the indentation experiments.

$$P_{load}(h) = \frac{4}{\pi} E^* \tan(\alpha) h^2 \quad (.)$$

The above equation only accounts for the force with respect to indentation depth when a material sample is being loaded; that is, the force applied to the sample is being increased. When the material is being unloaded, its elastic properties cause it to recover some of the indentation that has occurred and these elastic restorative forces correspond to a force-displacement curve. The equation for the unloading portion of the curve depends on the maximum loading force, the maximum indentation depth from the loading function, and a parameter h_r , which is equal to the depth of the indentation after it has been unloaded.

$$P_{unload}(h) = \left[\frac{h - h_r}{h_{max} - h_r} \right]^2 P_{load}(h_{max}) \quad (.)$$

The parameter h_r is defined as

$$h_r = \left[\left(1 - \frac{2}{\varepsilon} \right) h_{max} + \frac{2}{\varepsilon} h_c \right], \quad (.)$$

with the parameter ε defined as

$$\varepsilon = \frac{2(\pi - 2)}{\pi}, \quad (.)$$

and h_c defined as

$$h_c = \frac{h_{\max}}{\left(\frac{\pi}{2} - 1\right) \sqrt{\frac{2 \tan(\alpha) \tan(\alpha)}{\pi} \cot(\alpha) + 1}}. \quad (.)$$

Plotting both the loading and unloading curves on the same plot yields the graph in Fig. -. The curve in this figure is the model used to gauge the accuracy of the results from the actual indentation experiments that were performed.

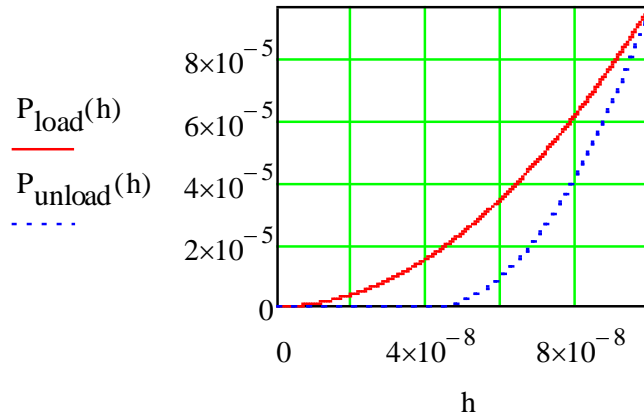


Fig. -. Theoretical loading and unloading functions for indentation plot from MathCAD.

7. Results

Several indentation tests were run in order to get a large sample set of data and to increase the likelihood of obtaining usable data. The images from these indentation tests were input to a Python script program similar to the one used to model the indentation process. The major difference between the programs is this one takes images as inputs instead of creating the image data from theory. All the images from indentation and retraction were unwrapped and the following graphs in Fig. - and Fig. - show the resulting beam contours.

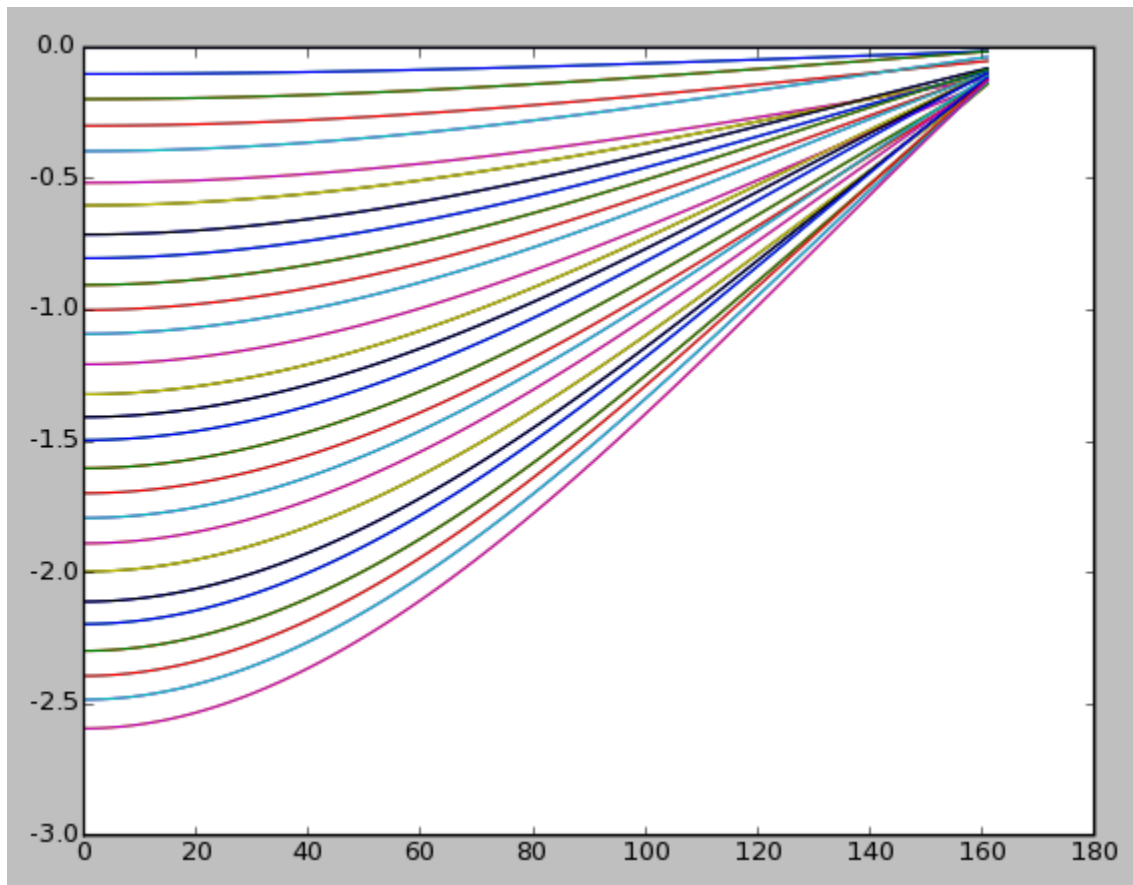


Fig. -. Unwrapped cantilever contour during indentation.

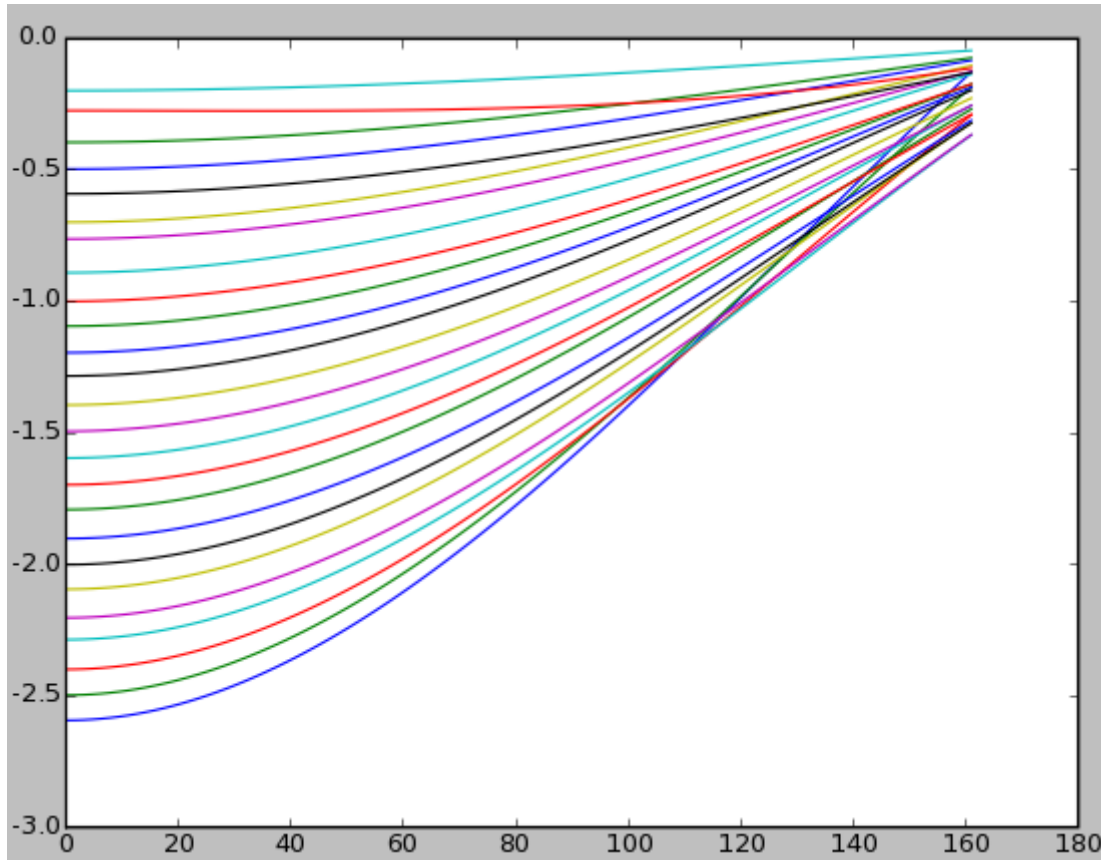


Fig. -. Unwrapped cantilever contour during retraction.

After running the images from all of the indentation tests through the script and attempting to extract the results using ANSYS, it was determined that ANSYS was not solving the problems correctly. An alternative method of finding the force and displacement was implemented. This new method involved looking at the total deflection of the beam from end to end and using that value to determine the force that would cause that deflection based on the following equation.

$$Force = \frac{3(Defl)EI}{L^3} \quad (.)$$

The indentation depth is then determined by subtracting the end position of the beam from its initial position based on the first image input to the program.

This alternative method is accomplished by solving for the force and indentation depth directly in the Python script and outputting a data file. With this method forces of the correct order of magnitude were measured with the images, however, the indentation depths do not match the expected results. The figure below shows a graph of the indentation and retraction data obtain and the best fit lines showing that they follow the appropriate trends.

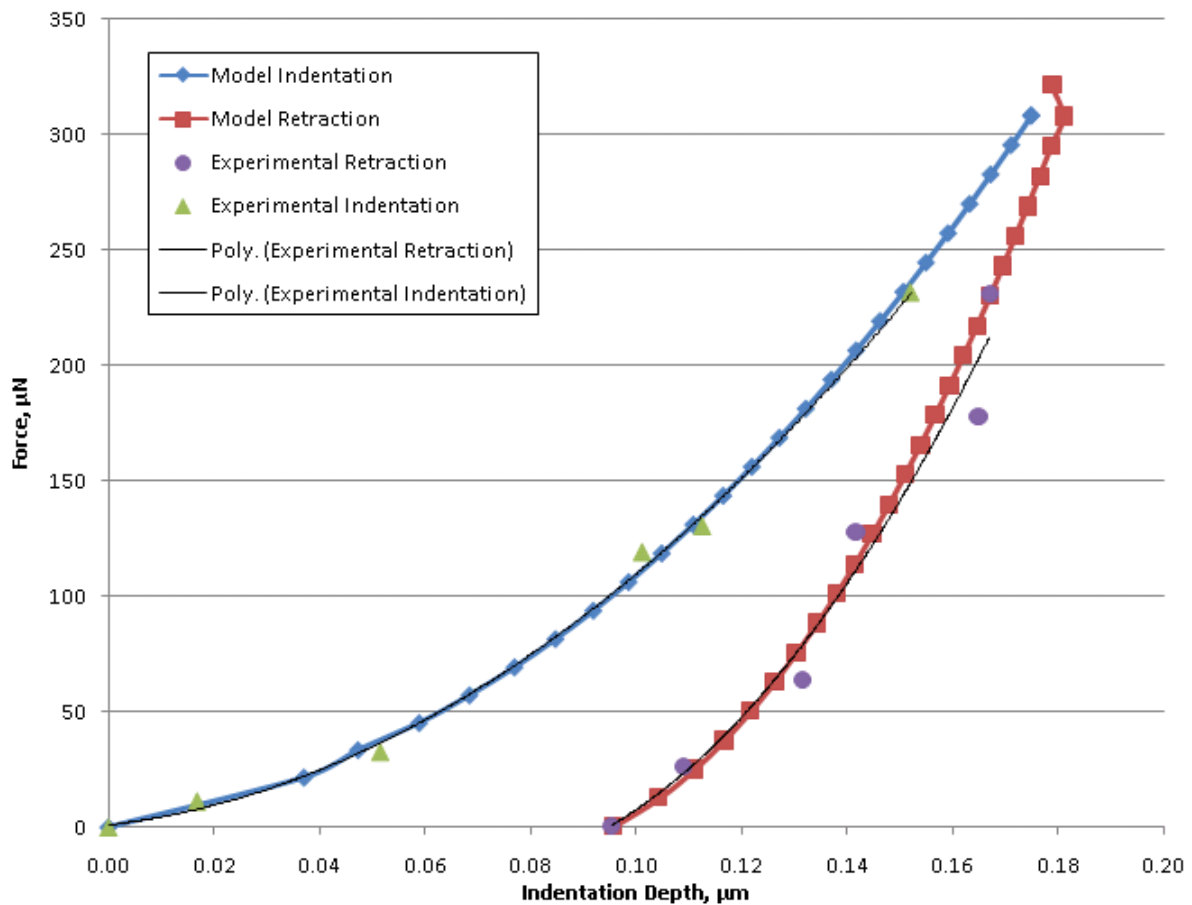


Fig. -. Experimental indentation and retraction data with analytical data.

The curves follow the trends that were predicted, but there is a large amount of noise in the displacement data. This is probably caused by the fact that the indentation was performed by actuating the piezo manually. If the actuation had been computer controlled and done in one smooth motion, the data would more closely match the theoretical data. Also, imperfections in

the surface being indented and stress relaxation of the material during the indentation process could have contributed to the discrepancies in the data.

Below are images of the indentations that were taken with an optical microscope. A scanning electron microscope was preferred, but due to scheduling and time constraints, it was not used.

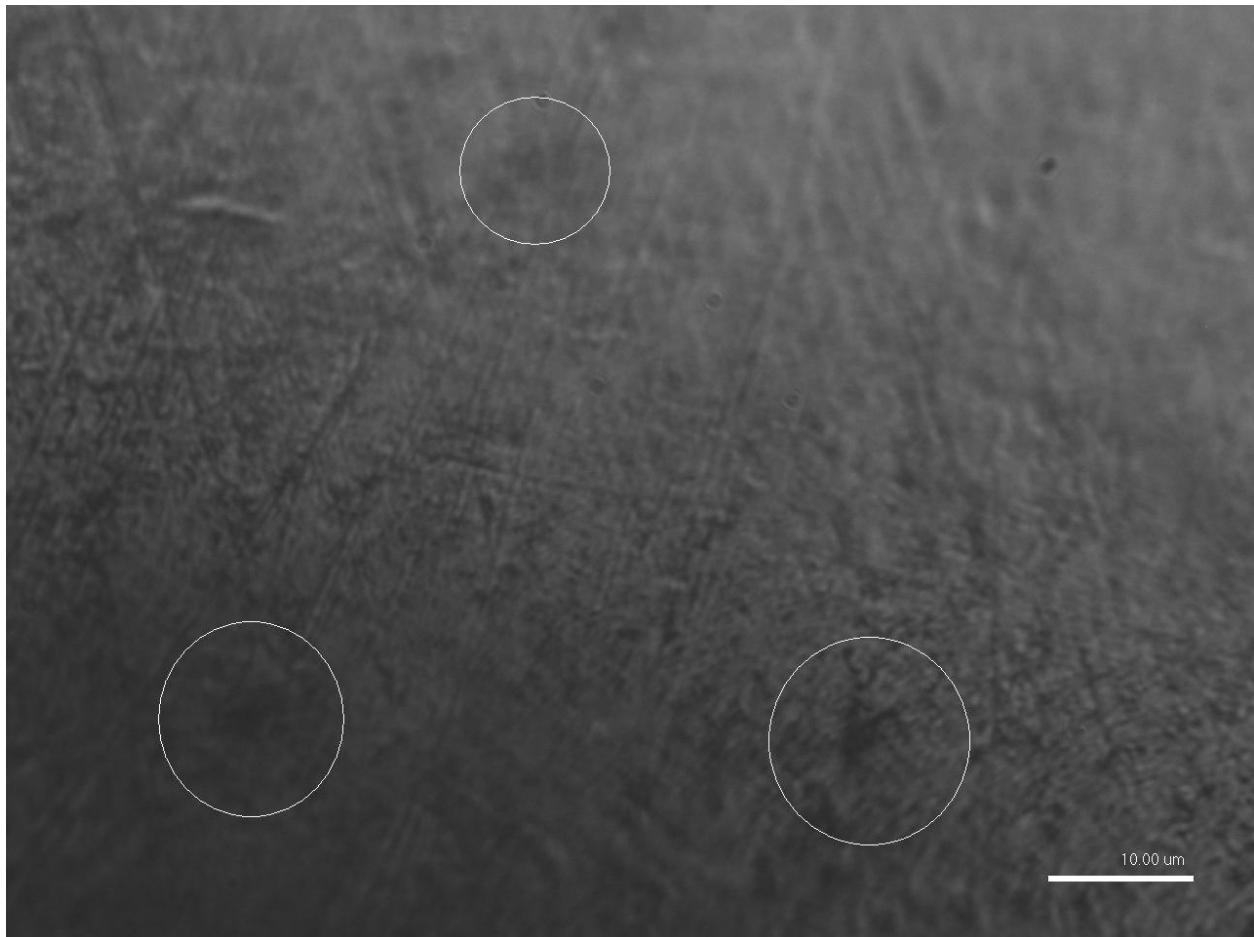


Fig. -. Three indentations created in aluminum sample with the nanoindenter developed as part of this project.

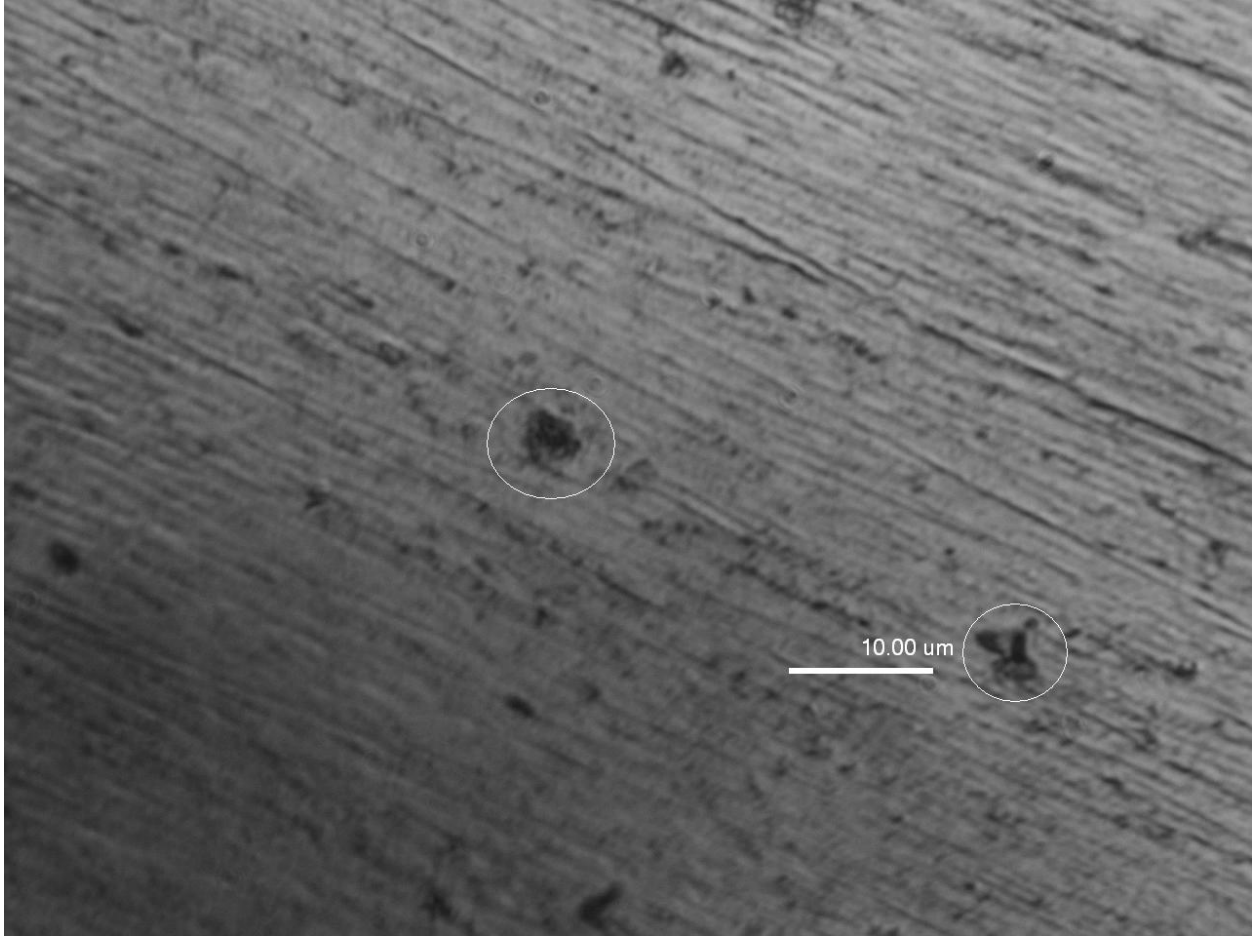


Fig. -. Two indentations created with the nanoindenter developed as part of this project.

8. Conclusions

The inverse approach to nanoindentation was shown to produce data that follows the trends of the theoretical indentation model. Some future work needs to be done to improve the process, streamline the gathering of results, and improve the quality of data obtained. This is outlined in chapter .

The mechanism for performing the indentation was proven to be a robust device. It is easily installed as a module on an existing interferometer with minimal setup work. This project also proved that nanoindentation can be performed without spending over \$100,000 on equipment. The total cost of the equipment used for the construction of this nanoindenter was less than \$5000. All of the parts to build it were either manufactured with three-axis CNC milling machines or purchased as stock parts. This construction of the devices and performing the experiments outlined in this paper can be easily repeated if desired.

Despite the amount of effort that was put into this project, the scope of the project was such that all of the potential work could not be completed within the established timeframe. As a result, there are many improvements that could be made and several further experiments which could be completed if time permitted. This aspect is discussed in chapter .

9. Recommendations

When performing indentation tests, there are certain key aspects of the process that you must pay close attention to detail. Due to the fragile nature of the AFM probes, you should have several spares on hand when conducting your tests. The entire interferometer should be placed on a vibration isolation table because any small disturbance in the room while the tip is close to the sample could cause the tip to break. Before bringing the tip into contact with the sample, ensure that the piezo is fully zeroed and has the complete range of motion. If it is not zeroed properly, it might only move a few microns. Also, when using the motorized z stage to bring the tip in contact with the sample, if it makes contact you should back out at least 100 encoder ticks to ensure that the motor will actuate in the correct direction. For some reason at very small increments the motor has been seen to drive in the wrong direction. This could be due to the fact that the stage is older and could be fixed by upgrading the equipment as outlined in chapter 10.

10. Future Work

The overall project was a success and the data obtained from the actual indentation tests followed a similar trend to the simulated indentation data created during the early phase of the project. The inverse method of finding forces was proven to be effective for this application. There are however, several things that could be done to improve and streamline the indentation process.

While there was large amounts of data obtain from this approach to nanoindentation, there is still much more that can be achieved. This project only scratched the surface of what can be achieved with not only this setup, but also this approach.

When obtaining the indenting and retracting force curves through the inverse method the hard data was a bit noisy, this is why best fit curves were used to plot. These best fit curves were extremely close to the theoretical analysis completed earlier in the process, but not exactly. For some future research into why the forces were correct, but the indentation depths were so scattered. Being able to figure out why this occurs can prove to enhance the output data and form a better original curve.

The research project was mostly completed by manual inputs such as manually jogging the positioning stages, refocusing the interferometer on the being when actuating the stages, and actuating the piezo. To be able to automate all of this in to LabView or even writing new software to process all of these steps in one program then it would make the indentation process smoother and easier. Creating a program may also help improve the data acquisition which can turn into smoother force curves and better results, but that can only be determined by creating an additional program. While the program seems simple it is most likely very complicated due to the visual checks that the computer must obtain to focus and refocus the cantilever beam. This is

also a much larger investment since there will have to be other cameras and lasers to process the other tasks such as focusing, finding the beam, finding the sample, and measuring the distance to the sample.

Within the timeframe available for the project, there was not sufficient time to automate the indentation process. Currently, a user needs to manually give serial commands to the motorized vertical axis of the indenter and must use his or her best judgment regarding how close the indenter tip is to the sample. Once in contact, they must manually move the piezo in a given increment, capture an image, and record the position of the piezo and the image number. This data is then input into the computer and the images are associated with their corresponding piezo positions to obtain the correct data. When it is time to extract the data from the images, the user needs to find the coordinates of the cantilever beam on the image and manually enter them into the Python script.

Ideally, these processes would be automated to increase the speed and reliability of the entire indentation cycle. Using computer vision algorithms, the motorized z stage could be brought into contact with the sample under computer control. This would require complex coding and might not be feasible in the near future.

Automation of the piezo actuation and image acquisition steps can be readily implemented using LabView. The T-Cube Piezo Driver and Strain Gauge Readers are controlled by ActiveX controls that can be added to a LabView block diagram. A LabView program could actuate the piezo by a given increment, acquire an image, associate that image with the piezo position, and repeat the process until the desired indentation was achieved. This is the most obvious improvement to be made, and would be done if there was more time available for the

project. Automating the process could easily reduce the time per nanoindentation from about 30 minutes to a few seconds.

Another topic for future research would be to upgrade the components of the module. Using piezos that have better resolution may further improve the data. With this being said, the piezo that was used in this project was more than acceptable with a resolution of 5 nm, but with a piezo that has a resolution of 0.5 nm, could add more critical points in the indentation process which may again make the force curve smoother. A downside to this, again, is money. While it may not be more expensive to obtain such a piezo, it is much more likely to be more expensive and not just for the piezo, but for the controller as well.

Another component to upgrade would be the AFM probes. Getting probes that have anything different such as, longer or shorter length, different thickness, different probe materials, or different shape of the tip can give different results. While these results may not be better than the results obtained in this research project, the possibilities of indenting different materials are endless. A specific example would be to obtain an AFM probe that has a larger thickness, shorter length, and a high spring constant. With this example it would be possible to indent harder materials.

The next component that could be upgraded to make the design better would be the stages. While the stages that were used in this project were more than sufficient, after the returning of the ThorLabs stage and obtaining the National Aperture stages, there was only one stage that was motorized and that stage was more than 15 years old. Having all the stages motorized would make controlling the module much easier and not having to touch the isolation table during the indenting process is very valuable. This is noticeable since there were a couple

tips that were broken during the indentation process and actuating the stages. With all the stages motorized this would also make the indenting process easily convertible to be fully automated.

11. References

Asylum Research, Inc., “The MFP-3D NanoIndenter for Quantitative Surface Characterization,” Accessed 28 February 2008, <http://www.asylumresearch.com/Products/NanoI/NanoIndenter.pdf>.

Cornell University, “Nanoindentation Testing Machines,” Accessed 28 February 2008, <http://www.nanoindentation.cornell.edu/Machine/Nanoindentation-Machine.htm>.

Cornell University, “The Ultra Micro Indentation System,” Accessed 28 February 2008, <http://www.nanoindentation.cornell.edu/Machine/Nanoindentation-Machine.htm>.

Fischer-Cripps, A. C., 1997, “Elastic–plastic behaviour in materials loaded with a spherical indenter,” *J. Mtls. Sci.*, 32, 727-736.

Fischer-Cripps, A. C., 2005, *The IBIS Handbook of Nanoindentation*, Fischer-Cripps Laboratories, Forestville, NSW, Australia.

Furman, B.J., 2007, “Sensors – Applications,” http://www.engr.sjsu.edu/bjfurman/courses/ME106/ME106pdf/sensor_overview.pdf.

Hertz, H., 1896, *Miscellaneous Papers*, Macmillan, London, Chap. 5 and 6.

Hibbeler, R.C., 2005, *Mechanics of Materials, Sixth Ed.* Pearson, Upper Saddle River, New Jersey.

Honeywell Sensotec, “Model S5 AC-AC Ultra Precision LVDT,” Accessed 3 September 2007. http://www.sensotec.com/pdf_catalog06/S5.pdf.

Johnson, K.L., 1985, *Contact Mechanics*, University Press, London, Chap. 6.

Khazan, A. D., 1994, *Transducers and Their Elements*, Prentice Hall, Upper Saddle River, New Jersey.

Micro Photonics Inc, “Nanovea Series Nanoindentation Tester,” Accessed 28 February 2008, <http://www.microphotonics.com/nanoindentationtester.htm>.

MicroStrain, Inc., “Subminiature DVRT®,” Accessed 3 September 2007, http://www.microstrain.com/sdvrt_specs.aspx.

MicroStrain, Inc., “Subminiature DVRT®,” Accessed 3 September 2007, http://www.microstrain.com/sdvrt_buy.aspx.

Moaveni, S., 2003, *Finite Element Analysis: Theory and Application with ANSYS*, Pearson, Upper Saddle River, New Jersey.

MTS Nano Instruments, "Nanoindenter G200," Accessed 28 February 2008, http://www.mtsnano.com/pdf/MTS_G200brochure_web.pdf.

Omega Engineering, "Miniature DC Displacement Transducers," Accessed 3 September 2007, <http://www.omega.com/Pressure/pdf/LD400.pdf>.

Omega Engineering, "Non-Contact Linear Displacement Sensors," Accessed 3 September 2007, <http://www.omega.com/Pressure/pdf/LD701.pdf>.

National Aperture, Inc., "NAI – MM-3M-F Folded Micro-Mini Stage," Accessed 21 February 2008, <http://www.naimotion.com/mm3mf.htm>.

National Instruments. "Laser Interferometer – Developer Zone." Accessed 28 February 2008. <<http://zone.ni.com/devzone/cda/ph/p/id/126/>>.

NEC/Tokin Corporation, "Multilayer Piezoelectric Actuators," Accessed 17 September 2007, http://www.nec-tokin.com/english/product/pdf_dl/MultilayerPiezoelectricActu.pdf.

Nihon Ceratec, Inc, "Piezo Positioner," Accessed 17 September 2007, http://www.ceratecinc.com/pdf/actuator/PMF_PRB_PFT_PFB.pdf.

Noliac A/S, "Plate Actuator 01," Accessed 17 September 2007, <http://www.noliac.com/Default.aspx?ID=739&NPMode=View&ProductID=8>.

Norton, R. L., 2006, *Machine Design: An Integrated Approach*, Pearson, Upper Saddle River, New Jersey.

Pacific Nanotechnologies, "Probe Clip Holder – Close Contact Mode," Accessed 28 February 2008, <http://www.probestore.com/prclholclcomo.html>.

Physik Instrumente GmbH, "Capacitive Displacement Sensors – Nanometrology Solutions," Accessed 28 February 2008, http://www.capacitance-sensors.com/pdf/Cap_Sens_Web_PI_Capacitive_Displacement_Nanometrology_Capacitance_Gauge_Sensor_E.pdf.

Physik Instrumente GmbH, "Open- and Closed-Loop Piezo Translators (HVPZT)," Accessed 17 September 2007, <http://www.physikinstrumente.com/en/products/prgraph.php?sortnr=101300>.

Physik Instrumente GmbH, "PICMA[®] Chip Monolithic Multilayer Piezo Actuators (LVPZT)," Accessed 17 September 2007, <http://www.physikinstrumente.com/en/products/prgraph.php?sortnr=100800>.

Piezोजना GmbH, "Series P – Stack Type Actuators," Accessed 17 September 2007, http://www.piezोजना.com/files.php4?dl_mg_id=19&file=dl_mg_1108382104.pdf.

RDP Electrosense, "How It Works – LVDT," Accessed 3 September 2007, <http://www.rdpe.com/us/hiw-lvdt.htm>.

TDI International, Inc., "Surface Mount Device Tweezers," Accessed 28 February 2008, <<http://www.tdiinternational.com/usr/tweezers/indv/smd.html>>.

Thorlabs, Inc., "Piezo-Electric Actuators," Accessed 17 September 2007, http://thorlabs.com/NewGroupPage9.cfm?ObjectGroup_ID=61.

Thorlabs, Inc., "T-Cube apt USB Piezo Controller," Accessed 21 February 2008, http://thorlabs.com/NewGroupPage9.cfm?ObjectGroup_ID=2421&pn=TPZ001.

Thorlabs, Inc., "T-Cube apt USB Strain Gauge Reader," Accessed 21 February 2008, http://www.thorlabs.com/newgrouppage9.cfm?objectGroup_ID=2423.

12. Appendices

12.1. Appendix A – Derivation of Hertzian Contact Formulae

Before a full understanding of Hertz's work can be obtained, one must fully understand Hooke's Law, as the work of Hertz is based on the work done by Hooke two hundred years earlier. As can be determined experimentally, "... the stress-strain diagrams for most engineering materials exhibit a *linear relationship (sic)* between stress and strain within the elastic region. Consequently, an increase in stress causes a proportionate increase in stress. This fact was discovered by Robert Hooke in 1676 using springs and is known as *Hooke's Law (sic)*" (Hibbeler, 2005). Expressed mathematically, Hooke's Law can be written

$$\sigma = E\varepsilon \quad (.)$$

where σ is defined as the stress, E is Young's modulus, and ε is the strain.

Although Hertz begins the explanation of his contact stress theory with the definitions of Kirchhoff's elastic coefficients, the final form is an analogue to Hooke's Law. To begin the derivation of the governing equations of the Hertzian contact stress theory, one must first make an assumption about the geometry at the interface between the indenter and the specimen. Assuming the indenter is of spherical geometry and the specimen is flat, the perimeter of contact between the tip and the sample will be circular. As mentioned above, Hertz (1896) begins with the definitions of K and θ , which are the elastic coefficients defined by Kirchhoff. The expression for θ is derived from the following expression:

$$\nu = \frac{\theta}{1 + 2\theta} \quad (.)$$

where ν is Poisson's ratio; algebraic manipulation of the equation yields:

$$\theta = \frac{\nu}{1-2\nu}. \quad (.)$$

To derive the expression for K, begin with

$$E = 2K \left(\frac{1+3\theta}{1+2\theta} \right) \quad (.)$$

and arrive at the expression

$$K = \frac{E}{2(1+\nu)} \quad (.)$$

Hertz then uses a contraction to simplify the mathematics of future steps, defining

$$\ell = \frac{2(1+\theta)}{K(1+2\theta)}. \quad (.)$$

If K and θ are substituted and the expression is simplified, the expression for ℓ becomes

$$\ell = \frac{4(1-\nu^2)}{E} \quad (.)$$

If P is assumed to be a function for the potential of a distribution of electricity on the finite part of $z = 0$ and $\nabla^2 P = 0$. Hertz then defines

$$\Pi = \frac{-zP}{K} + \frac{1}{K(1+2\theta)} \left(\int_z^i P dz - J \right), \quad (.)$$

where i is infinity and j is chosen to make Π finite. When the integral is evaluated, the resulting expression is

$$\nabla^2 \Pi = \frac{-4(1+\nu)}{E} \frac{dP}{dz} \quad (.)$$

If e , f , and g are defined as the components of displacement in the x -, y -, and z -directions, respectively, then, for equilibrium, each of the following conditions must be satisfied within each of the bodies in contact

$$0 = \nabla^2 e + \left(1 + \frac{2\nu}{1-\nu}\right) \frac{d\varepsilon}{dx} \quad (.)$$

$$0 = \nabla^2 f + \left(1 + \frac{2\nu}{1-\nu}\right) \frac{d\varepsilon}{dy} \quad (.)$$

$$0 = \nabla^2 g + \left(1 + \frac{2\nu}{1-\nu}\right) \frac{d\varepsilon}{dz} \quad (.)$$

$$\varepsilon = \frac{de}{dx} + \frac{df}{dy} + \frac{dg}{dz} \quad (.)$$

From these relationships, it can be determined that

$$e = \frac{d\Pi}{dx} \quad (.)$$

$$f = \frac{d\Pi}{dy} \quad (.)$$

$$g = \frac{d\Pi}{dz} + 2\ell P \quad (.)$$

And also that

$$\frac{de}{dx} = \frac{d^2\Pi}{dx^2} \quad (.)$$

$$\frac{df}{dy} = \frac{d^2\Pi}{dy^2} \quad (.)$$

$$\frac{dg}{dz} = \frac{d^2\Pi}{dz^2} + 2\ell \frac{dP}{dz} \quad (.)$$

Substituting these relationships into the expression for ε ,

$$\varepsilon = \frac{d^2\Pi}{dx^2} + \frac{d^2\Pi}{dy^2} + \frac{d^2\Pi}{dz^2} + 2\ell \frac{dP}{dz} \quad (.)$$

Simplifying yields

$$\varepsilon = \nabla^2\Pi + \frac{8(1-\nu^2)}{E} \frac{dP}{dz} \quad (.)$$

Substituting the expression for $\nabla^2\Pi$ into the expression for ε results in

$$\varepsilon = \left[\frac{-4(1+\nu)}{E} + \frac{8(1-\nu^2)}{E} \right] \frac{dP}{dz} \quad (.)$$

Simplifying yields

$$\varepsilon = - \left[\frac{(\nu-2)^2}{E} \right] \frac{dP}{dz} \quad (.)$$

To find the pressure components in a principal direction, begin with the expression

$$X_x = -2K \left[\frac{d^2\Pi}{dx^2} + \frac{2\theta}{K(1+2\theta)} \frac{dP}{dz} \right] \quad (.)$$

Simplifying the above expression yields

$$X_x = \frac{-E}{(1-\nu)} \frac{de}{dx} - 4\nu \frac{dP}{dz} \quad (.)$$

Similarly, to find the shear pressures, begin with the expression

$$X_y = -2K \frac{d^2\Pi}{dx \cdot dy} \quad (.)$$

Simplifying the above expression yields

$$X_y = \frac{-E}{(1+\nu)} \frac{d^2\Pi}{dx \cdot dy} \quad (.)$$

Also,

$$X_z = -2K \left(\frac{d^2\Pi}{dx \cdot dz} \ell \cdot \frac{dP}{dx} \right) \quad (.)$$

which simplifies to

$$X_z = 2z \frac{d^2P}{dx \cdot dz} \quad (.)$$

Using the same formulation results in the following equations

$$Y_z = -2K \left(\frac{d^2\Pi}{dy \cdot dz} + \ell \frac{dP}{dy} \right) = 2z \frac{d^2P}{dy \cdot dz} \quad (.)$$

$$Z_z = -2K \left[\frac{d^2\Pi}{dz^2} + \left[\frac{2(2+3\theta)}{K(1+2\theta)} \right] \frac{dP}{dz} \right] = \frac{-E}{1+\nu} \frac{dg}{dz} - 4(2-\nu) \frac{dP}{dz} \quad (.)$$

Hertz then introduces a function which equates electric potential and pressure, which is defined

as the derivative of P with respect to z:

$$\frac{dP}{dz} = \frac{-3p}{8\pi} \cdot z \cdot \int_u^\infty \frac{1}{\lambda \cdot \sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} d\lambda \quad (.)$$

Simplifying the equation above yields

$$\frac{dP}{dz} = \frac{-3p}{8\pi \cdot a \cdot b} \cdot \frac{z^2 \left(\frac{du}{dz} \right)}{u\sqrt{u}}, \quad (.)$$

where p is defined as force.

Substituting dP/dz into the equation of Z_z yields

$$Z_z = \frac{3p}{2\pi \cdot a \cdot b} \cdot \sqrt{1 - \frac{x^2}{a^2} - \frac{y^2}{b^2}}, \quad (.)$$

for the z = 0 plane.

Inside of the contact surface, the following conditions must be satisfied:

$$P = L - Mx^2 - Ny^2 \quad (.)$$

$$(\ell_1 + \ell_2)M = A \quad (.)$$

$$(\ell_1 + \ell_2)N = B \quad (.)$$

$$(\ell_1 + \ell_2)L = h \quad (.)$$

Thus, a, b, and h must be defined to satisfy these conditions. To accomplish this, a and b are defined as follows:

$$a = \frac{A}{\ell_1 + \ell_2} \cdot \frac{16\pi}{3p} \quad (.)$$

$$b = \frac{B}{\ell_1 + \ell_2} \cdot \frac{16\pi}{3p} \quad (.)$$

Hertz then gives four principal curvatures, denoted by ρ_{11} , ρ_{12} , ρ_{21} , and ρ_{22} . If it is assumed that

$$\rho_{11} = \rho_{12} = \frac{1}{R_1} = \rho_1, \text{ and} \quad (.)$$

$$\rho_{21} = \rho_{22} = \frac{1}{R_2} = \rho_2 \quad (.)$$

where R_1 and R_2 are the radii of the spheres, then the area of contact will be circular, meaning that

$$a = b = \sqrt[3]{\frac{3p(\ell_1 + \ell_2)}{16(\rho_1 + \rho_2)}} \quad (.)$$

From the equations above, the equation for h can be derived:

$$h = \frac{3p(\ell_1 + \ell_2)}{16a} \quad (.)$$

Hertz then introduces the concepts of a composite radius, R, and a composite elastic modulus, E^* . Both derivations are similar to that of the derivation of springs in series through the use of Hooke's Law:

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} = \rho_1 + \rho_2 \quad (.)$$

and

$$\frac{1}{E} = \frac{(1-\nu_1^2)}{E_1} + \frac{(1-\nu_2^2)}{E_2} \quad (.)$$

Recall that

$$\ell_1 = \frac{4(1-\nu_1^2)}{E_1} \quad (.)$$

$$\ell_2 = \frac{4(1-\nu_2^2)}{E_2} \quad (.)$$

Therefore,

$$\ell_1 + \ell_2 = \frac{4}{E^*} \quad (.)$$

If we then define the two surfaces of contact as

$$z_1 = A_1x^2 + Cxy + B_1y^2 \quad (.)$$

$$z_2 = A_2x^2 + Cxy + B_2y^2, \quad (.)$$

the resulting expression is

$$z_1 - z_2 = Ax^2 + By^2, \text{ where} \quad (.)$$

$$A = A_1 - A_2 \text{ and} \quad (.)$$

$$B = B_1 - B_2. \quad (.)$$

Because A represents the curvature in the x-direction and B represents the curvature in the y-direction,

$$2(A + B) = \rho_{11} + \rho_{12} + \rho_{21} + \rho_{22}. \quad (.)$$

Therefore, assuming the two planes are in contact,

$$2A = 2B = \rho_{11} + \rho_{12} + \rho_{21} + \rho_{22}, \quad (.)$$

which, using the simplification presented earlier in the derivation, leads to the following expression:

$$A = B = \rho_1 + \rho_2. \quad (.)$$

This result is then substituted into the integral of the equation Hertz uses to equate pressure and electrical potential, resulting in the following:

$$P = \frac{3p}{16\pi} \int_u^\infty \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} - \frac{3p}{16\pi} \int_u^\infty \frac{x^2}{a^2 + \lambda} \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} - \frac{3p}{16\pi} \int_\alpha^\infty \frac{y^2}{b^2 + \lambda} \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} \quad (.)$$

which can be simplified to

$$P = L - Mx^2 - Ny^2, \quad (.)$$

where

$$L = \frac{h}{\ell_1 + \ell_2}, \quad (.)$$

$$M = \frac{A}{\ell_1 + \ell_2}, \text{ and} \quad (.)$$

$$N = \frac{B}{\ell_1 + \ell_2}. \quad (.)$$

Manipulation of the expression for L results in:

$$\frac{h}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \int_u^\infty \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} = \frac{3p}{16\pi} \int_u^\infty \frac{d\lambda}{\sqrt{(a^2b^2 + \lambda b^2 + \lambda a^2 + \lambda^2)\lambda}}. \quad (.)$$

Further simplification yields:

$$\frac{h}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \int_u^\infty \frac{d\lambda}{\sqrt{(\lambda^3 + \lambda^2(a^2 + b^2) + \lambda(a^2b^2))}} \quad (.)$$

If we introduce

$$\omega = (\lambda^3 + \lambda^2(a^2 + b^2) + \lambda(a^2b^2)), \quad (.)$$

and, consequently,

$$d\omega = (3\lambda^2 + 2\lambda(a^2 + b^2) + a^2b^2)d\lambda, \quad (.)$$

then the expression for L becomes

$$\frac{h}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \int_u^\infty \frac{2\omega}{\omega^2} = \frac{3p}{16\pi} \left[\frac{2\omega^{\frac{1}{2}}}{3\lambda^2 + 2\lambda(a^2 + b^2) + a^2b^2} \right], \quad (.)$$

evaluated over the range from u to ∞ . The equivalent integral becomes

$$\frac{h}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \left[\frac{2\sqrt{\lambda^3 + \lambda^2(a^2 + b^2) + \lambda(a^2b^2)}}{3\lambda^2 + 2\lambda(a^2 + b^2) + a^2b^2} \right], \quad (.)$$

evaluated from 0 to ∞ . Rewriting a portion of the equation for P yields:

$$\int_u^\infty \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} = \int_u^\infty \frac{d\lambda}{\sqrt{a^2(1 + \frac{\lambda}{a^2})(b^2 + \lambda)\lambda}} = \frac{1}{a} \int_u^\infty \frac{d\lambda}{\sqrt{(1 + \frac{\lambda}{a^2})(b^2 + \lambda)\lambda}} = \frac{1}{a}. \quad (.)$$

Substituting into the equation for L yields:

$$\frac{h}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \int_u^\infty \frac{d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)\lambda}} = \frac{3p}{16\pi} \cdot \frac{1}{a} \quad (.)$$

Therefore,

$$h = \frac{3p(\ell_1 + \ell_2)}{16\pi \cdot a}. \quad (.)$$

Similarly, manipulation of the expression for M given above results in:

$$\frac{A}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \int_0^\infty \frac{x^2 d\lambda}{\sqrt{(a^2 + \lambda)^3(b^2 + \lambda)\lambda}} = \frac{3p}{16\pi} \int_0^\infty \frac{x^2 d\lambda}{\sqrt{(a^2)^3(1 + \frac{\lambda}{a^2})^3(b^2 + \lambda)\lambda}}. \quad (.)$$

Continuing to expand,

$$\frac{A}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \frac{1}{a^3} \int_0^\infty \frac{x^2 d\lambda}{\sqrt{(1 + \frac{\lambda}{a^2})^3 (b^2 + \lambda)\lambda}} = \frac{3p}{16\pi} \frac{1}{a^3}, \quad (.)$$

which leads to

$$a^3 = \frac{3p}{16\pi} \frac{(\ell_1 + \ell_2)}{A} = \frac{3p(\ell_1 + \ell_2)}{16\pi(\rho_1 + \rho_2)}. \quad (.)$$

Solving for a yields

$$a = \sqrt[3]{\frac{3p(\ell_1 + \ell_2)}{16\pi(\rho_1 + \rho_2)}}. \quad (.)$$

Lastly, beginning with the expression for N,

$$\frac{B}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \int_0^\infty \frac{y^2 d\lambda}{\sqrt{(a^2 + \lambda)(b^2 + \lambda)^3 \lambda}} = \frac{3p}{16\pi} \int_0^\infty \frac{y^2 d\lambda}{\sqrt{(b^2)^3 (1 + a^2)(1 + \frac{\lambda}{b^2})^3 \lambda}}; \quad (.)$$

continuing the expansion,

$$\frac{B}{\ell_1 + \ell_2} = \frac{3p}{16\pi} \frac{1}{b^3} \int_0^\infty \frac{y^2 d\lambda}{\sqrt{(a^2 + \lambda)(1 + \frac{\lambda}{b^2})\lambda}} = \frac{3p}{16\pi} \frac{1}{b^3}. \quad (.)$$

If the equation is put in the same form as that for a given above, the resulting equation is

$$b = \sqrt[3]{\frac{3p(\ell_1 + \ell_2)}{16\pi(\rho_1 + \rho_2)}} \quad (.)$$

From these results, it can be seen that

$$a = b = \sqrt[3]{\frac{3p \cdot 4E^*}{16 \frac{1}{R}}} \quad (.)$$

Through re-arrangement of the equation above,

$$a^3 = \frac{3pR}{4E^*} \quad (.)$$

Similarly, the expression for h can be written

$$h = \frac{3p(4\frac{1}{E^*})}{16a} \quad (.)$$

Simplifying the equation above yields the following expression:

$$h = \frac{3p}{4E^*a} \quad (.)$$

This result is significant, as h is the depth that will result from an applied force P at the axis between the centers of the surfaces in contact. From this result, a function for the depth of indentation as a function of the radius of the indenter tip can be derived.

If it is assumed that

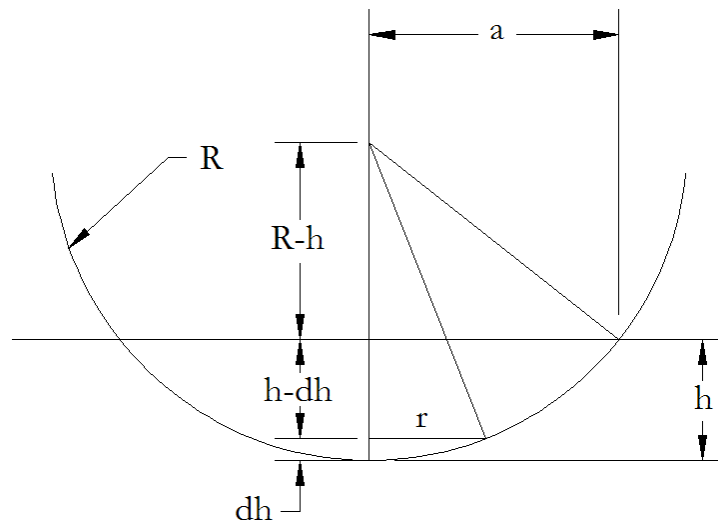


Fig. -. Cross section of an indentation tip.

$$x^2 + y^2 = R^2, \text{ then} \quad (.)$$

$$r^2 + (R - (dh))^2 = R^2. \quad (.)$$

Simplification yields

$$r^2 = 2(dh)R - (dh)^2. \quad (.)$$

Through the use of similar triangles,

$$a^2 + (R - h)^2 = R^2, \quad (.)$$

which simplifies to

$$a^2 = 2hR - h^2. \quad (.)$$

It is assumed that h^2 and $(dh)^2$ are negligible,

$$r^2 = 2(dh)R, \text{ and} \quad (.)$$

$$a^2 = 2hR. \quad (.)$$

If we divide the two equations, the result is

$$\frac{r^2}{a^2} = \frac{dh}{h}. \quad (.)$$

If it is also assumed that

$$h(r) = h \text{ at } r = 0, \text{ and} \quad (.)$$

$$h(r) = 0 \text{ at } r = a, \quad (.)$$

then

$$h(r) = h - dh. \quad (.)$$

From the previous derivation,

$$dh = h \frac{r^2}{a^2}. \quad (.)$$

Substituting into the expression for $h(r)$,

$$h(r) = h - h \frac{r^2}{a^2} = h \left(1 - \frac{r^2}{a^2}\right). \quad (.)$$

Substituting the expression for h into the expression for $h(r)$,

$$h(r) = \frac{3p}{4E^*} \left(1 - \frac{r^2}{2a^2}\right). \quad (.)$$

Recall that

$$h = \frac{3p}{4E^* a}, \text{ and} \quad (.)$$

$$a^3 = \frac{3pR}{4E^*} \quad (.)$$

Substituting the expression for a into that for h yields

$$h^3 = \left(\frac{3p}{4E^*}\right) \frac{1}{R}. \quad (.)$$

Thus,

$$h = \left(\frac{3}{4}\right)^{\frac{2}{3}} p^{\frac{2}{3}} (E^*)^{\frac{-2}{3}} R^{\frac{-1}{3}}, \text{ and} \quad (.)$$

$$a = \left(\frac{3}{4}\right)^{\frac{1}{3}} p^{\frac{1}{3}} R^{\frac{1}{3}} (E^*)^{\frac{-1}{3}}. \quad (.)$$

If the equation for a is re-written as

$$a^2 = \left(\frac{3}{4}\right)^{\frac{2}{3}} p^{\frac{2}{3}} R^{\frac{2}{3}} (E^*)^{\frac{-2}{3}}, \quad (.)$$

and the equations for a^2 and h are divided,

$$h = \frac{a^2}{R}, \quad (.)$$

or, equivalently,

$$a = \sqrt{hR} \quad (.)$$

To find the mean pressure p_m , at the contact surface, recall that the force p is given by the expression:

$$p = \frac{4}{3} h E^* a. \quad (.)$$

Dividing p by the contact area yields

$$p_m = \frac{P}{A} = \frac{\frac{4}{3}hE^*a}{\pi \cdot a^2}. \quad (.)$$

Simplifying yields

$$p_m = \frac{4E^*h}{3a\pi} = \frac{4E^*a}{3\pi R} \quad (.)$$

The derivation for the mean pressure can then be applied to determine the stresses in each of the three mutually perpendicular directions. At the center, where x and y are both equal to zero, and a is equal to b,

$$Z_z = \frac{3p}{2ab\pi} \sqrt{1 - \frac{x^2}{a^2} - \frac{y^2}{b^2}}, \quad (.)$$

which simplifies to

$$Z_z = \frac{3p}{2a^2\pi}. \quad (.)$$

If the expression for a is substituted, and the resulting equation is simplified,

$$Z_z = \frac{1}{\pi} \sqrt[3]{\frac{6p(E^*)^2}{R^2}}. \quad (.)$$

As demonstrated earlier, since

$$Z_z = -2 \frac{dp}{dz} = \frac{3p}{2ab\pi} \sqrt{1 - \frac{x^2}{a^2} - \frac{y^2}{b^2}}, \text{ then} \quad (.)$$

$$\frac{dp}{dz} = \frac{-3p}{4ab\pi} \sqrt{1 - \frac{x^2}{a^2} - \frac{y^2}{b^2}}; \quad (.)$$

as with the expression for Z_z , this relationship is true only at the surfaces where the pressure is being applied, as the pressure decreases away from the origin. At the origin, however, x is equal to y, and both are equal to zero,

$$\frac{dp}{dz} = \frac{-3p}{4ab\pi}, \quad (.)$$

And, for spherical bodies, since a is equal to b,

$$\frac{dp}{dz} = \frac{-3p}{4a^2\pi} \quad (.)$$

For the stresses in the x-direction,

$$X_x = \frac{-E}{(1+\nu)} \frac{de}{dx} - 4\nu \left(\frac{-3p}{4a^2\pi} \right). \quad (.)$$

Simplifying this equation slightly yields

$$X_x = \frac{-E}{(1+\nu)} \frac{de}{dx} + \left(\frac{3p\nu}{a^2\pi} \right). \quad (.)$$

Similarly, for the stresses in the y-direction,

$$Y_y = \frac{-E}{(1+\nu)} \frac{df}{dy} - 4\nu \left(\frac{-3p}{4a^2\pi} \right), \quad (.)$$

and simplification yields

$$Y_y = \frac{-E}{(1+\nu)} \frac{df}{dy} + \left(\frac{3p\nu}{a^2\pi} \right). \quad (.)$$

Having determined the values of each of the principal stresses, the idea of an equivalent stress can be explored. If

$$X_x = \sigma_x, \quad (.)$$

$$Y_y = \sigma_y, \text{ and} \quad (.)$$

$$Z_z = \sigma_z, \text{ and} \quad (.)$$

$$X_y = \tau_{xy}, \quad (.)$$

$$X_z = \tau_{zx}, \text{ and} \quad (.)$$

$$Z_y = \tau_{yz}, \quad (.)$$

the intermediate quantities I_1 , I_2 , and I_3 can be determined:

$$I_1 = \sigma_x + \sigma_y + \sigma_z \quad (.)$$

$$I_2 = \sigma_x \sigma_y + \sigma_y \sigma_z + \sigma_z \sigma_x - \tau_{xy}^2 - \tau_{yz}^2 - \tau_{zx}^2 \quad (.)$$

$$I_3 = \sigma_x \sigma_y \sigma_z + 2\tau_{xy} \tau_{yz} \tau_{zx} - \sigma_x \tau_{yz}^2 - \sigma_y \tau_{zx}^2 - \sigma_z \tau_{xy}^2. \quad (.)$$

It should be noted that the exact values of the shear stresses are unknown, but can be determined through numerical methods. If it is then defined that

$$\sigma^3 - I_1 \sigma^2 + I_2 \sigma - I_3 = 0, \quad (.)$$

and the roots of the equation above, denoted σ_1 , σ_2 , and σ_3 , are the principal normal stresses,

the von Mises equivalent stress can be defined as

$$\sigma' = \sqrt{\frac{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2}{2}}. \quad (.)$$

Using the von Mises and yield stresses, the elastic and plastic regimes can be defined. In the elastic regime,

$$\sigma' \leq \sigma_{yield}, \text{ and} \quad (.)$$

$$\sigma' > \sigma_{yield} \quad (.)$$

for the plastic regime. Since the point selected as the origin experiences the highest forces, and, consequently, the highest pressures, the only component of stress at the origin is Z_z ; Hertz uses this component of stress to define the elasto-plasticity. If hardness, h , is defined as the maximum stress achieved before exceeding the elastic limit, the deformation that occurs can also be characterized into the elastic and plastic regimes using the following criteria:

$$Z_z \leq h, \quad (.)$$

for the elastic region, and

$$Z_z > h \quad (.)$$

for the plastic region.

12.2. Appendix B: Component-Level Research

12.2.1. Piezoelectric Actuation

In most of the nanoindenters currently on the market, the motion of the indenter tip is controlled by a magnetic voice coil or a piezoelectric actuator. Emphasis was placed on piezoelectric actuation during the research phase of the project because they offer greater control over small-scale motions. Since the National Institute for Standards and Technology (NIST) standards for a nanoindenter stipulate that the range of indentation must be less than one micrometer, the piezoactuators that can be used in this application become limited. Although an actuator with a large range of motion can be used, it requires more bits of analog to digital conversion to achieve the same resolution, which often means a more expensive converter and a lower acquisition frequency.

Physik Instrumente (PI) is amongst the leaders in piezo technology, and markets several different lines of piezoactuators. Two of these are of particular interest – the PICMA[®] Chip Monolithic Multilayer Piezo Actuators, and the Open- & Closed-Loop Piezo Translators. The PICMA line of piezoactuators is of interest because of their size; the largest has a cross-section of 5 square millimeters, and has a range of motion of 2.2 micrometers at 100 volts (Physik Instrumente GmbH, 2007). A range of approximately 2 micrometers is amongst the smallest seen in commercially-available piezoactuators; thus, this particular piezo would serve nicely in a nanoindenter, with the only drawback being the additional bits of analog-to-digital conversion needed. The closed-loop actuator, by contrast, has a minimum range of motion of 10 micrometers with a resolution of 0.05 nanometers; the advantage to this particular actuator is

that a closed-loop displacement sensor is included in the actuator itself, eliminating the need for an additional external displacement sensor (Physik Instrumente, GmbH, 2007).

The Tokin division of NEC also markets piezoactuators; the model that would best suit our application is the AE0203D04, which has a range of 3 micrometers, but this range has an uncertainty of 1.5 micrometers, or half of the effective range (NEC-Tokin Corporation, 2007). Thus, when compared with the PI actuators, the loss of precision makes the NEC-Tokin actuators a worse choice for an application where the displacement is crucial to gain valid results.

The smallest displacement piezoactuator made by Piezojena (the P4/10 P-110-00) has a range of 4 micrometers; the resolution is not specified (Piezosystem Jena GmbH, 2007). The force applied by this actuator, however, is much too high for this application; because the actuator can apply up to 1000 pounds, the resolution needed in the micro-Newton range would be much too coarse, if any reading could be detected at all.

Noliac A/S, a Danish firm, markets piezoactuators with specifications similar to those made by PI; the Noliac CMAP-01 has a free stroke of 2.2 micrometers, with no specified resolution. As with the Piezojena actuator, however, the force applied is several orders of magnitude too high for a nanoindenter (Noliac A/S, 2007).

Nihon-Ceratec also advertises piezoactuators with range and force similar to their competitors listed above; Ceratec lists the free range of the smallest one as greater than or equal to 3 micrometers, rather than having a set displacement and uncertainty. The magnitude of the force that can be applied is listed as greater than or equal to 800 Newtons, also (Ceratec, Inc., 2007).

Thorlabs markets actuators similar in specification to those made by NEC-Tokin, with a free range of 3.0 ± 1.5 micrometers. Although there is a claim of “precise nanometer

positioning,” no specific data is given to support this claim. Perhaps the item of most interest with the Thorlabs actuators is the price; the model with 3 micrometers of travel sells for \$72 (Thorlabs Inc. , 2007).

The piezo that was purchased from Thorlabs is a model PAZ-020 closed-loop piezoactuator that has a distance of travel of 20 micrometers. This is a larger range of motion than is necessary; it is about ten microns more than are required for the indentation process. Having the extra travel is beneficial, however; if a deeper indentation was needed, or the sample was not positioned close enough to the indenter tip, the extra travel provides a means to adjust. The twenty-micron closed-loop piezo was purchased for several reasons; it was the piezo with the smallest travel distance in that line of closed loop piezoactuators, and closed-loop operation was a highly desirable feature for this application. The piezo has a resolution of five nanometers, which is acceptable for this application. Other manufacturers produced piezoactuators that are accurate to half of a nanometer, but the pricing for such an actuator exceeded the budget for this module. It is also worth noting that a resolutions of less than five nanometers are not needed for this application.

12.2.2. Linear Variable Differential Transformers

Amongst the current technologies used to accurately determine the penetration depth of a nanoindenter are linear variable differential transformers, or LVDTs. Within an LVDT, there are three coils, a primary and two secondary, through which current is passed. At the center of the coils’ windings is a magnetic core, which moves relative to the coils and creates an electrical current. At the center of the LVDT, both of the secondary coils read zero electric field, and there

is no output. When a voltage is applied across the primary coil, the magnetic core moves away from the center, and there is an increase in the field strength in one of the secondary coils, and a corresponding decrease in the field strength of the other secondary coil. The difference between the field strength in the secondary coils then becomes the output, which can be converted from voltage to a linear displacement (RDP Electrosense, 2007).

The advantages to using an LVDT are that there is no wear within the device, as no moving parts come into contact with each other; resolution is infinite; the LVDT is insensitive to temperature changes, and, if properly sealed, environment; and, lastly, the devices are highly reliable. Compared to some competing technologies, however, they are more expensive, and have a limited frequency response (Furman, 1997).

Leaders in LVDT technology include Omega Engineering, Micro Strain, and Honeywell Sensing and Control. The LD400 series of miniature DC output displacement transducers manufactured by Omega Engineering feature infinite resolution, and are compatible with standard DC signal conditioners (Omega Engineering, 2007). The model of most interest to individuals in the nanoindenting field would be the LD400-2.5, as it has a linear range of motion that is 2.5mm, which is common amongst commercially-available nanoindenters. The LD400-2.5 lists for \$355, a price which includes only the LVDT (Omega Engineering, 2007).

Omega also markets a non-contact displacement transducer, which operates on inductive technology and requires the use of a metal target. While initially more attractive than a standard LVDT due to a lower cost (\$195 for the LD701-2/5, a model comparable to the LVDT above), the repeatability of measurement with the non-contact transducer is not as good as the LVDT, as the non-contact model is only repeatable within $\pm 10 \mu\text{m}$ (Omega Engineering, 2007). For a

nanoindenter, this resolution is not nearly high enough, which excludes this sensor from contention.

Micro Strain markets a differential variable reluctance transducer, or DVRT; the DVRT operates on the same principle as an LVDT, with the difference being that the DVRT only uses half of a Wheatstone bridge, whereas the LVDT uses all four branches of the Wheatstone bridge. For the DVRT, however, the “nano resolution” model only has a stroke of 500 μm or less, with a resolution of 10 nm (Microstrain, Inc. , 2007). While this is much closer to the resolution needed for a nanoindenter, the range of motion at full stroke is slightly lacking. It can be argued, however, that this range is acceptable; if the output from the DVRT was required only after the indenter tip was in contact with the specimen, 500 μm is enough range to take an accurate reading. If the output for the DVRT was needed for coarse adjustment to bring the indenter tip into contact with the specimen, a much larger range is needed. Micro Strain does not quote prices for individual components on their website; for a “starter kit” with one channel and a less precise DVRT, prices start at \$1625 (Microstrain, Inc. , 2007).

Honeywell Sensing and Controls markets the S5 AC-AC Ultra Precision LVDT, which can be purchased in different configurations, with strokes ranging from 0.01 in to 0.5 in. Honeywell, like Omega, advertises their LVDT to have infinite resolution (Honeywell International, Inc. , 2007). The Honeywell website is under construction, meaning prices were unavailable as of this writing; however, they should be available in the future.

12.2.3. Strain Gauges

If we are going to look at implementing strain gages into our project, we are going to have to evaluate a Gauge Factor and determine position the strain gages in our system. The strain gages themselves, can measure both distance (length) and force. In our case we are looking to use them for a force measurement since this proposed design has a LVDT which will be measuring the distance the device is being moved.

The basic equation for strain itself is as follows:

$$\varepsilon_{Lateral} = \frac{D_2 - D_1}{D_1} = \frac{\Delta D}{D} = \frac{dD}{D} \quad (.)$$

This equation defines what lateral strain is when we are trying to measure it. When talking about taking forces there has to be more added to the equation. We have to determine what the forces are on the material in order to stretch the item to that distance. This involves the stresses which are used to deform or stretch the material.

$$\varepsilon_x = \frac{\sigma_x - \nu\sigma_y}{E} \quad (.)$$

$$\varepsilon_y = \frac{\sigma_y - \nu\sigma_x}{E} \quad (.)$$

These two equations show the strain relationship in terms of the stresses that are exerted on the material. E and ν are constants. One is a constant in terms of the material, E (the Modulus of the material) and the other is a constant for all the materials, ν (Poisson's Ratio).

$$\nu = \frac{-\varepsilon_L}{\varepsilon_a} \quad (.)$$

Fig. - is a simple picture of what a strain gauge looks like. What we must determine for our project is the Gauge Factor that we would need to use for one of the proposed designs.

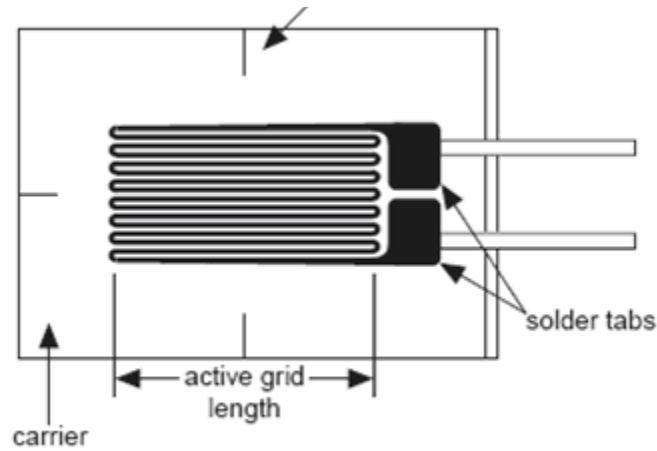


Fig. -. A schematic of a strain gauge.

$$GF = \frac{\frac{\Delta R}{R}}{\frac{\Delta L}{L}} = \frac{\Delta R}{\varepsilon R} \quad (.)$$

The Gauge Factor is defined here in the equation above. In our case we would like to have a delta L to be about 0.5 nm and our total range of about 2mm. We would also need to have a high R in order to have a great enough resolution to read small changes to R. This is where we would need multiple strain gauges in order to measure such miniscule changes. Have a high R then increase our delta R as much as possible to get a smaller value on the top of the equation. The Gauge Factor will be smaller and more realistic.

In our case the use of four strain gauges that have 1000 ohm resistance. We have a combined resistance of 4000 ohms and with our 20-bit DAQ we can measure a change in R of about 0.005 ohms. With these numbers we can achieve a Gauge Factor by simply plugging the values into the equation above.

With our calculated numbers we get a Gauge Factor of 5 which is reasonable with the given strain gauges.

12.2.4. Parallel Plate Capacitive Displacement Sensing

A capacitive displacement sensor is one means of sensing small movements. The sensor is based on the idea of a capacitor with constant voltage across it and a constant cross sectional area. By holding these constant, the current across the capacitor is the only thing that changes as the distance between the plates of the capacitor changes. The relationship between the distance between the plates and the capacitance is not linear, but it can be approximated as linear over small displacements. To make the sensor act in a more linear fashion it is possible to use two capacitors in series. A sensor such as this can resolve displacements as small as 0.1 nm. It can also be built to fit in a relatively small package. The sensor will only work over a small displacement range which is acceptable for application in a nanoindenter (Cornell University, 2007).

One of the world's leading manufacturers of nanomechanical devices is PI. They make commercially available nanosensors that are based on the principle of capacitive distance sensing. PI also makes capacitive sensors which can measure forces in the micro Newton range. They offer a wide range of sensors with ranges between 10 and 100 micrometers. These sensors are very compact and have a high resolution which can be used to accurately measure displacements much smaller than a nanometer (Physik Instrumente GmbH, 2007).

12.2.5. Commercially Available Nanoindenters

12.2.5.1. CSIRO

CSIRO Material Science and Engineering builds nanoindenters that work on a different principle. The load is applied off axis and then cantilever leaf springs transfer the load to the indenter tip. The indenter uses two linear variable differential transformers to measure force and displacement on the z axis (Cornell University, 2007).

12.2.5.2. MTS Nano Instruments

MTS Nano Instruments is one of leading manufacturers of nanoindenters. The indentation axis on their machines is actuated by a magnet and coil. A variable magnetic field is induced in the coil by varying the current through it. The magnetic field of the coil is repelled by the permanent magnet, causing the displacement. The total displacement on the axis is between 1 and 2 mm and can produce a load up to 500 mN. The axis is stabilized by springs to ensure that it moves smoothly. The displacement on the axis is measured by a capacitive distance sensor which can measure sub-nanometer displacements (MTS Systems Corporation, 2007).

12.2.5.3. Micro Photonics Inc.

Micro Photonics Inc. sells the Nanovea Series Nano-indentation Tester. These machines can be used for a variety of tests including nano-indentation, nano-scratch, creep, fatigue, and fracture toughness. The company has an interactive software package which runs under Windows to control the tester and make data acquisition simple. The base machine is versatile and can accept different attachments for different applications (Micro Photonics, Inc. , 2007).

12.2.5.4. Asylum Research

Asylum Research is a company that builds nanoindenters for use in atomic force microscopy. The displacements in its machines are created using high precision piezoelectric actuators. The displacement of the indenter is measured using a precision nano positioning sensor. The force on the axis is measured by measuring the deflection of cantilever springs of a known spring constant and computing the force created by the spring (Asylum Research, 2007).

12.2.5.5. Other Companies

The nanoindenters built by CSM work on the same principles as the ones made by MTS Nano Instruments. Hysitron builds nanoindenters that interface with an AFM so you can quickly and easily setup for nanoindentation testing. Micro materials is another company that manufactures and supplies nanoindenters.

12.2.6. Tip Characterization for Nanoindenters

The tip of the nanoindenter is very important. In such a precise and accurate type of measurement, the tip has to be in perfect working order. There are different types of tips out on the market, but most that are used for the nanoindenter application are the three-sided pyramid and conospherical tips. There are also specialty tips which are used for other applications.

Each tip has a different mathematical way of determining what the hardness of a material is based on the force, the depth of how far the tip went into the material, and the geometry of the tip. The tips can be used for different ways of finding the hardness the most common are indenting and scratching, which most all tips can be used for either, but it is important to

remember that one tip shouldn't be used for both applications. The main reason for using separate tips is because the scratching process can blunt the tip and cause for a different measurement when indenting.

12.2.6.1. Berkovich Tip

The Three-sided pyramidal tip is the most common tip that is used for both indenting and scratching, also known as the Berkovich tip. The Berkovich tip is great for bulk samples except for materials with a roughness greater than 50 RMS.

Table -. Statistics for a Berkovich Tip

Included Angle (angle from one edge to the other)	142. 35 degrees
The Half Angle (angle from the perpendicular to one face)	65. 25 degrees
Aspect Ratio	1:8
Typical Radius of Curvature (for standard Berkovich tips)	150 nm
Typical Radius of Curvature (for Sharper Berkovich tips)	<50nm
Ideal Area for Berkovich tip	Area = $24. 5h_c^2$
Materials used for	Bulk metals, bulk ceramics, glasses, thin hard films greater than 100nm thickness, biomaterials, and hard polymers (modulus greater than 1 GPa)

12.2.6.2. 90-Degree tips

Another three sided pyramidal tip is the 90-degree corner tip. This tip is similar to the Berkovich tip, but since the included angle is 90 degrees and the aspect ratio is 1:1 it is easier to make the radius of curvature smaller.

The best advantage of this tip is that the indentations can be made, but are smaller and the hardness can still be measured accurately. This also allows measurements to be taken on smaller samples. Another advantage that the 90-degree has over the Berkovich tip is that its sharper tip makes it possible to repeat the measurement with better precision.

Table -. Statistics for a 90-Degree Tip

Included Angle (angle from one edge to the other)	90 degrees
The Half Angle (angle from the perpendicular to one face)	N/A
Aspect Ratio	1:1
Typical Radius of Curvature (for standard tips)	75nm
Typical Radius of Curvature (for Sharper tips)	Can be <30nm
Ideal Area for 90-degrees tip	Area = 2. 598 h_c^2
Materials used for	Ultra-thin coatings (<100nm), micro/nano composites, micro/nano structured materials

12.2.6.3. Conospherical tips

These tips are in the shape of a cone and are most of the time made of diamond (very few cases they are made with sapphire or other hard materials with similar properties). The smallest tips for the conospherical shape are limited to a radius of 0. 5um. Conospherical tips with radii greater than 10um are considered non-imaging tips and the non-imaging tips can get as large as 100um.

The imaging conospherical tips are primarily used for a pre-hardness test. Since the imaging resolution on these tips is not very high compared to the three-sided pyramidal shaped tip, they can be used in the cases to get a general hardness test for larger rougher surfaced

materials. The most common materials the conospherical imaging tips are used for polymers with a modulus of >0. 5GPa and hard biomaterials.

The Non-imaging Conospherical tips are used when nano-positioning is not important, but these tips are ideal for soft materials such as rubbery polymers and soft biomaterials.

Table -. Data for Conospherical Tips

Ideal Area Function for Spherical Tips	Area = $-\pi h_c^2 + 2\pi R$ (R=tip radius)
Materials used for	Soft Polymers (Modulus <0. 5 GPa), soft biomaterials, Structured samples, MEMS/NEMS

12.2.6.4. Specialty Tips

The specialty tips are categorized by they are any tip that wasn't specified in the categories above. There are four-sided pyramidal tips and also some flat ended tips.

The first specialty tip that is also a common tip is the Vickers tip. This tip has a similar aspect ratio to the Berkovich tip, but where it lacks is the radius of curvature which is on the scale of 500nm. The Vickers tip is used typically to find a connection between the nano-indentation and micro-indentation and performed with a microindenter.

The next specialty tip that is some interest to the group is the Knoop. Much like the Vickers tip, the Knoop tip is a four-sided pyramidal tip. This tip differs in the cross section where this tip is a rhomboidal so one axis is longer than the other. A down side to the Knoop tip is that it was developed only for micro-indentation. So indentation at the nano-scale is out of the question for this tip and also because of its larger size it is limited to large load applications.

12.3. Appendix C – Python Scripts

12.3.1. IndentationImageProcessor. py

```
import Image
import ImageFilter
from numpy import *
from pylab import *
import datetime
import os
import RussPolyFit
from RussPolyFit import rpolyfit

# Output header
today = datetime. date. today()
print "Anthony DiOrio, Russell Morin, and Eric Wilusz\n", today
print "Nanoindenter MQP\nWorcester Polytechnic Institute\nMechanical Engineering"
print "Image Processing Program for Nanoindentation\n"

# Keyboard input
numofimages = raw_input ("Please enter the number of images in the directory for indentation: ")

firstnumber = raw_input ("Please enter the number of the first indentation image: ")

# Load base filename
basefilename = raw_input ("Please input the base filename for indentation: ")

# Load offset file
offsetfilename = raw_input ("Please input the filename of the offset values for indentation: ")
offsets = open(offsetfilename,'r')

# Keyboard input
numofimages2 = raw_input ("Please enter the number of images in the directory for retraction: ")

firstnumber2 = raw_input ("Please enter the number of the first indentation image: ")

# Load base filename
basefilename2 = raw_input ("Please input the base filename for retraction: ")

# Load offset file
```

```
offsetfilename2 = raw_input ("Please input the filename of the offset values for retraction: ")
offsets2 = open(offsetfilename2,'r')
```

```
# Beam mechanical and material properties
```

```
beamlength = 160 # micrometers
```

```
beamwidth = 37.5 # micrometers
```

```
beamheight = 7.0 # micrometers
```

```
area = beamwidth * beamheight # micrometers^2
```

```
I = (1/12.0) * beamwidth * (beamheight*beamheight*beamheight) # micrometers^4
```

```
E = 169000.0 # kg/(s^2*micrometer)
```

```
magnifiedpixel = 6.25/4 # micrometers
```

```
numelements = 18
```

```
lightwavelength = 0.630 # micrometers
```

```
halfwavelength = lightwavelength / 2.0 # micrometers
```

```
heightstep = halfwavelength / 256.0 # micrometers
```

```
# Beam rectangle box
```

```
box = (370, 230, 372, 333)
```

```
# Process the images for indentation
```

```
w = int(firstnumber) - 1
```

```
secondoffset = []
```

```
forces = []
```

```
indents = []
```

```
while(w < int(numofimages) + int(firstnumber) - 1):
```

```
    # Load image file
```

```
    if ((w + 1) < 10):
```

```
        im = Image.open(basefilename + "000" + str(w + 1) + ".bmp")
```

```
    elif ((w + 1) < 100):
```

```
        im = Image.open(basefilename + "00" + str(w + 1) + ".bmp")
```

```

elif ((w + 1) < 1000):
    im = Image.open(basefilename + "0" + str(w + 1) + ". bmp")
else:
    im = Image.open(basefilename + str(w + 1) + ". bmp")

# Crop the image and store it in the region
region = im.crop(box)
region = region.convert("L")

# Take all the values of the pixels and store them in the matrix
x = 0
pixellines = []
while x < region.size[0]:
    currentline = []
    y = 0
    while y < region.size[1]:
        pixel=(x,y)
        currentline.append(region.getpixel(pixel))
        y = y + 1
    pixellines.append(currentline)
    x = x + 1

# Process pixel data
x = 0
lineholder = []
while x < 1:
    plotlinex = []
    plotliney = []
    y = 0
    offset = 0
    while y < region.size[1]:
        plotlinex.append(y * magnifiedpixel)
        if y > 3:
            if pixellines[x][y] > pixellines[x][y - 1] and pixellines[x][y] > pixellines[x][y - 2] and
pixellines[x][y] > pixellines[x][y - 3] and pixellines[x][y] > pixellines[x][y - 4] and
abs(pixellines[x][y] - pixellines[x][y - 1]) > 15:
                offset = offset + halfwavelength
            elif y > 0 and abs(pixellines[x][y] - pixellines[x][y - 1]) > 20:
                offset = offset + halfwavelength
            flipped = 255 - pixellines[x][y]
            #print w + 1, flipped, pixellines[x][y], offset, (flipped + 1) * heightstep + offset
            plotliney.append((flipped + 1) * heightstep + offset)
            y = y + 1

```

```

x = x + 1
lineholder.append(plotline $x$ )
lineholder.append(plotline $y$ )

# Get offset value
currentoffset = offsets.readline()

# Base offset
i = 1
j = 0
baseoffset = lineholder[i][j]
while j < len(lineholder[i]):
    lineholder[i][j] = lineholder[i][j] - baseoffset
    j = j + 1

# Secondary offset
if w == int(firstnumber) - 1:
    i = 0
    while i < len(lineholder[1]):
        secondoffset.append(lineholder[1][i])
        i = i + 1
i = 1
j = 0
while j < len(lineholder[i]):
    lineholder[i][j] = lineholder[i][j] - secondoffset[j]
    j = j + 1

# Rotation baseline
coefs = polyfit(lineholder[0],lineholder[1],3)
slope = coefs[2]
theta = arctan(slope)

# Rotate function
m = 0
x_0 = lineholder[0][0]
y_0 = lineholder[1][0]
while m < len(lineholder[0]):
    r = sqrt((lineholder[0][m]-x_0)*(lineholder[0][m]-x_0) + (lineholder[1][m]-
y_0)*(lineholder[1][m]-y_0))
    if lineholder[0][m]-x_0 > 0:
        phi = arctan(float(lineholder[1][m]-y_0)/float(lineholder[0][m]-x_0))

```

```

    angle = phi - theta
    lineholder[0][m] = r * cos(angle)
    lineholder[1][m] = r * sin(angle)
else:
    lineholder[0][m] = x_0
    lineholder[1][m] = y_0
m = m + 1

# Find the coefficients of the cubic curve through the points
#coefs = rpolyfit(lineholder[0],lineholder[1])
coefs = polyfit(lineholder[0],lineholder[1],3)

# Best fit the curve
bestfit_x = []
bestfit_y = []
q = 0
while q < len(lineholder[0]):
    x_val = lineholder[0][q]
    bestfit_x.append(x_val) # Append the x values to the list
    y_val = coefs[0]*x_val*x_val*x_val + coefs[1]*x_val*x_val + coefs[2]*x_val + coefs[3] #
Calculate the best fit y value
    #y_val = coefs[0]*x_val*x_val*x_val + coefs[1]*x_val*x_val # line for rpolyfit
    bestfit_y.append(y_val) # Append the best fit y values to the list
    q = q + 1

# Subtract macro y offset
i = 1
j = 0
while j < len(bestfit_y):
    bestfit_y[j] = bestfit_y[j] - float(currentoffset)
    j = j + 1

#Output force-indentation data
beamdefl = abs(bestfit_y[len(bestfit_y)-1] - bestfit_y[0])
force = beamdefl * 3 * E * I / (beamlength * beamlength * beamlength) # Force caused by this
much deflection in the beam (kg*micrometer)/(s^2)
indent = abs(bestfit_y[len(bestfit_y)-1])

forces.append(force)
indents.append(indent)

#plot(bestfit_x,bestfit_y)
#plot(lineholder[0],lineholder[1])

```



```

# Element offset
skipentries = len(bestfit_x) / 18.0

# Open file for writing
filename = basefilename + str(w+1) + ".log"
FILE = open(filename, "w")

# Empty commands list
commands = []

# Initial starting commands
commands.append("/CLEAR,NOSTART\n")
commands.append("/FILENAME,MQP_Trial3,0\n")

# Format the output in the table
commands.append("/FORMAT,7,G,17,10, ,\n")
commands.append("/GFORMAT,7,G,17,10, ,\n")

# Open preprocessor
commands.append("/PREP7\n")

# Add element type
commands.append("ET,1,BEAM3\n")

# Assign beam properties
beamprops = "R,1," + str(area) + "," + str(I) + "," + str(beamheight) + ", , , \n"
commands.append(beamprops)

# Assign material properties
commands.append("MPTEMP,,,,,,,,\n")
commands.append("MPTEMP,1,0\n")
commands.append("MPDATA,EX,1,,169000\n")
commands.append("MPDATA,PRXY,1,,0.17\n")

# Create nodes

```

```

i = 0
while i <= numelements:
    if i == 0:
        value = 0
    else:
        value = int(skipentries * i) - 1
    command = "N, ," + str(bestfit_x[value]) + "0,0,,,,\n"
    commands.append(command)
    i = i + 1

# Create elements
j = 0
while j < numelements:
    command = "FLST,2,2,1\nFITEM,2," + str(j+1) + "\nFITEM,2," + str(j+2) + "\nE,P51X\n"
    commands.append(command)
    j = j + 1

# Apply y constraints to the nodes
i = 0
while i <= numelements:
    if i == 0:
        value = 0
    else:
        value = int(skipentries * i) - 1
    ydisp = (bestfit_y[value])
    command = "FLST,2,1,1,ORDE,1\nFITEM,2," + str(i + 1) + "\n!\n/GO\nD,P51X, ," +
str(ydisp) + ", , , ,UY, , , ,\n"
    commands.append(command)
    i = i + 1

# Solve the problem
commands.append("FINISH\n")
commands.append("/SOL\n")
commands.append("/STATUS,SOLU\n")
commands.append("SOLVE\n")

# Zoom to fit screen
commands.append("/AUTO,1\n")
commands.append("/REP,FAST\n")

# Plot deformed shape

```

```
commands.append("FINISH\n")
commands.append("/POST1\n")
commands.append("PLDISP,1\n")
```

```
# Close file
FILE.writelines(commands)
FILE.close()
```

```
w = w + 1
```

```
# Process the images for retraction
w = int(firstnumber2) - 1
while(w < int(numofimages2) + int(firstnumber2) - 1):
```

```
    # Load image file
    if ((w + 1) < 10):
        im = Image.open(basefilename2 + "000" + str(w + 1) + ". bmp")
    elif ((w + 1) < 100):
        im = Image.open(basefilename2 + "00" + str(w + 1) + ". bmp")
    elif ((w + 1) < 1000):
        im = Image.open(basefilename2 + "0" + str(w + 1) + ". bmp")
    else:
        im = Image.open(basefilename2 + str(w + 1) + ". bmp")
```

```
# Crop the image and store it in the region
region = im.crop(box)
region = region.convert("L")
```

```
# Take all the values of the pixels and store them in the matrix
x = 0
pixellines = []
while x < region.size[0]:
    currentline = []
    y = 0
    while y < region.size[1]:
        pixel=(x,y)
        currentline.append(region.getpixel(pixel))
        y = y + 1
    pixellines.append(currentline)
    x = x + 1
```

```

    pixellines. append(currentline)
    x = x + 1

# Process pixel data
x = 0
lineholder = []
while x < 1:
    plotline = []
    plotliney = []
    y = 0
    offset = 0
    while y < region. size[1]:
        plotline. append(y * magnifiedpixel)
        if y > 3:
            if pixellines[x][y] > pixellines[x][y - 1] and pixellines[x][y] > pixellines[x][y - 2] and
pixellines[x][y] > pixellines[x][y - 3] and pixellines[x][y] > pixellines[x][y - 4] and
abs(pixellines[x][y] - pixellines[x][y - 1]) > 15:
                offset = offset + halfwavelength
            elif y > 0 and abs(pixellines[x][y] - pixellines[x][y - 1]) > 20:
                offset = offset + halfwavelength
            flipped = 255 - pixellines[x][y]
            #print w + 1, flipped, pixellines[x][y], offset, (flipped + 1) * heightstep + offset
            plotliney. append((flipped + 1) * heightstep + offset)
            y = y + 1
        x = x + 1
    lineholder. append(plotline)
    lineholder. append(plotliney)

# Get offset value
currentoffset = offsets2. readline()

# Base offset
i = 1
j = 0
baseoffset = lineholder[i][j]
while j < len(lineholder[i]):
    lineholder[i][j] = lineholder[i][j] - baseoffset
    j = j + 1

# Secondary offset
i = 1
j = 0

```

```

while j < len(lineholder[i]):
    lineholder[i][j] = lineholder[i][j] - secondoffset[j]
    j = j + 1

# Rotation baseline
coefs = polyfit(lineholder[0],lineholder[1],3)
slope = coefs[2]
theta = arctan(slope)

# Rotate function
m = 0
x_0 = lineholder[0][0]
y_0 = lineholder[1][0]
while m < len(lineholder[0]):
    r = sqrt((lineholder[0][m]-x_0)*(lineholder[0][m]-x_0) + (lineholder[1][m]-
y_0)*(lineholder[1][m]-y_0))
    if lineholder[0][m]-x_0 > 0:
        phi = arctan(float(lineholder[1][m]-y_0)/float(lineholder[0][m]-x_0))
        angle = phi - theta
        lineholder[0][m] = r * cos(angle)
        lineholder[1][m] = r * sin(angle)
    else:
        lineholder[0][m] = x_0
        lineholder[1][m] = y_0
    m = m + 1

# Find the coefficients of the cubic curve through the points
#coefs = rpolyfit(lineholder[0],lineholder[1])
coefs = polyfit(lineholder[0],lineholder[1],3)

# Best fit the curve
bestfit_x = []
bestfit_y = []
q = 0
while q < len(lineholder[0]):
    x_val = lineholder[0][q]
    bestfit_x.append(x_val) # Append the x values to the list
    y_val = coefs[0]*x_val*x_val*x_val + coefs[1]*x_val*x_val + coefs[2]*x_val + coefs[3] #
Calculate the best fit y value
    #y_val = coefs[0]*x_val*x_val*x_val + coefs[1]*x_val*x_val # line for rpolyfit
    bestfit_y.append(y_val) # Append the best fit y values to the list
    q = q + 1

```

```

# Subtract macro y offset
i = 1
j = 0
while j < len(bestfit_y):
    bestfit_y[j] = bestfit_y[j] - float(currentoffset)
    j = j + 1

#Output force-indentation data
beamdefl = abs(bestfit_y[len(bestfit_y)-1] - bestfit_y[0])
force = beamdefl * 3 * E * I / (beamlength * beamlength * beamlength) # Force caused by this
much deflection in the beam (kg*micrometer)/(s^2)
indent = abs(bestfit_y[len(bestfit_y)-1])

forces. append(force)
indents. append(indent)

#plot(bestfit_x,bestfit_y)
#plot(lineholder[0],lineholder[1])

# Element offset
skipentries = len(bestfit_x) / 18.0

# Open file for writing
filename = basefilename2 + str(w+1) + ".log"
FILE = open(filename,"w")

# Empty commands list
commands = []

# Initial starting commands
commands. append("/CLEAR,NOSTART\n")
commands. append("/FILENAME,MQP_Trial3,0\n")

# Format the output in the table
commands. append("/FORMAT,7,G,17,10, ,\n")
commands. append("/GFORMAT,7,G,17,10, ,\n")

```

```

# Open preprocessor
commands.append("/PREP7\n")

# Add element type
commands.append("ET,1,BEAM3\n")

# Assign beam properties
beamprops = "R,1," + str(area) + "," + str(I) + "," + str(beamheight) + ", , , \n"
commands.append(beamprops)

# Assign material properties
commands.append("MPTEMP,,,,,,,,\n")
commands.append("MPTEMP,1,0\n")
commands.append("MPDATA,EX,1,,169000\n")
commands.append("MPDATA,PRXY,1,,0.17\n")

# Create nodes
i = 0
while i <= numelements:
    if i == 0:
        value = 0
    else:
        value = int(skipentries * i) - 1
    command = "N, ," + str(bestfit_x[value]) + "0,0,,,,\n"
    commands.append(command)
    i = i + 1

# Create elements
j = 0
while j < numelements:
    command = "FLST,2,2,1\nFITEM,2," + str(j+1) + "\nFITEM,2," + str(j+2) + "\nE,P51X\n"
    commands.append(command)
    j = j + 1

# Apply y constraints to the nodes
i = 0
while i <= numelements:
    if i == 0:
        value = 0
    else:

```

```

        value = int(skipentries * i) - 1
        ydisp = (bestfit_y[value])
        command = "FLST,2,1,1,ORDE,1\nFITEM,2," + str(i + 1) + "\n!* \n/GO\nD,P51X, ," +
str(ydisp) + ", , , ,UY, , , , \n"
        commands. append(command)
        i = i + 1

```

```

# Solve the problem
commands. append("FINISH\n")
commands. append("/SOL\n")
commands. append("/STATUS,SOLU\n")
commands. append("SOLVE\n")

```

```

# Zoom to fit screen
commands. append("/AUTO,1\n")
commands. append("/REP,FAST\n")

```

```

# Plot deformed shape
commands. append("FINISH\n")
commands. append("/POST1\n")
commands. append("PLDISP,1\n")

```

```

# Close file
FILE. writelines(commands)
FILE. close()

```

```

w = w + 1

```

```

FILE = open("curve. csv",'w')
i = 0
while i < len(forces):
    FILE. writelines(str(indents[i]) + "," + str(forces[i]) + "\n")
    i = i + 1
FILE. close()

```

```

# End of program

```


12.3.2. ANSYSMacroCreator. py

```
import Image
import ImageFilter
from numpy import *
from pylab import *
import datetime
import os

# Output header
today = datetime. date. today()
print "Anthony DiOrio, Russell Morin, and Eric Wilusz\n", today, "\nNanoindenter
MQP\nWorcester Polytechnic Institute\nMechanical Engineering\n"

# Empty commands list
commands = []

# Set base directory
commands. append("/CWD,'E:\\Documents and Settings\\rmorin\\Desktop\\MQPTrial10\n")

# Initial starting commands
commands. append("/CLEAR,NOSTART\n")
commands. append("/CLE\n")

# Get base filename
basename = raw_input("Please enter the base filename: ")

# Open file for writing
filename = str(basename) + ". log"
FILE = open(filename,"w")

# Open output file
commands. append("*CFOPEN," + str(basename) + ",csv\n")

# Get number of files
numfiles = input("Please enter the number of files to be processed: ")

# Get number of first image
firstimage = input("Please enter the number of the first image: ")
```

```

# Input ANSYS files
a = firstimage - 1
while a < numfiles + firstimage - 1:

    # Clear ANSYS
    commands.append("/CLEAR,NOSTART\n")
    commands.append("/CLE\n")

    # Input files to be processed
    commands.append("/INPUT," + basename + str(a + 1) + ",log\n")

    # Create arrays to store the data
    if a == firstimage - 1:
        commands.append("*DIM,DISPARRAY,ARRAY,1,1\n")
        commands.append("*DIM,FORCEARRAY,ARRAY,1,1\n")

    # Write reaction force and displacement at node to arrays
    commands.append("*VGET,DISPARRAY(1,1),NODE,19,U,Y, , ,2\n")
    commands.append("*VGET,FORCEARRAY(1,1),NODE,19,RF,FY\n")

    # Write the array values to the file
    commands.append("writer,'DISPARRAY',1,1\n")
    commands.append("writer,'FORCEARRAY',1,1\n")

    # Clear up the nodes and elements already existing
    commands.append("FINISH\n")
    commands.append("/PREP7\n")
    commands.append("FLST,2,18,2,ORDER,2\n")
    commands.append("FITEM,2," + str(((a - firstimage + 1) * 18 + 1)) + "\n")
    commands.append("FITEM,2," + str(-18 * ((a - firstimage + 1) + 1)) + "\n")
    commands.append("EDELE,P51X\n")

    # Write commands to file
    FILE.writelines(commands)

# Clear commands

```

```
commands = []

# Increment loop
a = a + 1

# Close output file
commands.append("*CFCLOS\n")
FILE.writelines(commands)

# Close file
FILE.close()

# End of Program
```

12.3.3. ConverttoCSV. py

```
import string

from string import *

# Convert Output. csv to actually be a CSV file
basename = raw_input("Enter the base filename: ")

FILE = open(str(basename) + ". csv",'r')

output = FILE. readlines()

FILE. close()

newlines = []

i = 0
while i < len(output):
    newlines. append(lstrip(rstrip(output[i],"\\n"),' -') + "," + lstrip(rstrip(output[i+1],"\\n"),' -') +
"\\n")
    i = i + 2

FILE = open(str(basename) + ". csv",'w')

FILE. writelines(newlines)

FILE. close()
```

12.3.4. RPolyFit. py

New polynomial fitting function

This polyfit function fits the points given to a curve where $y = a_1x^2 + a_0x^3$

```
def rpolyfit(xvals, yvals):
```

```
    i = 0
```

```
    A = 0
```

```
    while i < len(yvals):
```

```
        A = A + yvals[i]
```

```
        i = i + 1
```

```
    i = 0
```

```
    B = 0
```

```
    while i < len(xvals):
```

```
        B = B + xvals[i]*xvals[i]
```

```
        i = i + 1
```

```
    i = 0
```

```
    C = 0
```

```
    while i < len(xvals):
```

```
        C = C + xvals[i]*xvals[i]*xvals[i]
```

```
        i = i + 1
```

```
    i = 0
```

```
    D = 0
```

```
    while i < len(xvals):
```

```
        D = D + xvals[i]*yvals[i]
```

```
        i = i + 1
```

```
    i = 0
```

```
    E = 0
```

```
    while i < len(xvals):
```

```
        E = E + xvals[i]*xvals[i]*xvals[i]*xvals[i]
```

```
        i = i + 1
```

```
    a1 = (B*D-C*A)/(B*E-C*C)
```

```
    a0 = (A-a1*C)/B
```

```
    coefs = []
```

```
    coefs.append(a1)
```

```
    coefs.append(a0)
```

```
    return coefs
```

12.3.5. MQPModel. py

```
import Image
import ImageFilter
from numpy import *
from pylab import *
import datetime
import os

# Load image file
im = Image.open("afmtip1_1. bmp")

# Function for beam displacement
def disp(x, force, E, I, length):
    y = (force * x * x * (3 * length - x))/(6 * E * I)
    return y

# Indentation depth as a function of force
def depth(force, Estar, alpha):
    h = sqrt(force * pi / 2 / Estar / tan(alpha))
    return h

# Output header
today = datetime. date. today()
print "Anthony DiOrio, Russell Morin, and Eric Wilusz\n", today, "\nNanoindenter
MQP\nWorcester Polytechnic Institute\nMechanical Engineering\n"

# Beam mechanical and material properties
beamlength = 160. 0 # micrometers

beamwidth = 37. 5 # micrometers

beamheight = 7. 0 # micrometers

area = beamwidth * beamheight # micrometers^2

E = 169000. 0 # kg/(s^2*micrometer)

alpha1 = 15. 00 * pi / 180

alpha2 = 17. 50 * pi / 180
```

```

tipangle = 16.25 * pi / 180 # Average tip angle to model as a conical indenter

alphaprime = tipangle + pi / 18 # Assumed value of alphaprime for simulation purposes

nusilicon = 0.17

I = (1/12.0) * beamwidth * (beamheight*beamheight*beamheight) # micrometers^4

# Sample material properties
Ealum = 68000.0 # kg/(s^2*micrometer)

nualum = 0.29

# Find composite modulus
Estar = 1 / ( (1 - nualum * nualum) / Ealum + (1 - nusilicon * nusilicon) / E ) #
kg/(s^2*micrometer)

# Initial values of piezo position and indentation depth (assume we begin at contact point)
piezoposition = 0.0 # micrometers
indentationdepth = 0.0 # micrometers

# Maximum depth to indent
#maxindentation = 0.100 # micrometers

# Maximum movement
maxmovement = 2.6

# Resolution of the piezo feedback
piezostep = 0.1 # micrometers (5 nanometer resolution)

# Number of elements to be used, determined by validation exercises
numelements = 18

# Filename setup
basefilename = os.getcwd() + "\\ANSYS
Files\\SimulationStepped\\MQP_SimulationIndentationStep"

```

```

# Max force and depth variables initialized
maxforce = 0.0
maxdepth = 0.0

a = 0
forcecontact = []
indent = []
while piezoposition <= maxmovement:

    # Open file for writing
    filename = basefilename + str(a+1) + ".log"
    FILE = open(filename,"w")

    # Calculate force and depth at given step
    beamdefl = piezoposition - indentationdepth
    force = beamdefl * 3 * E * I / (beamlength * beamlength * beamlength) # Force caused by this
    much deflection in the beam (kg*micrometer)/(s^2)
    indentationdepth = depth(force, Estar, tipangle) # Calculate the new indentation depth based
    on theory

    # Output point to be plotted
    forcecontact.append(force)
    indent.append(indentationdepth)

    # Empty commands list
    commands = []

    # Initial starting commands
    commands.append("/CLEAR,NOSTART\n")
    commands.append("/FILENAME,MQP_Validation,0\n")

    # Format the output in the table
    commands.append("/FORMAT,7,G,17,10, ,\n")
    commands.append("/GFORMAT,7,G,17,10, ,\n")

    # Open preprocessor
    commands.append("/PREP7\n")

    # Add element type

```



```

commands.append("ET,1,BEAM3\n")

# Assign beam properties
beamprops = "R,1," + str(area) + "," + str(I) + "," + str(beamheight) + ", , , \n"
commands.append(beamprops)

# Assign material properties
commands.append("MPTEMP,,,,,,,,\n")
commands.append("MPTEMP,1,0\n")
commands.append("MPDATA,EX,1,,169000\n")
commands.append("MPDATA,PRXY,1,,0.17\n")

# Create nodes
i = 0
while i <= numelements:
    command = "N, , " + str(beamlength / numelements * i) + "0,0,,,,\n"
    commands.append(command)
    i = i + 1

# Create elements
j = 0
while j < numelements:
    command = "FLST,2,2,1\nFITEM,2," + str(j+1) + "\nFITEM,2," + str(j+2) + "\nE,P51X\n"
    commands.append(command)
    j = j + 1

# Apply y constraints to the nodes
i = 0
while i <= numelements:
    ydisp = -piezoposition + disp(beamlength / numelements * i, force, E, I, beamlength)
    command = "FLST,2,1,1,ORDE,1\nFITEM,2," + str(i + 1) + "\n!* \n/GO\nD,P51X, , " +
str(ydisp) + ", , , UY, , , , \n"
    commands.append(command)
    i = i + 1

# Solve the problem
commands.append("FINISH\n")
commands.append("/SOL\n")
commands.append("/STATUS,SOLU\n")
commands.append("SOLVE\n")

```

```

# Zoom to fit screen
commands.append("/AUTO,1\n")
commands.append("/REP,FAST\n")

# Plot deformed shape
commands.append("FINISH\n")
commands.append("/POST1\n")
commands.append("PLDISP,1\n")

# Write commands to file
FILE.writelines(commands)

# Close file
FILE.close()

# Increment piezo position
piezoposition = piezoposition + piezostep
a = a + 1

# Update max force and depth
if force > maxforce:
    maxforce = force
if indentationdepth > maxdepth:
    maxdepth = indentationdepth

# Calculate parameters necessary for retraction calculations
eps = 2 * (pi - 2) / pi
hc = maxdepth / ((pi / 2 - 1) * sqrt(2 * tan(alpha1) * tan(alpha2) / pi) / tan(alphaprime) + 1)
hr = ((1 - 2 / eps) * maxdepth + 2 / eps * hc)

# Filename setup
basefilename = os.getcwd() + "\\ANSYS
Files\\SimulationStepped\\MQP_SimulationRetractionStep"

a = 0
while piezoposition >= 0:

    # Open file for writing

```

```

filename = basefilename + str(a+1) + ". log"
FILE = open(filename,"w")

# Calculate force and depth at given step
beamdefl = piezoposition - indentationdepth
force = beamdefl * 3 * E * I / (beamlength * beamlength * beamlength) # Force caused by this
much deflection in the beam (kg*micrometer)/(s^2)
indentationdepth = sqrt(force / maxforce) * (maxdepth - hr) + hr

# Ouput points to be plotted
forcecontact. append(force)
indent. append(indentationdepth)

# Empty commands list
commands = []

# Initial starting commands
commands. append("/CLEAR,NOSTART\n")
commands. append("/FILENAME,MQP_Validation,0\n")

# Format the output in the table
commands. append("/FORMAT,7,G,17,10, ,\n")
commands. append("/GFORMAT,7,G,17,10, ,\n")

# Open preprocessor
commands. append("/PREP7\n")

# Add element type
commands. append("ET,1,BEAM3\n")

# Assign beam properties
beamprops = "R,1," + str(area) + "," + str(I) + "," + str(beamheight) + ", , , ,\n"
commands. append(beamprops)

# Assign material properties
commands. append("MPTEMP,,,,,,,,\n")
commands. append("MPTEMP,1,0\n")

```

```

commands. append("MPDATA,EX,1,,169000\n")
commands. append("MPDATA,PRXY,1,,0. 17\n")

# Create nodes
i = 0
while i <= numelements:
    command = "N, ," + str(beamlength / numelements * i) + "0,0,,,,,\n"
    commands. append(command)
    i = i + 1

# Create elements
j = 0
while j < numelements:
    command = "FLST,2,2,1\nFITEM,2," + str(j+1) + "\nFITEM,2," + str(j+2) + "\nE,P51X\n"
    commands. append(command)
    j = j + 1

# Apply y constraints to the nodes
i = 0
while i <= numelements:
    ydisp = -piezoposition + disp(beamlength / numelements * i, force, E, I, beamlength)
    command = "FLST,2,1,1,ORDE,1\nFITEM,2," + str(i + 1) + "\n!* \n/GO\nD,P51X, ," +
str(ydisp) + ", , , ,UY, , , ,\n"
    commands. append(command)
    i = i + 1

# Solve the problem
commands. append("FINISH\n")
commands. append("/SOL\n")
commands. append("/STATUS,SOLU\n")
commands. append("SOLVE\n")

# Zoom to fit screen
commands. append("/AUTO,1\n")
commands. append("/REP,FAST\n")

# Plot deformed shape
commands. append("FINISH\n")
commands. append("/POST1\n")
commands. append("PLDISP,1\n")

```

```
# Write commands to file  
FILE. writelines(commands)
```

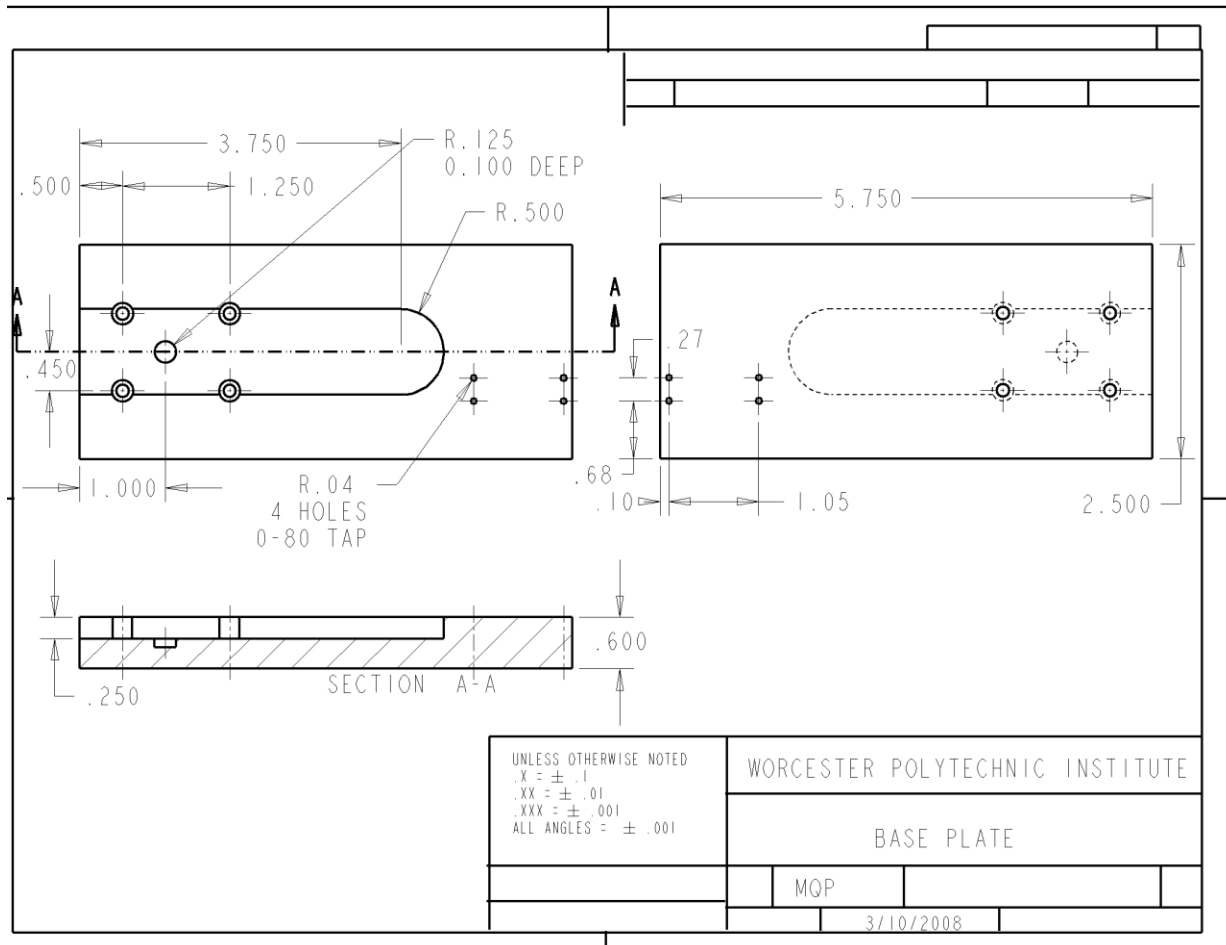
```
# Close file  
FILE. close()
```

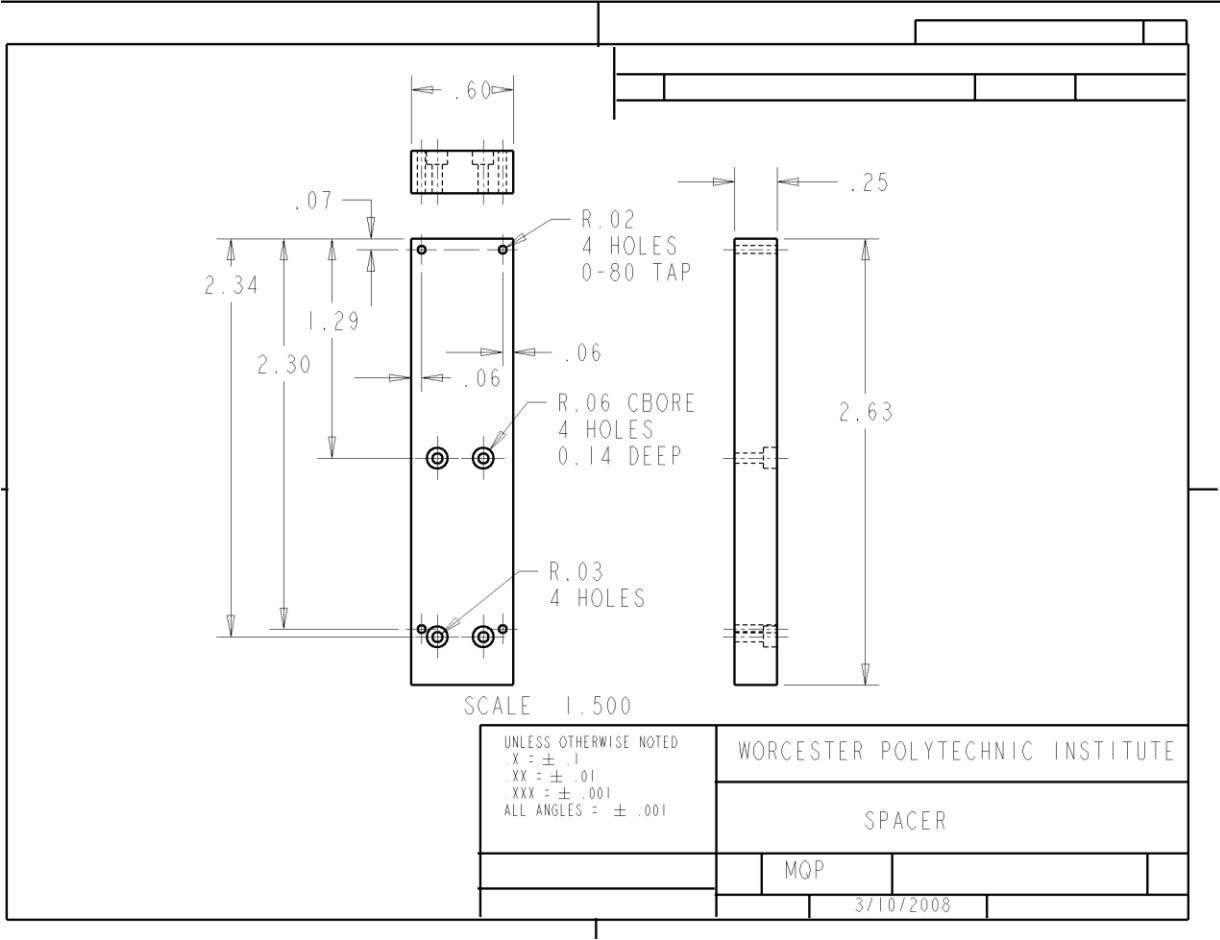
```
# Increment piezo position  
piezoposition = piezoposition - piezostep  
a = a + 1
```

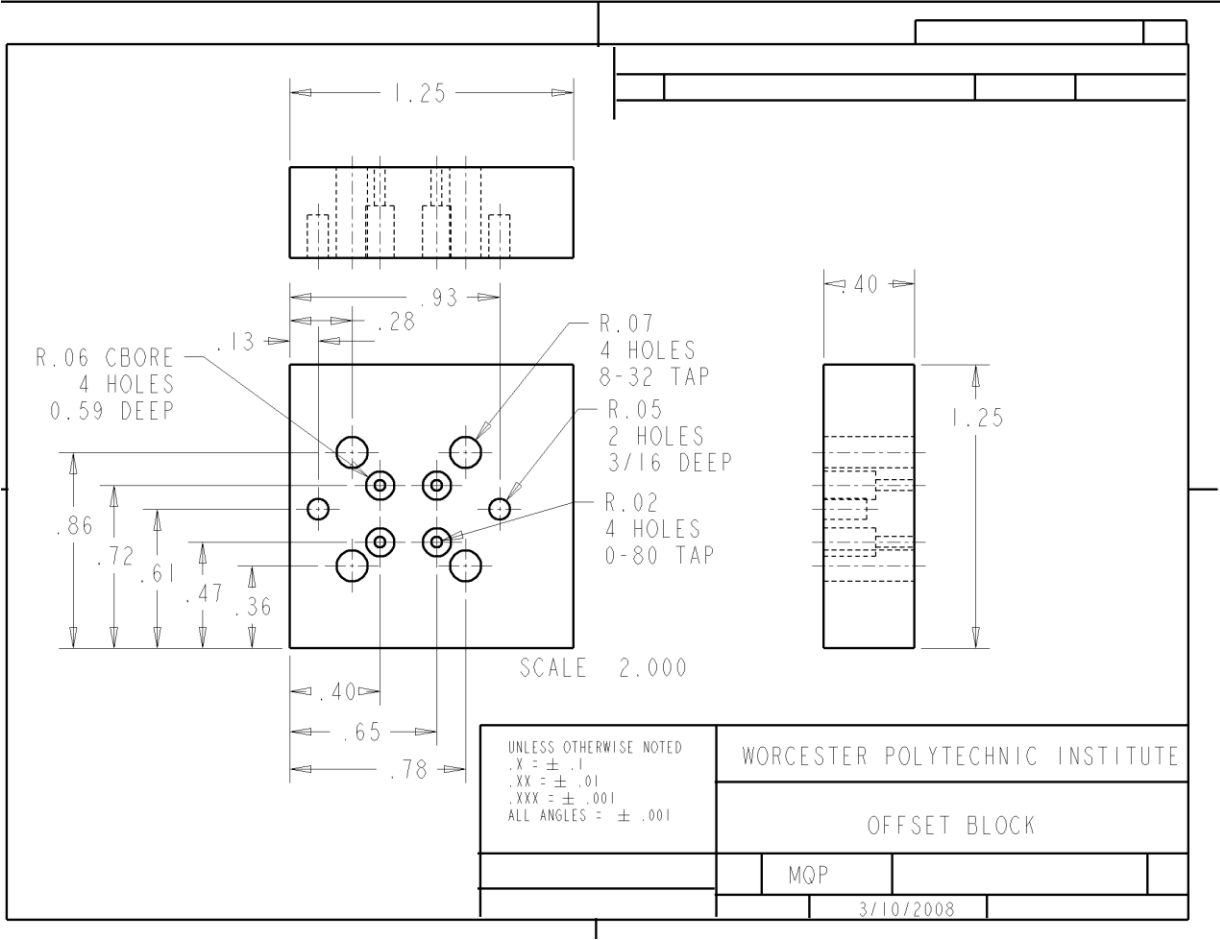
```
plot(indent, forcecontact)  
show()
```

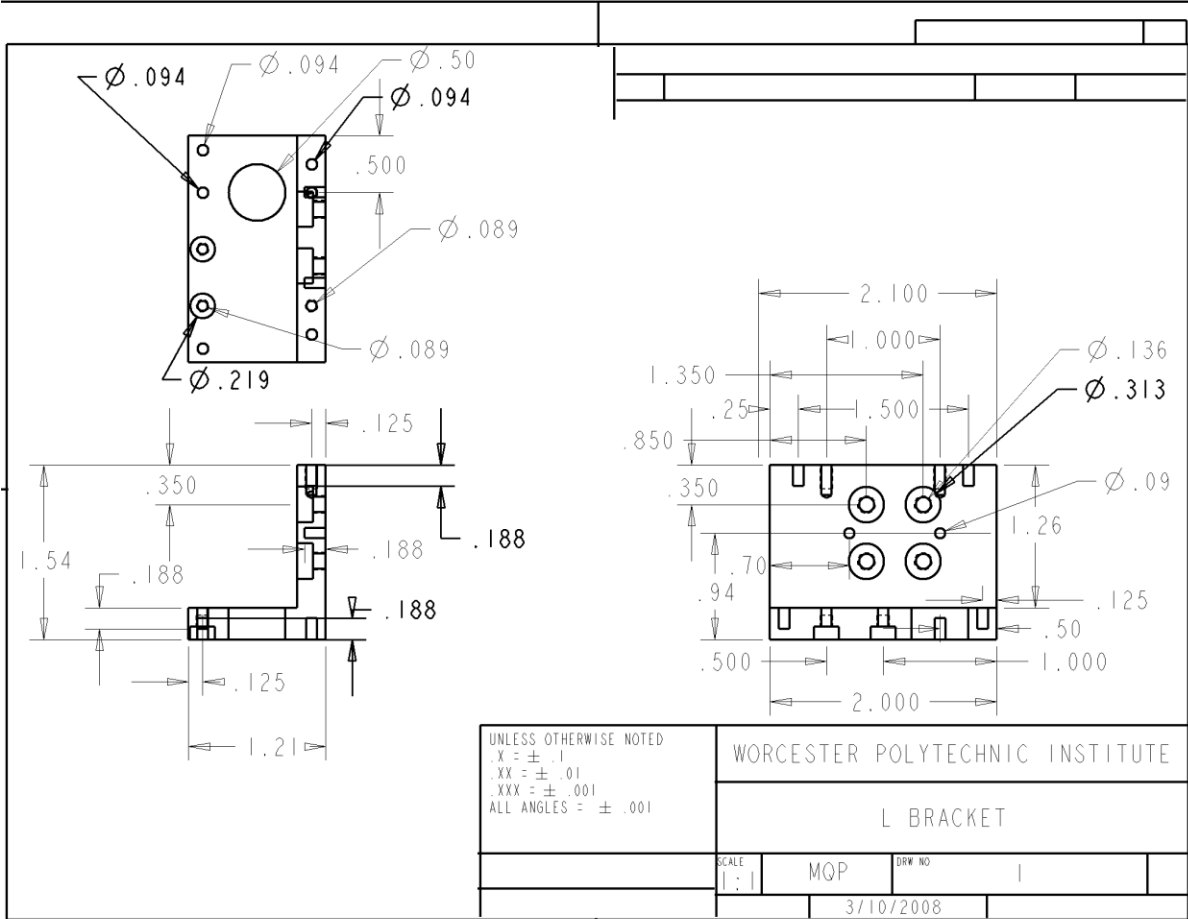
```
# End of Program
```

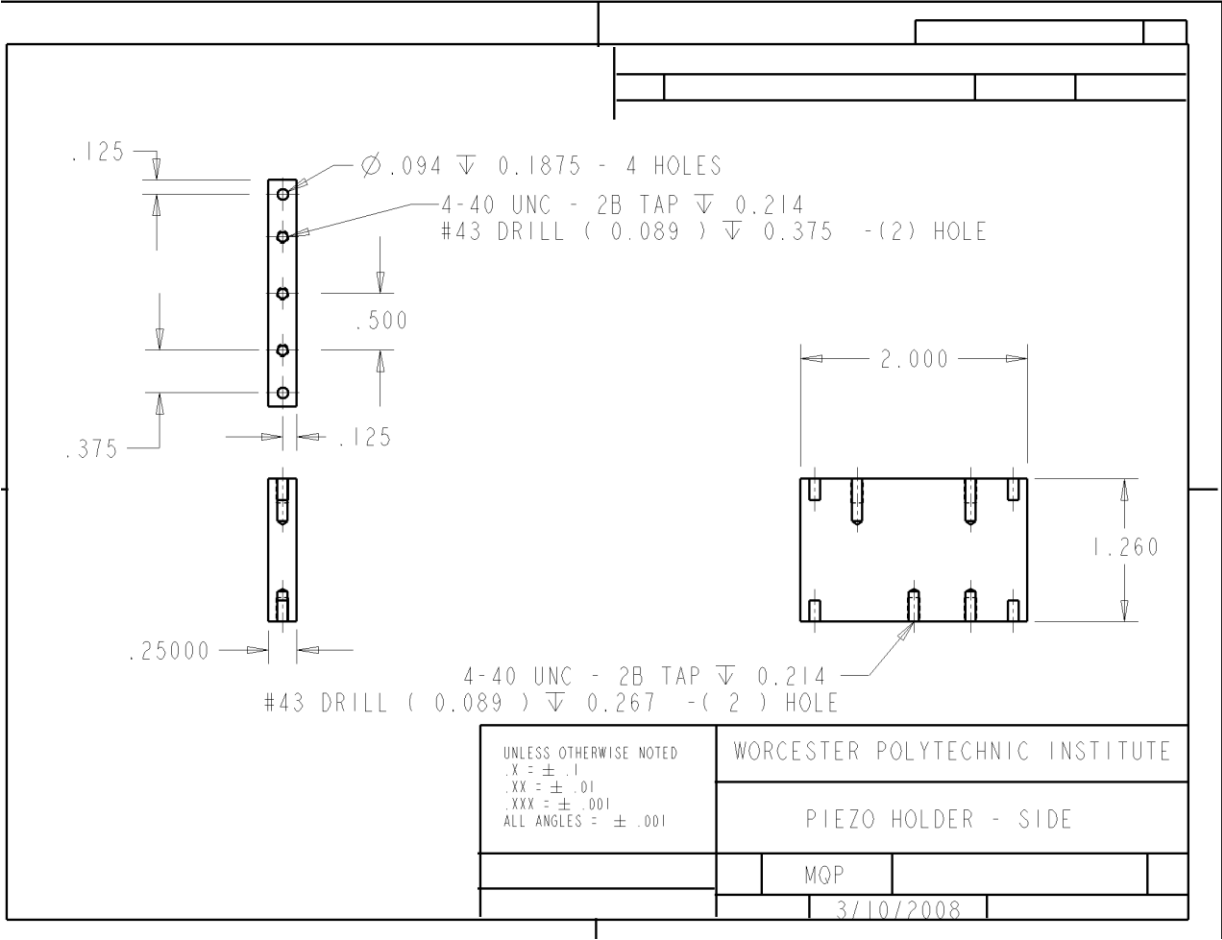
12.4. Appendix D – Drawings of Manufactured Components

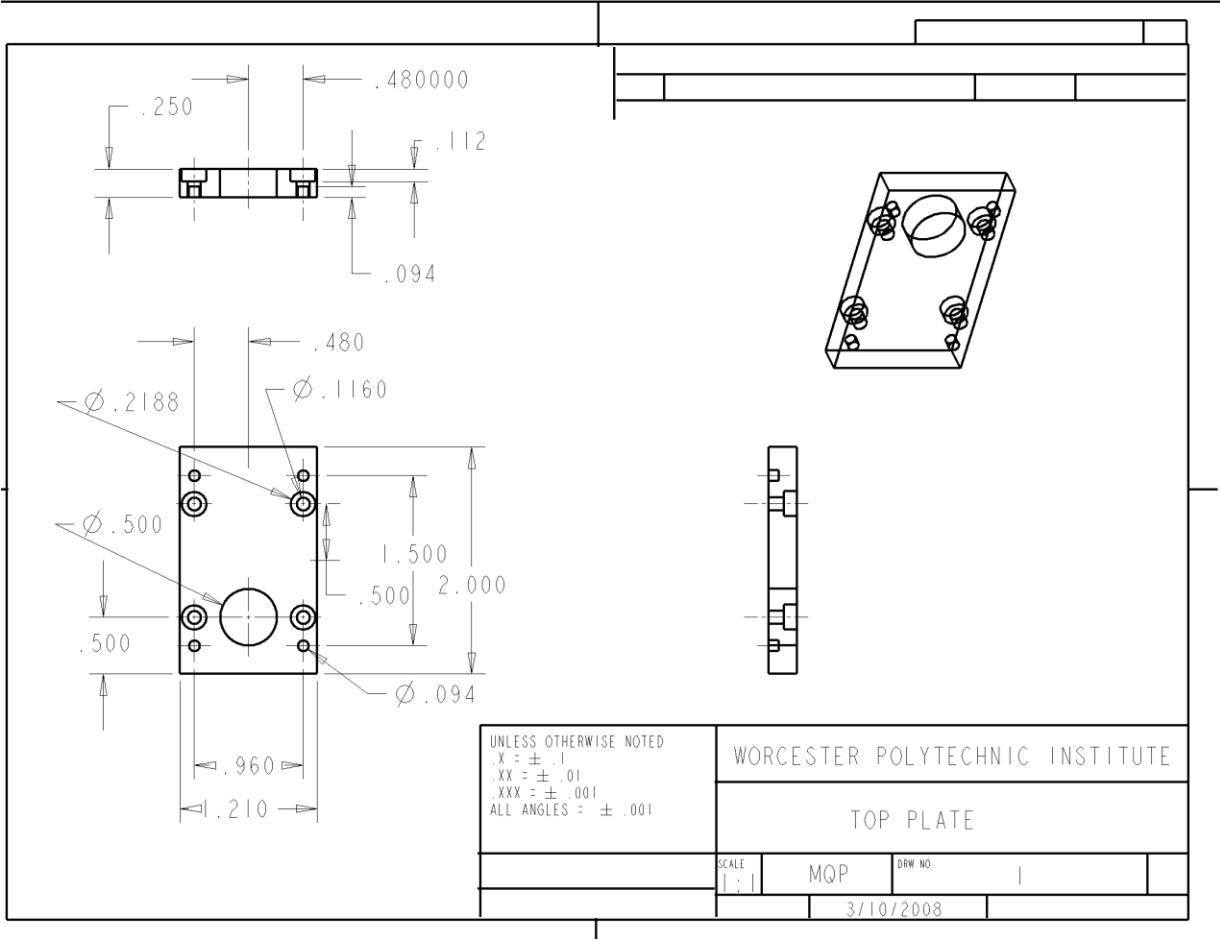


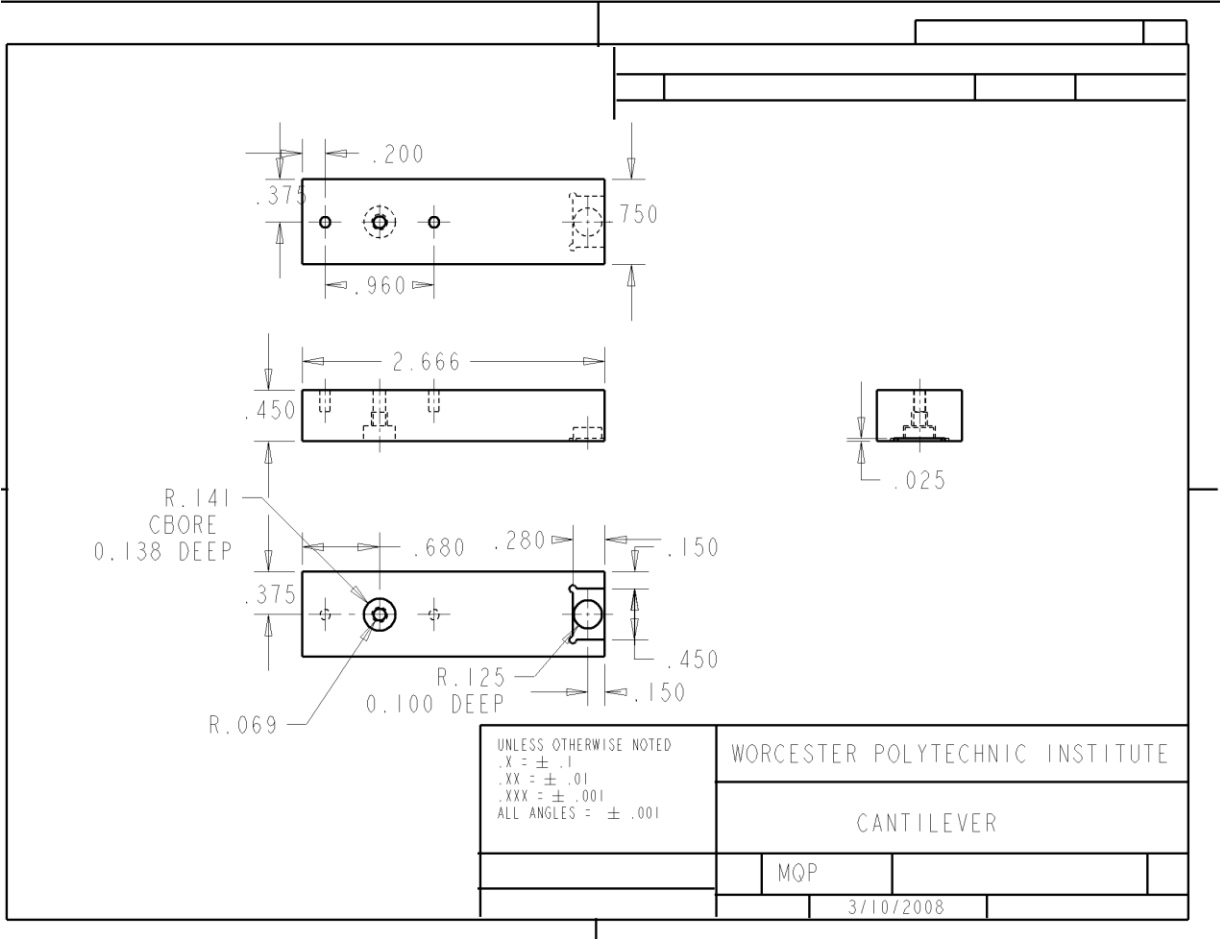












12.5. Appendix E – Attached Files

PythonScripts.zip

ConverttoCSV.py

RPolyFit.py

MQPModel.py

IndentationImageProcessor.py

ANSYSMacroCreator.py

Writer.mac

NanoindentationMQP.doc

NanoindentationMQP.pdf

CAD Files.zip

Cantilever.drw

Cantilever.prt

CantileverAssembly.asm

Design_Plate.drw

Design_Plate.prt

LongAlignmentPin.prt

Magnet.prt

MQPV2.asm

NAI-MM-3M-F-1_0.prt

OffsetBlock.drw

OffsetBlock.prt

Piezo.prt

Piezo_Holder_L.drw

Piezo_Holder_L.prt

PiezoAssembly.asm

PiezoHolderSide.drw

PiezoHolderSide.prt

Spacer.drw

Spacer.prt

Top_Plate_Drawing.drw

TopPlate.prt

XYStage.prt

Z_Bracket1.prt

Additional Project Files

Data Files