# A3

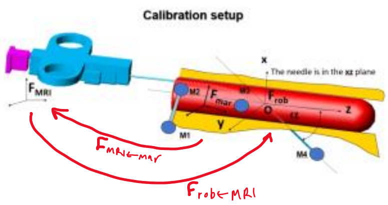
November 13, 2024 11:51 AM

## Q1: NAVIGATION TRANSFORMATION

Write up the math formula to transform a biopsy target point from MRI scanner frame (FMRI) to robot frame (Frob).

I have changed this equation from what is in the lecture notes because I assume MRI frame is home frame because we are tracking the markers so we have the marker frame and we have the robot coordinates from calibrating the robot (getting the axes and the center of the frame). This way we can translate marker to robot by having the MRI as the home position in the transforms.

$$P_{rob} = F_{rob \leftarrow MRI} \, F_{MRI \leftarrow mar} \, P_{mar}$$




For reference:
(extra)

$$T_{hev} = \begin{bmatrix} 1 & 0 & 0 & -O_{vx} \\ 0 & 1 & 0 & -O_{vy} \\ 0 & 0 & 1 & -O_{vz} \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1}$$

$$R_{hev} = \begin{bmatrix} V_{1x} & V_{2x} & V_{3x} & 0 \\ V_{1y} & V_{2y} & V_{3y} & 0 \\ V_{1z} & V_{2z} & V_{3z} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$F_{hev} = T_{hev} R_{hev}$$

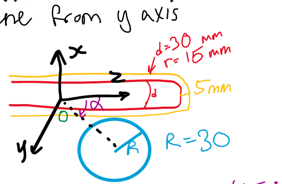
$$R \, T \, P_h = P_v$$

## Q2: WORKSPACE

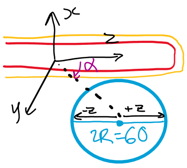
Find 3 DOF: Translation, insertion, rotation

**Translation:**

Scene from y axis




O = home = (0,0,0)  $\alpha = 45°$




minimum required range of motion is:

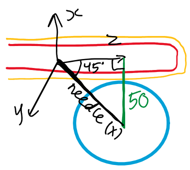
*assume at home (0,0,0) needle is at the center of the prostate

$z_{min} = -30_{mm}$

$z_{max} = 30_{mm}$

**Translation:** $\Delta z = z_{max} - z_{min} = 60_{mm}$

**Insertion:**




$$\frac{50}{\sin(45)} = \frac{X}{\sin(90)}$$

$$50 \sin(90) = X \sin(45)$$

$$\frac{50}{\sin(45)} = X$$

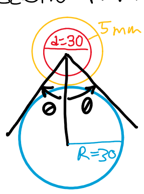
$70.71 = X$ length of needle

range $= [X-30, X+30]$

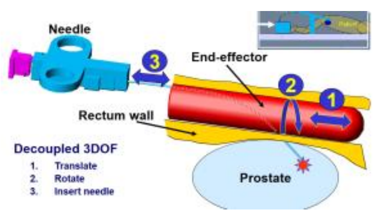
**Insertion:** $[40.71, 100.71]$

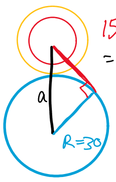
**Rotation:**

Scene from Z axis

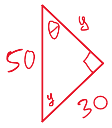



Tangent to the prostate at both sides. Looking into the rectum through the z coordinate axis. This is when the robot rotates seen as "2" in this image →






$15+5+30$
$=50$



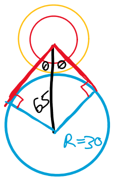

y is tangent to the circle

$$\frac{50}{\sin(90)} = \frac{30}{\sin(\theta)}$$

$$50 \cdot \sin(\theta) = 30 \sin(90)$$

$$\sin(\theta) = \frac{30}{50}$$

$$\theta = 36.87^\circ$$




$\theta = 36.87^\circ$ $2\theta = 73.74^\circ$

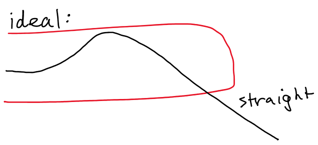
Rotation of the needle from edge to edge of the prostate is 73.74 degrees.
From the center assuming the rotation is 0 degrees at home position it would rotate 36.87 degrees in one direction and -36.87 degrees in the other direction.
But the full rotation is 73.74 degrees.

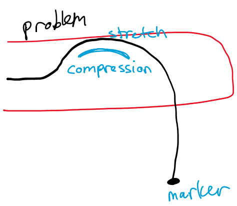
Q3: ROBOT CALIBRATION

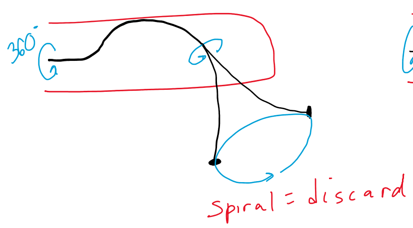
**Q3.1 Needle Bending:**

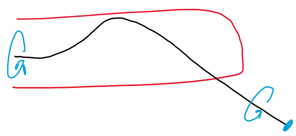
Bending the needle causes a stretch and compression along each side of the needle which ultimately causes the end of the needle to bend in. This is not good because then the target cannot be hit easily. We can test each needle before sterilizing it for the OR by rotating it. If rotated and the marker moves in a circle then we can conclude that it is bent and if the marker doesn't move and just rotates in its spot then the needle is straight.

We can do this in water which is thicker than air but less than the body. We can bend the needle to 45 degrees (like specified in the assignment) and rotate it 360 degrees. If we see a rotation and a circle in the needle or in the M4 marker it is a bad needle and can be discarded.
You could calculate the circumference of the circle. The bigger it is the more bent the needle is. If the circumference is 0 then the needle is straight.



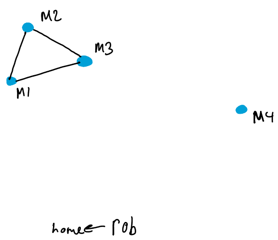



marker doesn't move
= straight
= keep.

Quantify by how much the needle would miss the biopsy target at maximum insertion depth:
If the needle is bent then it would miss the target by the same measurement as the radius of the circle in the experiment I talk about. So rotate it 360 degrees and find the circumference and radius. The radius is the measurement it would miss the target by.

**Q3.2 Calibration Design:**

Test and sketch




Start with determining the z axis:
Move the robot along the rectum to track the translation axis (z axis). Collect points along the rectum and normalize the line found to find z. If you rotate around z you can create a circle around z with the markers. If you take one of the furthest markers (M1 and M4) and find the center it is a point on the z axis. Now we have the point and the direction for the z axis.

Next we know that the needle insertion is on the xz plane so y is the orthogonal axis so we move the needle and collect points from its movement and we can find y from the orthogonal line from point of z and its direction (or the line of the z axis).

Then we can find x from the cross product of z and y.

Now O is the intersection of the 3 axes. Or we can find this by doing the symbolic intersection of the lines in 3D. You can find the v matrix with -v1 =needle, v2=z axis, v3=normalize cross product of the needle and the z axis. And the P matrix would be points along the needle and then we can easily find t and the L equations to find O.

To find alpha the dot product of the needle direction and the z axis is taken because the dot product measures how much one vector points in the same direction as the other. The arc cosine function is used to compute the angle in radians. And it is converted to degrees.

**Q3.3 Calibration Implementation:**
I start with moving the robot along the rectum (translation) in 1 step, which in this case is 5mm. With this I found the z axis direction vector. Then I simulate the rotation by rotating m1 and m4 because they are the furthest from the z axis and generate points along their rotational path to find the center points.

Then we want to find the direction of needle insertion so we take one step in the direction from home to M4. We assume home is (0,0,0) in the resting robot. We find this needle direction which is along the xz plane, so y should be 0. We also know the needle is pointed down so the x value of the needle direction should be negative.

Since needle moves in XZ plane, y-axis is orthogonal to both z-axis and needle direction. We normalize this to get the y axis.

Then we calculate the x axis by doing the cross product of the y and z axis since x is orthogonal to them. Again normalizing it.

To find alpha the dot product of the needle direction and the z axis is taken because the dot product measures how much one vector points in the same direction as the other. The arc cosine function is used to compute the angle in radians. And it is converted to degrees.

For O, we can do the symbolic intersection of the lines in 3D. You can find the v matrix with -v1=needle, v2=z axis, v3=normalize cross product of the needle and the z axis. And the P matrix would be points along the needle and then we can easily find t and the L equations to find O.

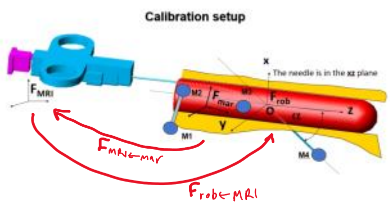
**Q3.4 Calibration Software Test:**
We can translate the robot across the workspace in 5 mm steps, rotate the robot in a 360 deg range in 30 deg increments, and insert/retract the needle across the workspace in 5 mm increments. Run the calibration, and prove that you can reproduce the ground truth Frob(x, y, z, O, alpha) calibration parameters.

For this my code is the same as 3.3 but instead of using 1 step for the DOF, we use a for loop for each DOF and translate in 5mm steps, rotate in 30 deg increments and insert needle in 5mm increments.

When I run this it outputs the same thing as 3.3 proving I can reproduce the ground truth Frob(x, y, z, O, alpha) calibration parameters.

**Q3.5 Marker-to-Robot Frame Transformation:**
I can prove my equation of Q1 by running 3.5.
I can get the x, y, and z axis and frame center information from running Q3.4. I pass this information into Q3.5.
First I generate an orthogonal frame from the M1,M2,M3 points on the robot. Transform this frame to home with the function from A1. Next I take the robot frame axes and center information passed in from running Q3.4 and transform this to home too. But I take the inverse of this since we are looking for home to robot. Lastly, we multiply the frame transform from marker to home by home to robot to get robot to marker which is the output.

$$P_{rob} = F_{rob \leftarrow MRI} \, F_{MRI \leftarrow mar} \, P_{mar}$$




Q 4: KINEMATICS

Q4.1 Forward kinematics:

1. Translate
2. Rotate
3. Insert needle

Forward kinematics involves computing the location of the needle tip based on the robot's motion stages, which include translation, rotation, and insertion. The task requires calculating the resulting position of the needle tip after applying a translation, a rotation, and an insertion. The forward kinematics is in this order above which means we multiply T*R*needle vector. The method involves applying these motions one by one to the robot starting from its home position, updating the position and orientation of the needle accordingly.
We use the translation vector Ov=[Ovx,Ovy,Ovz] similar to assignment 1 but it is only a translation in the z direction so we keep x and y = 0. The homogeneous rotation matrix used is the z axis one that is in the notes. We know alpha is 45 so we move a ground truth point (P0) in the needle vector direction.

$$T_{h \leftarrow v} = \begin{bmatrix} 1 & 0 & 0 & -O_{vx} \\ 0 & 1 & 0 & -O_{vy} \\ 0 & 0 & 1 & -O_{vz} \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \qquad R_{h \leftarrow v} = \begin{bmatrix} V_{1x} & V_{2x} & V_{3x} & 0 \\ V_{1y} & V_{2y} & V_{3y} & 0 \\ V_{1z} & V_{2z} & V_{3z} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

rotate robot around z-axis:

$$R_z(\alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

we know:

$\rightarrow F_{h \leftarrow v} = T_{h \leftarrow v} R_{h \leftarrow v}$

$\rightarrow R \, T \, P_h = P_v$

So in my code:

$\rightarrow F = (T*R)*N$

Q4.1 Inverse kinematics:

1. (u,v,w) in mri scanner frame

The inverse involves determining the required translation, rotation, and insertion values that will bring the needle to a desired target position in space. We do the inverse by inverting the forward kinematics because it is a first in last out type of motion. The angle is determined by the arc tangent of x/z coordinates. The translation is the z-coordinate before rotation. We need to project the target point back onto the z-axis

Q4.3 Kinematic Test:
Generate 10 random target points within workspace. Generate random target within workspace. Compute inverse kinematics. Compute forward kinematics with these outputs. Calculate error.
The error should be 0 because the points should be the same. My error is 0 for each of the 10 random target points.

Q 5: MARKER TRACKING

We know:
M1z=M2z
M2y=M3y
M2x=M3x
M3z=M2z+80
||M2 - M1|| = 60
||M3 - M2|| = 80
||M3 - M1|| = 100

My function q5_marker_tracking reconstructs the 3D positions of four markers (M1, M2, M3, M4) based on their gradient locations in MRI signal data.
Initially, it defines the ground truth positions for M1, M2, M3, and M4. just to find that M3 is 80mm down from M2.

Then I use the z-coordinates from the input gradient data to assign matching z-values to M1, M2, and M3, ensuring that M1 and M2 have the same z-coordinate, while M3 has a z-coordinate that is a fixed distance from M2. M4's z-coordinate is determined as the last remaining value.

For the x and y coordinates, the function identifies matching x and y values for M2 and M3, and calculates the Euclidean distances to find the most accurate positions for M1 and M4 based on their known distances from M2 and M3. Finally, the positions of all markers are assigned, with M1 and M4 adjusted to minimize the error in their distances from M2 and M3.