Lauren Yates 10195969

# CISC 330 Assignment 3

## Part 1: Operating Room Use

Drill Bit Tracking: math written on paper and submitted in pdf format.

Targeting Error: math written on paper and submitted in pdf format.

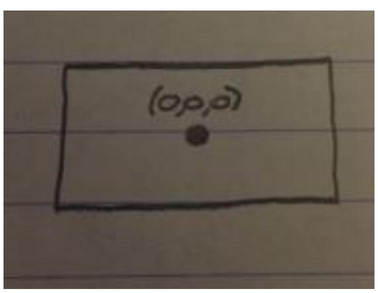
## Part 2: Calibration

### Drill Tip Calibrator

Drill Tip Calibration Steps and Diagrams

** Note: hand drawn diagrams are not necessarily accurate depictions, use as a visual aid to understand method **

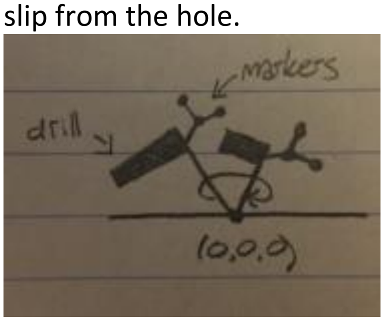
The method chosen for the drill tip calibration was the sphere fitting calibration. This method uses the generation of 3 concentric spheres associated with the three markers rotated around one point. The average of these spheres centers is the center in tracker coordinates, which can be converted to marker coordinates using the associated rotation matrix and translation vector. This method is beneficial as it can use multiple poses to find a more accurate calibration of the drill. The restriction for this method is that it rotates around one center point, meaning that it's important to keep the drill steady and in one place otherwise if the drill slips or moves the spheres generated will no longer be accurate and the calibration will not be accurate.




Step 1: drill a small divot into the wooden bench top as a slot for the drill tip to sit and mark where the hole is as the coordinate (0,0,0)

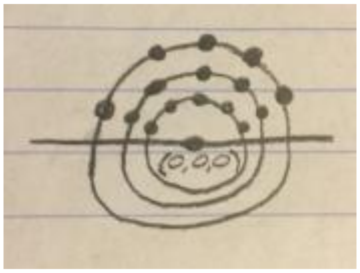
Step 2: Place drill tip in the hole in the wooden bench top.

Step 3: track the markers and pivot the drill in the divot/hole in the bench, ensuring it does not slip from the hole.




Step 4: capture a set number of poses from the rotations/pivots and build a matrix containing all of the poses in order (A1,B1,C1,A2,B2,C2,…An,Bn,Cn).

Step 5: Separate the matrix into 3 smaller matrices containing all As, all Bs, and all Cs.



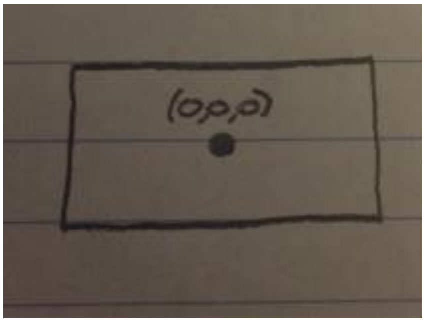

Step 6: calculate the spheres and their centers associated with the data sets of As, Bs, and Cs (3 concentric spheres)
Step 7: Calculate the average of the three centers of the spheres and use this as the pivot point/drill tip in tracker frame (Tt).
Step 8: Calculate the orthonormal coordinate systems for each position and use as Fm (marker frame). Use the orthonormal coordinate systems of Fm and Ft/Fd to calculate the rotation matrix and translation vector for the positions.
Step 9: Calculate the new center Pcali using the associated rotation matrix and translation vector for each position (A,B,C).
Step 10: Take the average of the Pcal values and use as the pivot point/drill tip in marker coordinates (Tm).

2. If the enforcement constraint of rotating around one pivot point failed, that would mean that the drill tip slipped out of the divot in the board and is no longer localized in one spot. If the center point slipped during the simulation/collection of points for calibration, then the clouds of points would have two different centers and therefore two different spheres would be generated. These two clouds could be detected using MATLAB classifiers.
3. To detect excessive random error in the tracking of the markers, I would compare the distances of the points to the generated spheres and remove the points that are too far away and recalculate the spheres.
4. MATLAB code drillTipCalib.m included in zip file.

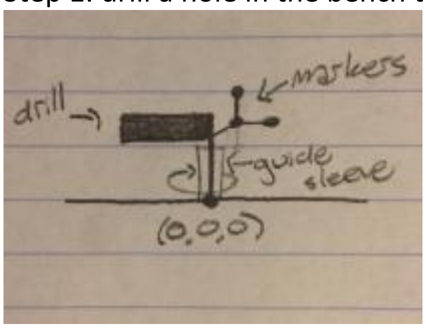
## Drill Axis Calibrator

Drill Axis Calibration Steps and Diagrams
** Note: hand drawn diagrams are not necessarily accurate depictions, use as a visual aid to understand method **
The method chosen for calibrating the drill axis was the 3 planes method. This method uses the 3 planes generated when rotating a level drill around in a circle. Each plane is associated with a point (1 plane for all A points, 1 plane for all B points, and 1 plane for all C points). The normals of these points are calculated and the average of the normal vectors is used as the drill vector in tracker coordinates as the planes should be parallel to the table (and therefore the drill vector should be perpendicular to the table) as the drill is being held level by a guide sleeve placed on the drill. The drill vector in tracker coordinates is then converted to marker coordinates using the rotation matrices associated with each marker pose. The average of these vectors is then used as the drill axis/vector in marker coordinates. This method is an accurate representation if the method is done right, but can be restrictive as the drill must be held level

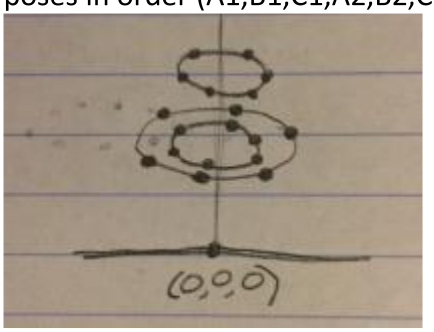
to the table at all times otherwise the generated planes will not be parallel to the table and the vector calibration will be offset.




Step 1: drill a hole in the bench to act as a divot deep enough to hold the drill in one place.




Step 2: Place a guide sleeve around the drill and on the table to hold the drill steady and level to rotate around.
Step 3: track the markers and rotate the drill around, keep it as level to the table as possible.
Step 4: Capture a set number of poses from this rotation and add them to a matrix of all the poses in order (A1,B1,C1,A2,B2,C2,…An,Bn,Cn).

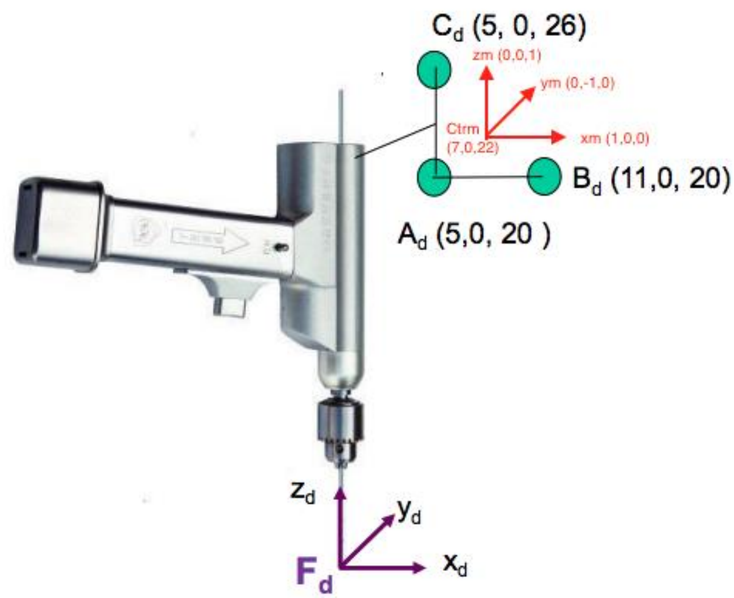



Step 5: Separate the data into sets of all A points, all B points, and all C points. Calculate the planes that fit these data points accordingly (i.e. a plane for all A points, a plane for all B points, and a plane for all C points).
Step 6: take the average of the normal vectors for the planes and use it as the vector in tracker coordinates (vt).
Step 7: Calculate the orthonormal coordinate systems for each position and use as Fm (marker frame). Use the orthonormal coordinate systems of Fm and Ft/Fd to calculate the rotation matrix for the positions.
Step 8: Calculate the vector in marker coordinates for each position by multiplying it by the rotation matrix associated with that position.
Step 9: take the average of these vectors in marker coordinates and use as the drill axis/vector in marker coordinates.

1. If the enforcement constraint of keeping the drill level fails, that would mean that the drill is now wobbling and not steady, therefore the planes being generated from the data points will be more generalized and may not be parallel to the plane. To detect this error, I would compare the normal of each plane to the ground truth z direction vector and determine how much of an angle difference there is from the plane normal to the z direction vector. Having a wobbling drill also means that not all points will be on the plane as the drill will be moving around, so I will detect any points that are not on the plane they are supposed to be on within an error margin (the approximate plane they are supposed to be on will be calculated before the collection of data points).
2. To detect if there is excessive random error in the tracking I would use the same method in question 2. I would calculate the distances of the points to the calculated plane and the ideal plane and remove any outliers that aren't within a margin of error. The plane would be calculated again and the process would continue until the error is within the proper margin.
3. MATLAB code drillVecCalib.m included in zip file.

# Part 3: Simulation

## Compute Marker Frame

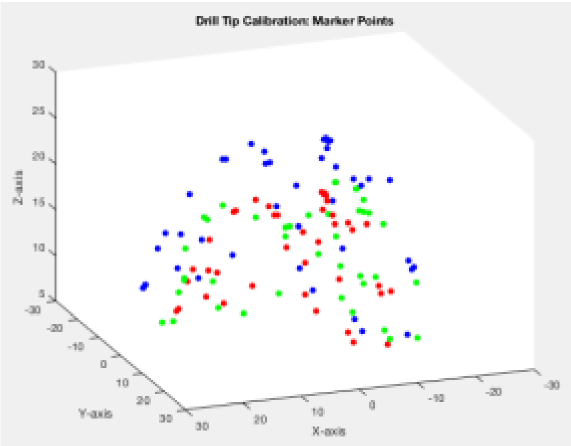


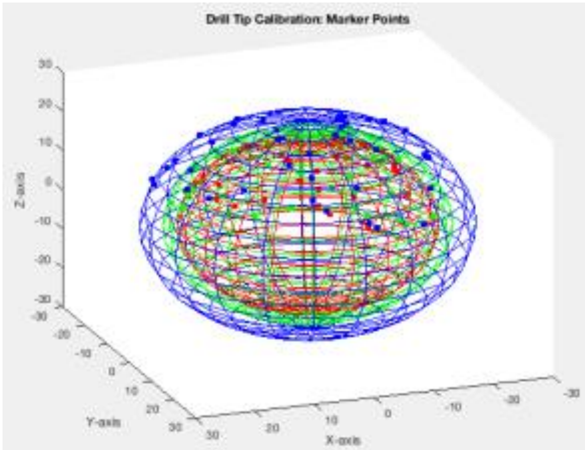

The figure above is an illustration of the computed marker frame. The marker frame was computed using the OrthonormalCoordinateSystem function (verified in a previous assignment) and the three points Ad Bd and Cd. The coordinate system is centered in the center of mass of the triangle and has 3 corresponding orthonormal vector bases (see labels in image above).

## Compute Ground Truth

The ground truths using the Fm marker frame, Fd drill frame, vt and Tt have been calculated to be Tm = (-7,0,-22) and vm = (0,0,1). The math can be seen on the attached pdf titled Ground Truth Computation.

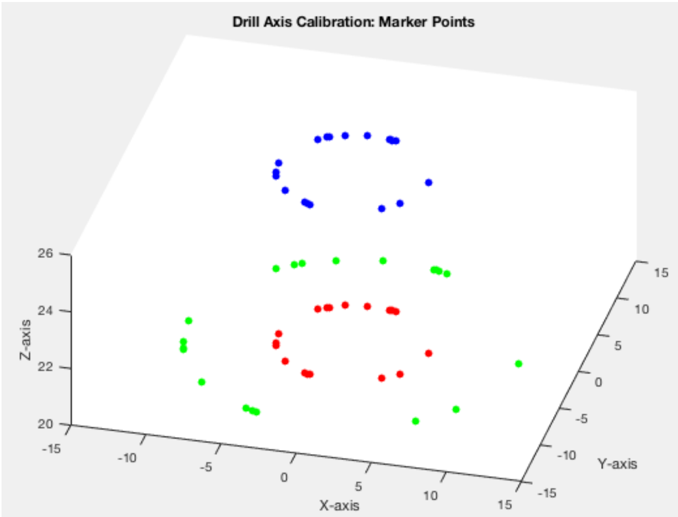
## Drill Tip Simulator




When running the drill tip testing file, the first test run is used to verify the accuracy of the drill tip simulator. The left graph consists of the 20 randomly generated positions within the angle range of 75 degrees with zero error. Although it can be difficult to see, this graph shows that the points seem to center around the point (0,0,0) in tracker coordinates. The graph on the right further verifies this simulation function as the generated spheres around the points are all centered at the point (0,0,0) in tracker coordinates and all the points are on the surface of their corresponding spheres with the correct corresponding radiuses illustrating no error in this simulator function (which makes sense as the inputted error was 0).

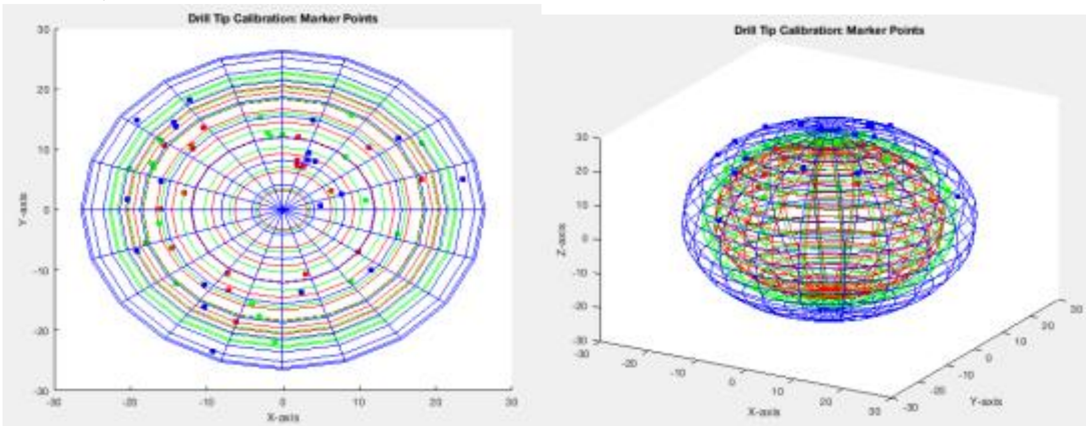
# Drill Axis Simulator




While running the drill axis testing file, the first test run is used to verify the accuracy of the drill axis (vector) simulator. The graph above illustrates the randomly generated points from rotating the drill and therefore markers around 360 degrees. Note the lack of restriction on rotation range is for visualization purposes to verify this function properly generates points that form circles with their centers found on the vector [0,0,1]. This graph clearly illustrates that as 3 distinct circles of points with proper sizes are visible and are all centered along the vector [0,0,1]. This graph also verifies the simulator because all of the points line up with one other properly in their poses meaning there is no error in these generated points.

# Part 4: Calibration Test

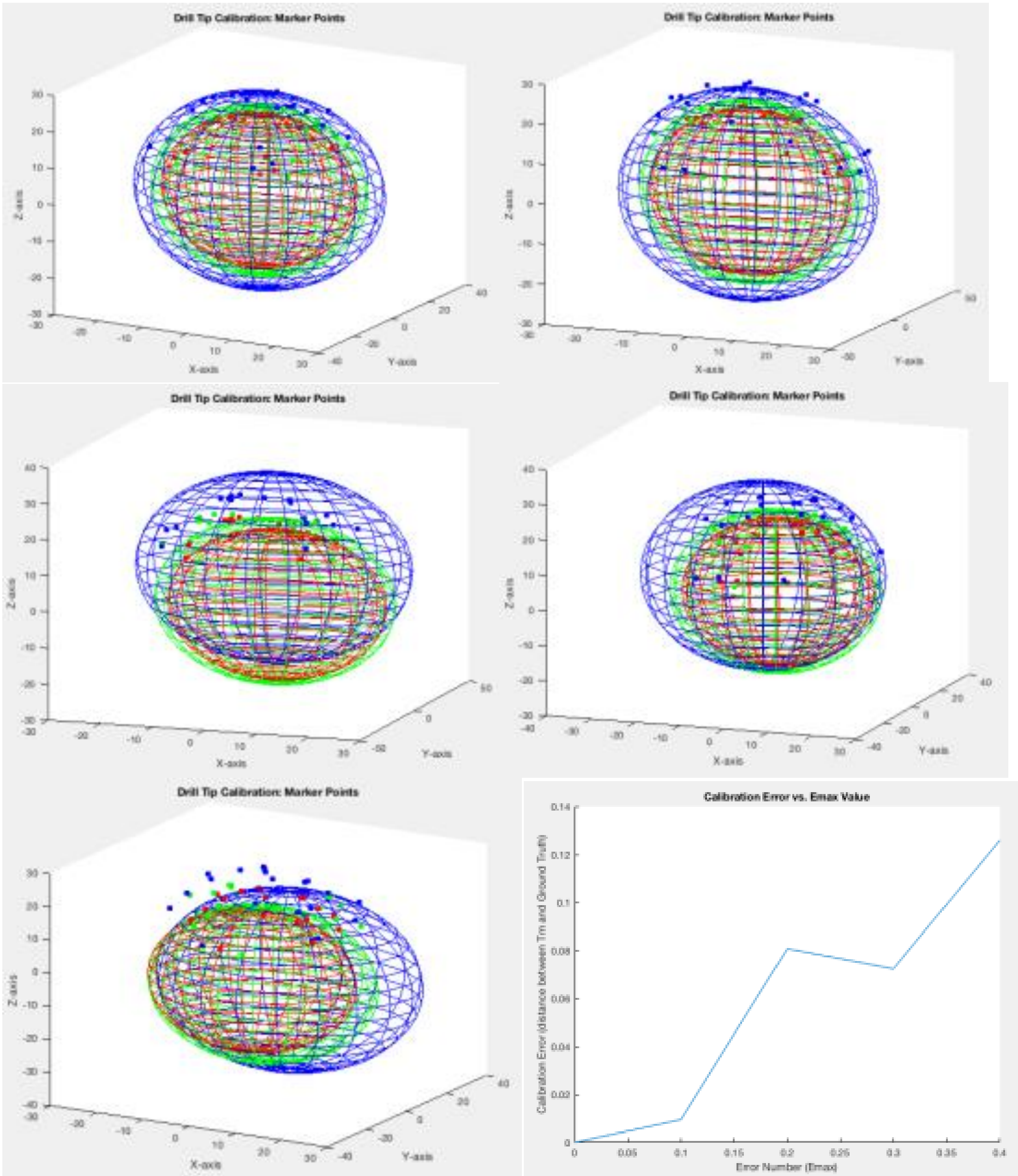
## Drill Tip Calibration Test




```
This calibration function works within the required accuracy.

Tm =

   -7.0000
   -0.0000
  -22.0000


gt =

    -7
     0
   -22
```

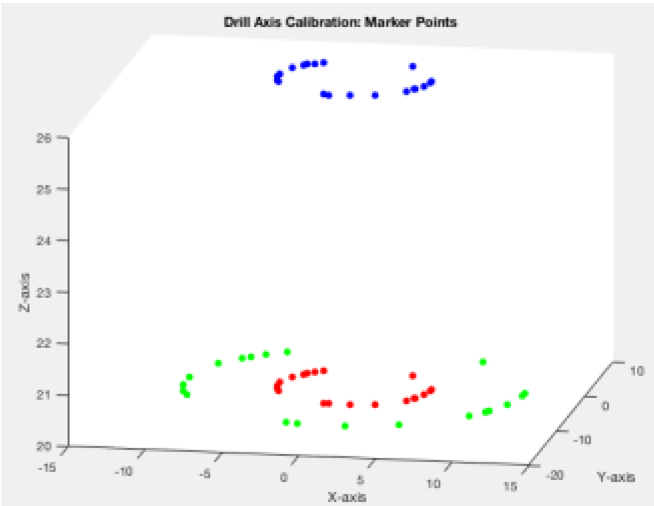
The number of poses chosen for this test is 20 poses, as that was the recommended amount for testing the simulation function and it provides a variety of poses for calibration without having a high chance of duplicate poses. The range chosen for this calibration test is 75 degrees. Since the drill is bulky, it wouldn't be possible to lay it completely flat on the ground at a 90-degree angle so having the range be 90 degrees wouldn't be possible in the lab (therefore anything more than 90 degrees wouldn't make sense either). 75 degrees provides enough of a range for the sphere fit function to generate an accurate sphere without putting too much strain on the handler of the drill to reach a difficult angle within a larger range. In the loop function built for the robustness test, the first test uses an error of 0, which is the verification test for the calibration function to see if it calculates the ground truth. As seen in the Command Window output above, when the test file is run this statement prints showing that the calibration function works within the required 1mm level of accuracy. The calculated Tm value and the ground truth value is then printed out below and it can be seen that the two points are the same, meaning that the difference between the two points is 0. This can be further verified by observing the spheres above from a birds eye view and a side perspective, as this illustrates the center of the spheres to be in (0,0,0) in tracker coordinates which when properly transformed into marker coordinates is (-7,0,-22).

# Drill Tip Calibration Robustness Test




In the graphs shown above, the first 5 graphs are the plots of the generated points and their corresponding spheres, going from 0 Emax to 5mm of Emax. It is clearly illustrated that as the error value increases, the points and three spheres become less and less concentric and have their own centers, making it more difficult to find the actual average. In this case, the maximum value of Emax when the calibration is no longer clinically satisfactory is 0.4 as the calibration error exceeds the threshold of 1mm and instead is approximately 0.12mm. This is visually illustrated on the 6$^{th}$ graph in the bottom right corner as it shows the increase in calibration error as the Emax value increases, eventually surpassing the threshold of 0.1cm (1mm) when the error is 0.4cm (4mm).

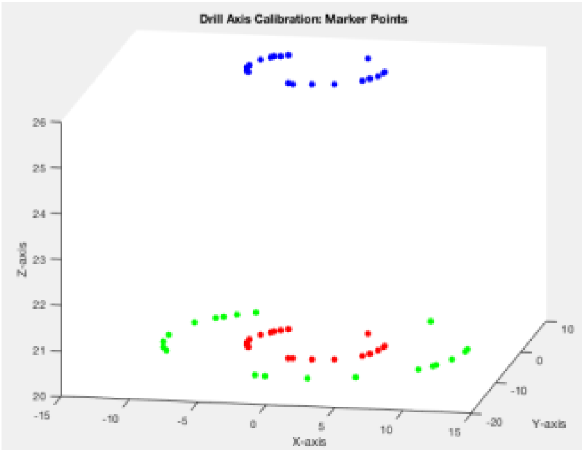
# Drill Axis Calibration Test

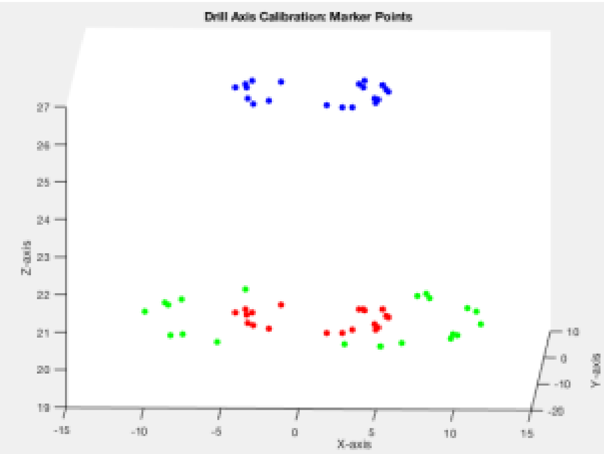


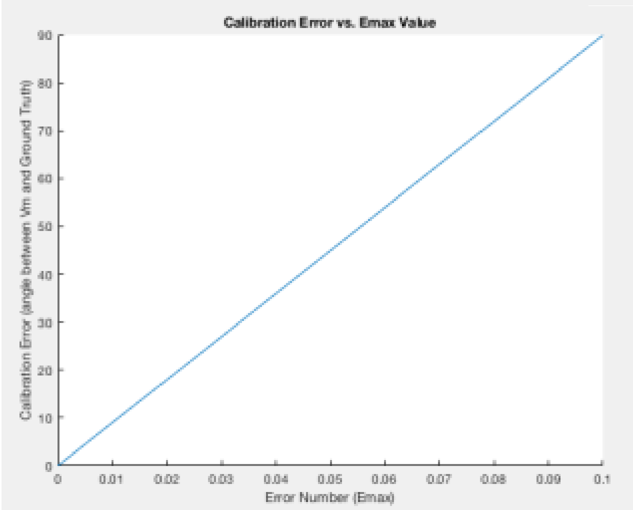

```
This calibration function works within the required accuracy.

Vm =

     0
     0
     1


gt =

     0
     0
     1
```

The number of poses chosen for this test is 20 poses, as that was the recommended amount for testing the simulation function and it provides a variety of poses for calibration without having a high chance of duplicate poses. The range chosen for this calibration test is 360 degrees. Since the drill is being held level by using a guide sleeve and the drill tip is held in place in the hole drilled in the table, the drill can be fully rotated around the drill axis (z axis) without wobbling. Having a 360-degree range allows for better data collection for the generation of the surface of the plane as well. If we used 180 degrees only half of the sphere would be collected and the generated plane may not be as accurate or as level compares to generating the whole plane. This would be even worse if we used 90 degrees as only a quarter of the circle would be generated. In the loop function built for the robustness test, the first test uses an error of 0, which is the verification test for the calibration function to see if it calculates the ground truth. As seen in the Command Window output above, when the test file is run this statement prints showing that the calibration function works within the required 2 degrees level of accuracy. The calculated Vm value and the ground truth value is then printed out below and it can be seen that the two points are the same vector (0,0,1), meaning that the difference between the two points is 0. This can be further verified by observing the plotted data points above as the 3 planes that would be generated on these points would be parallel to the table and therefore have the same normal vector as the ground truth at (0,0,1).

## Bonus: Drill Axis Robustness Test




In the graphs shown above, the first two graphs are the plots of the generated points with their corresponding errors (first graph is no error, second graph is 0.1 error). In the second graph, the points are not level with one another as they have been randomly dispersed by 0.1 mm along the z axis. Since these points are no longer on the same plane, it becomes more difficult to calculate the correct corresponding planes to capture all of the points. The third graph shows the correlation between the Emax value and the Calibration error result (the angle between the calculated Vm vector and the ground truth vector). When the Emax value is set to zero, there is not calibration error for the Vm and the ground truth as they are the same vector. When error is introduced though, the calibration error spikes upwards and is no longer within the threshold of 2 degrees of error. This robustness test shows that although this method is very accurate if the collected data points are accurate, the method is not very robust as even an introduction of 1mm of error can provide a completely wrong drill axis vector and the calibration will no longer be clinically satisfactory.