Introduction to Robotics

Lab Report

Areeb Adnan Khan
Nimra Sohail

Task 2.1 Model of the arm (20 points)

We know that a robot manipulator is mathematically modeled as a kinematic chain, made up of joints and links. Identify all the joints and links in this arm.

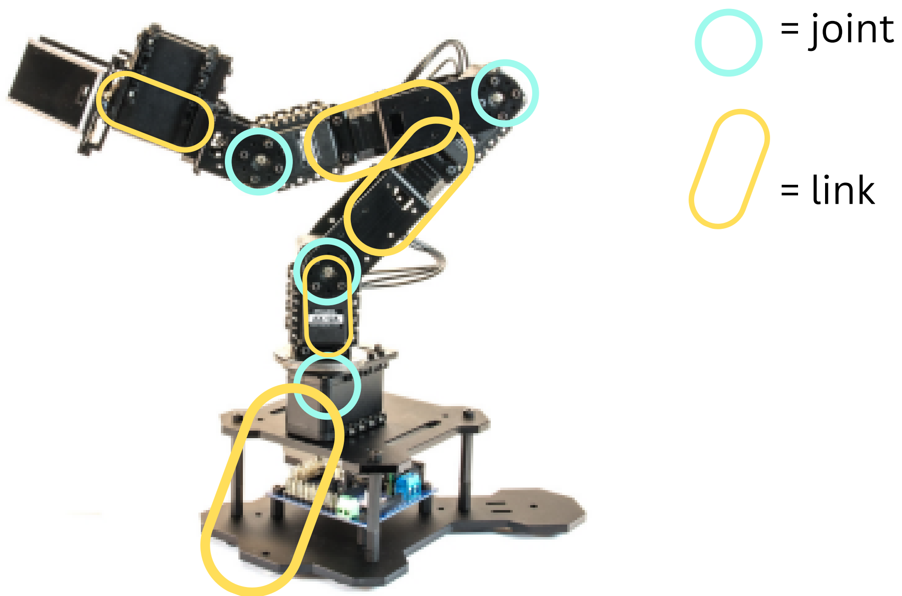
(a) Mark all the joints and links in Figure 2.1 or any other image of the arm.

(b) How many joints and links are in this arm? Note that the motor attached to the grasper is only responsible for opening and closing the grasper.

(c) What is the joint type? Provide a symbolic representation of the kinematic chain corresponding to this arm. Recall that a kinematic chain is symbolically represented as a sequence of joint symbols.

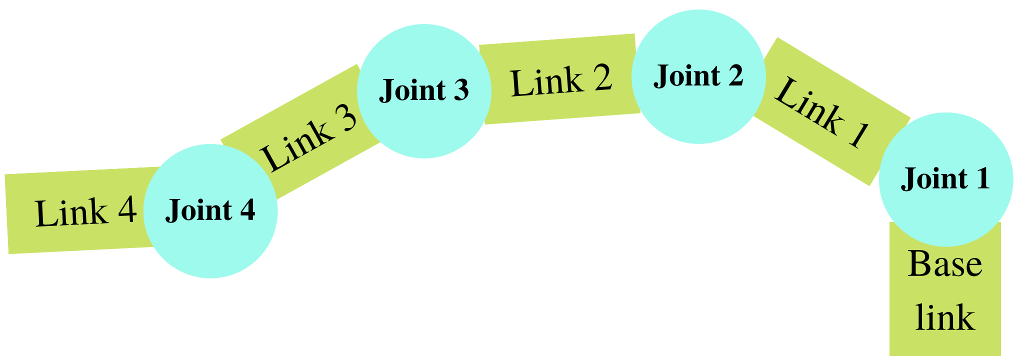
(d) How many degrees of freedom does this arm possess? *Hint: You can use Grubler's formula from the class slides.*

(a)The following is an illustration of joints and links in the robot:



(b)There are 4 joints and 5 links in total.

(c)



(d) 4 degrees of freedom

Task 2.2 Configurations Exploration (20 points)

Play around with the different modes of motion in the software and explore the capabilities and limitations of this arm.

(a) Move the arm to a configuration in which it reaches the farthest possible point. Draw this configuration as a diagram. In this diagram, links can be represented by line segments and revolute joints by circles.

(b) In Cartesian mode, move the robot to an arbitrary (x, y, z) location. Change the wrist angle from the panel and observe what happens to the other joints of the arm. Document your observations and comment on the reasons behind what you observe.

(c) Grab one each of the provided objects. In this task, you'll place each of these objects at a fixed location in your workspace, move the arm using `ArmLink` to that location, pick the object, and place it at another location. During this activity, how is the real world environment being sensed and how is the arm motion being adjusted based on the received sensing data? Where is this processing happening?

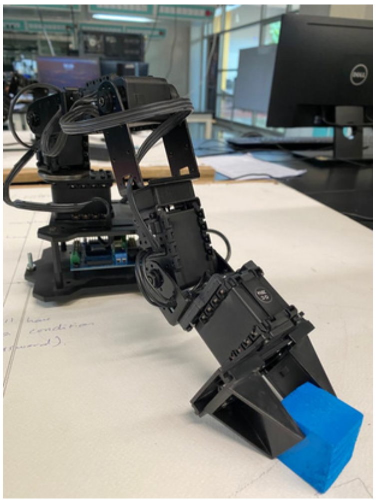
(d) [*][a] The coordinates in the Cartesian or Cylindrical mode describe task space locations. Task space can be used to describe tasks to be carried out by the manipulator, e.g. grabbing a water bottle. Give an example of a task that can be described better in Cartesian coordinates, and a task, which is best expressed in cylindrical coordinates.

(a)

(b)

When we change the wrist angle then what we initially thought was only the gripper motor will change since then it would be better able to grip the position, but we observe that all of the motors along with the wrist motor were moving. This can be explained in the sense that when we enter the angle of the motor than we changed our corresponding (x,y,z) co-ordinates so to maintain the position and the angle all the motors adjusted themselves accordingly

(c)



Analysis:

In this robotic arm, there are no sensors that can feed the real time data automatically and adjust the position and angle, infact the sensory part of this robotic arm is being carried out manually by the human eye. The brain in our case is the controller and when we have to pick an object, we predict the co-ordinates and the position and input them in the ArmLink software. The software and the code behind the software makes the dc motors work and turn towards the location where we have to pick up an object, but it is rarely the case that our object is picked up by the arm in the first try since our brain can't precisely predict the co-ordinates so we have to use the trial and error method. We carried out the pick and drop task by this method as shown from the pictures

(d)

The placement of points in space can be easily described using the Cartesian coordinate system. However, some surfaces can be challenging to represent using equations based on the Cartesian system. As the name implies, calculating the capacity of a cylinder-shaped water tank or the flow rate of oil through a pipe can be done with the use of cylindrical coordinates.

Let's suppose we have to Determine the velocity of a submarine subjected to an ocean current. Typically, a submarine travels in a straight route. Rectangular coordinates are an excellent option because there isn't any rotational or spherical symmetry that applies in this scenario. Probably, the z-axis should be vertical. It is possible to align the x- and y-axes to point east and north, respectively.

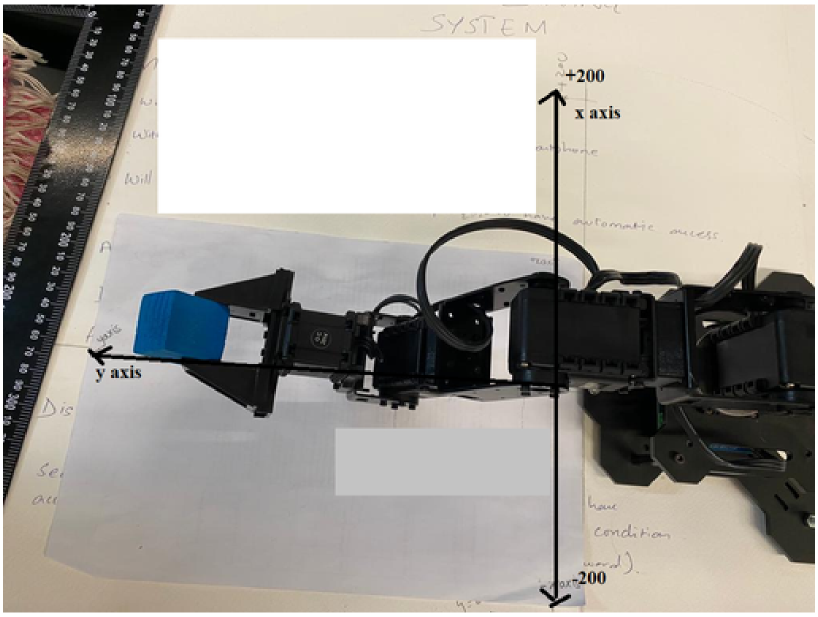
And let's suppose we have the task to determine the amount of leather required to make a soccer ball Since the rotation of a soccer ball is symmetric on a central axis, cylindrical coordinates are ideal. The ball's axis and the z-axis should line up. The ball's center may serve as the genesis. The x-axis' location is arbitrary.

Task 2.3 Coordinate Axes (10 points)

(a) Determine the directions of positive x, y, and z axes and mark them on paper, in relation to the shape of the black base.

(b) We'll set the origin of the x and y axes at the center of the shaft of the first motor, and the origin of the z axis at the level of the wood platform. If 1 unit in the ArmLink system corresponds to 1 unit in the real world, identify the units being utilized in the real world and the point of the arm whose position is being determined.

(a)



(b) By using the mm rule given to us the following calculations were done and it was found out that 1 unit in software is approximately 1 mm in real world.

240(max distance in y)-50(min distance in y)=190units

distance in mm rule =189mm

190units are 189mm in real world

1 unit is 189/190 =0.9947 mm in real world

Task 2.4	Accuracy and Repeatability (50 points)

Identify a method for physically measuring the position of the wrist reference point of a robot arm.

(a) Design an experiment for determining the accuracy and repeatability of the robot arm and provide details of this experiment.

For this experiment we have taken 5 points on the workspace of the robot
Our number of experiments here is 3 to check for the errors in the robotic structure
We have used here a marker to determine the point of contact with the white sheet of paper.
The command pose co-ordinates are as follows:
P1(200,199,210) P2(107,185,210) P3(-9,185,180) P4(-79,227,164) P5(-111,170,164)

We did 3 Trails:
n = 1
P1(189,188,205) P2(100,188,205) P3(-4,186,186) P4(-82,220,163) P5(-112,171,163)
n = 2
P1(188,187,204) P2(104,183,208) P3(-7,171,185) P4(-80,228,178) P5(-110,173,165)
n = 3
P1(190,189,206) P2(103,186,206) P3(-10,190,182) P4(-77,230,164) P5(-113,169,162)

We wrote the MATLAB script as follows and ran it for each point for the three trails and got t the positioning accuracy and repeatability for each point as follows:

```
xc=200;
yc=199;
zc=210;
n=3;

xj=[195 196 202];
yj=[197 203 205];
zj=[210 211 213];
xbar=sum(xj)/n;
ybar=sum(yj)/n;
zbar=sum(zj)/n;
AP_p=sqrt( (xbar-xc).^2 + (ybar-yc).^2 + (zbar-zc).^2)


lj=[];
for x=1:n
   lj(x)= sqrt( (xj(x)-xbar).^2 +(yj(x)-ybar).^2 + (zj(x)-zbar).^2 );
end
lbar=sum(lj)/n;
inner=[];
for x=1:n
  inner(x)=(lj(x)-lbar).^2;
```

For point 1:
xc=200;
yc=199;
zc=210;
n=3;
 	
xj=[189 188 190];
yj=[188 187 189];
zj=[205 204 206];
 	
Output:
AP_p =
 	
16.3401
 	
RP_l =
 	
4.1547
	For point 2:
	xc=107;
	yc=185;
	zc=210;
	n=3;
	
	xj=[100 104 103];
	yj=[188 183 186];
	zj=[205 208 206];
	
	Output:
	AP_p =
	
	5.9722
	
	RP_l =
	
	7.3951
		For point 3:
		xc=-9;
		yc=185;
		zc=180;
		n=3;
		
		xj=[-4 -7 -10];
		yj=[186 171 190];
		zj=[186 185 182];
		xbar=sum(xj)/n;
		ybar=sum(yj)/n;
		zbar=sum(zj)/n;
		
		output:
		AP_p =
		
		5.4671
		
		RP_l =
		
		17.8284
			For point 4:
			xc=-79;
			yc=227;
			zc=164;
			n=3;
			
			xj=[-82 -80 -77];
			yj=[220 228 230];
			zj=[163 178 164];
			
			Output:
			AP_p =
			
			4.4969
			
			RP_l =
			
			13.3538
				For point 5:
				xc=-111;
				yc=170;
				zc=164;
				n=3;
				
				xj=[-112 -110 -113];
				yj=[171 173 169];
				zj=[163 165 162];
				
				Output:
				AP_p =
				
				1.3744
				
				RP_l =
				
				6.3758

(c)
The accuracy indeed varies with the distance from the base, as we can see from the point P1 which is the farthest point from the base it's accuracy is worse than all the points and then we have the point P2 which is arguably the second most farthest point from the base. It has better accuracy than point P1 but it's worse than the other 3 points. We have determined this conclusion in such a way that we designed our experiment that point P1 was farthest from the base and P1 was following closely behind and we can see the general trend that follows.

(d) Industrial arm positioning accuracy and repeatability refer to the ability of the arm to accurately and consistently reach a desired position. These values can vary depending on the specific industrial arm being used, but generally, the accuracy and repeatability of industrial arms is quite high. For example, the accuracy of some industrial arms can be on the order of microns, while repeatability can be within a few microns. These values are significantly higher (more better than previous ones) than the ones obtained in the previous part, as industrial arms are designed for precise and repeatable movement in industrial settings. Just like Phantom X Pincher Arm has a repeatability of 0.1 mm and an accuracy of 0.5 mm. These values are considered to be very high for a robotic arm. The values that we got were not close to the original one because we used a more traditional method that gave way to lot of human errors.

(e) Low accuracy of the robot arm means that it may not be able to accurately reach the desired position to pick up or place an object, which could result in errors in the pipeline. For example, if the robot arm is picking up an object and the accuracy is low, it might not be able to pick up the object securely or it might pick up an incorrect object.

Low repeatability of the robot arm means that it may not be able to consistently reach the same position, which can result in errors in the pipeline. For example, if the robot arm is placing objects in a specific location and the repeatability is low, it might place the object in slightly different locations each time, resulting in a lower overall accuracy and efficiency of the pipeline.

Overall, low accuracy and repeatability of a robot arm can lead to a higher rate of errors in the pick and place pipeline and reduce the efficiency of the process.