ENPM662: Introduction to Robot Modeling

Final Project Report

Gantry Robot

Submitted by:

Jeffin Johny K (118293929)

Pradipkumar Kathiriya (117678345)

Date: 12/10/2021

UNIVERSITY OF MARYLAND 1856

A. JAMES CLARK
SCHOOL OF ENGINEERING

CONTENTS

CHAPTER 1

INTRODUCTION

Robot Manipulator is one of the widely used machine in the industry. Manipulator is consists of the number of links which are connected by various joints such as revolute, prismatic and spherical etc.. Hence, it has high degree of freedom (typically 5-7 in the robot used in the industry). Due to high degree of freedom, it has a very dexterous workspace and can be use to reach at any position and at any orientation within the reachable workspace. This makes it suitable for varieties of applications in the industry. It is used for welding, painting, tool changing in the CNC machine and machining etc.. It is also used for pick and place operation, however, in current situation it used for pick and place only where pick and place location are within the workspace. Typically, these robots are fixed at single place and hence it has a small reachable workspace (in a range of few meters). In this project, we have fixed the manipulator on the movable base and used this combine unit called gantry robot for pick and place operation where distance between pick and place location is large. The gantry robot can be used in place of the gantry crane for pick and place operation. Contrary to the gantry crane, gantry robot can approach the picking object from any orientation. Hence, with gantry robot it is possible to store the material in a stack which is not the case with gantry crane. In this project, we have modelled gantry crane and validate its forward and inverse kinematics. We have also moved the gantry robot in the gazebo to mimic pick and place operation.

The remaining structure of the report is as explained. The second chapter ‘Motivation’, briefly describes our inspiration to choose this project and also presents the intended application of the robot. ‘Robot type and Description’, gives an overview of the proposed model, along with the DOFs and dimensions of the robot. The next chapter ‘CAD model’, illustrates the isometric view of the designed model. Chapters (5),(6) and (7) deals with the determination of forward and inverse kinematics and its associated validation. After that the ‘Workspace study’ and ‘Assumptions’ are provided. Chapters (10) and (11), provides the explanation of controller selection and gazebo visualisation, respectively. Chapter (12), explains the challenges we faced during this project. Next the learning outcomes are provided and after that the next chapter is the conclusions drawn from this project. Finally, future scope of work is explained, followed by the references.

CHAPTER 2

MOTIVATION

Material storage and material handling (Movement from one place to other or loading material on the machine) is one of the primary concerns for most industries especially in the manufacturing sector. It has a significant impact on productivity and cost of production. Overhead crane (Gantry crane) is one of the widely used machines in the industry for material movement. With the modern available planning tools, most of the industries have optimized their shop floor for maximum storage of material and there is little scope to further optimise the shopfloor.

One way to further improve the storage capacity is to store the materials in a stack. This can be accomplished by placing the material directly one above the other or designing appropriate storage racks. However, it would not be possible to use the gantry crane for material handling if the material is stored in stack, as it is impossible to handle the material which is on the bottom of the rack by gantry crane. To tackle this problem, most industries are using ground material handling equipment like forklift and transfer trucks in conjunction with the gantry crane. The problem with using ground material handling equipment is that it requires substantial space on the ground to move around, which in turn reduces the storage capacity.

One of the potential solutions for the above-mentioned problem is to use a “Gantry robot”.

Gantry robot is a feasible solution to this problem especially where the material to be handled is of few hundred kilograms in weight. Robot manipulator can be attached to the gantry girder and this assembly can be used in place of a gantry crane. Gantry robots have many advantages. Due to its dexterity, it can handle the material which is at the bottom of the stack or rack and thereby eliminating the needs of ground material handling equipment. For repeated pick and place operation, Gantry robot can be automated. Automated robots will perform fast and precise pick and place operations. This will result in lower production cost and higher productivity. Moreover, automated material handling is safer than manual handling.

Thus, with this project we intend to propose and design a Gantry robot that can perform simple pick and place operations in an industrial shop floor.

CHAPTER 3

ROBOT TYPE AND DESCRIPTION

The proposed Gantry Robot has 8 DOFs and it can be divided into two parts:
1) Girder mechanism
2) 6 DOF manipulator attached to the girder (underneath the crab)

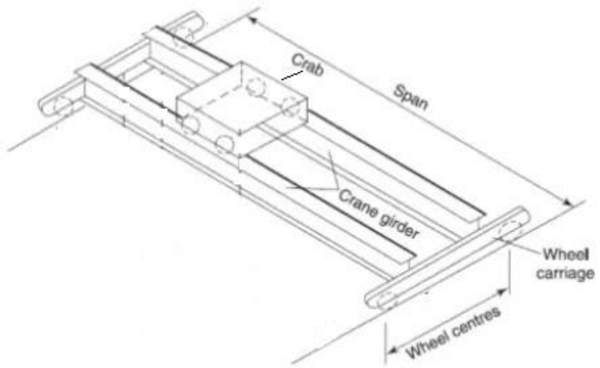


Fig.1: Schematic representation of girder

Girder mechanism, Crab and Frame:

The gantry comprises of 4 structural vertical beams at the corners of the workspace, which are connected by 2 parallel runway beams at the top. Now these 2 frame structures are connected with each other by another set of regular parallel beams, making it a single frame structure. The girder mechanism is perpendicular to the runway beams and hosts the crab. Girder mechanism and Crab, possesses 1 DOF each in horizontal plane.

Girder mechanism comprises of a girder placed on rails which holds the crab. This will allow the girder to cover the span from pillar to pillar in a horizontal plane. Crab (Link on which 6 DOF manipulator is attached to) will be installed on the girder and it is slotted to move along the girder. This will allow the crab to cover distance from pillar to pillar in the direction perpendicular to girder movement and in the same horizontal plane as girder.

Manipulator:

Finally, the 6 DOF manipulator will be installed on the crab. Girder mechanism will allow the 6 DOF manipulator to reach at any point in the horizontal plane. The wrist of this manipulator has 3 DOFs. Thus, the manipulator is dexterous and it can reach at any height on the shopfloor and from any orientation. So, this gantry robot can reach to any point on the shop floor, at any height and from any orientation. Gripper selection will depend on the type, size and nature of the material to be handled.

The specifications of the robot arm are chosen such that, a minimum clearance is present between the end effector and ground, when the arm is vertically downward.

Dimensions of the designed, scaled down model(7:1)

ROBOT MANIPULATOR	
DOFs	6
Mass	5.7kg
Range of joint angles	
$\boldsymbol{\theta}_1$	$-180° \; to + 180°$
$\boldsymbol{\theta}_2$	$-90° \; to + 90°$
$\boldsymbol{\theta}_3$	$-90° \; to + 90°$
$\boldsymbol{\theta}_4$	$-180° \; to + 180°$
$\boldsymbol{\theta}_5$	$-90° \; to + 90°$
$\boldsymbol{\theta}_6$	$-180° \; to + 180°$
Total length	450mm
Material	Carbon Steel
GANTRY (FRAME, CRAB, GIRDER)	
DOFs	2
Height of frame	642.86mm
Runway beam length	2142.86mm
Girder length	1430mm
Mass	406kg
Material	Carbon Steel

CHAPTER 4

CAD MODEL

A SolidWorks model was created from ground up, with the scaled down dimensions(7:1). All the dimensions were considered in millimetre. The base frame was designed such that its height was along the positive z axis, this was done so that the assembly will also be along positive z direction for both SolidWorks and gazebo. After modelling in Solidworks, the model was converted into urdf file using the urdf exporter.

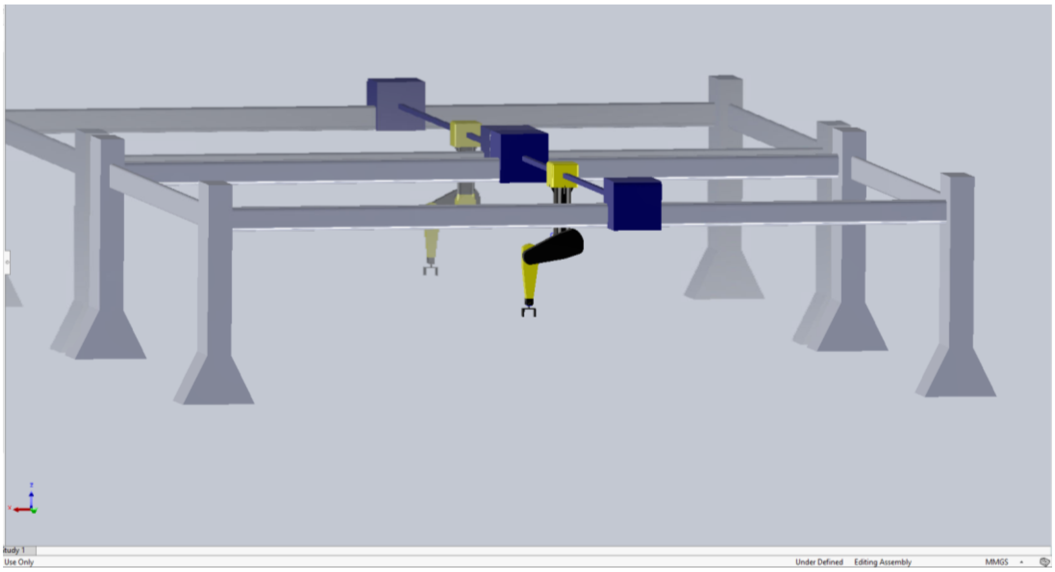


Fig.2: The designed model

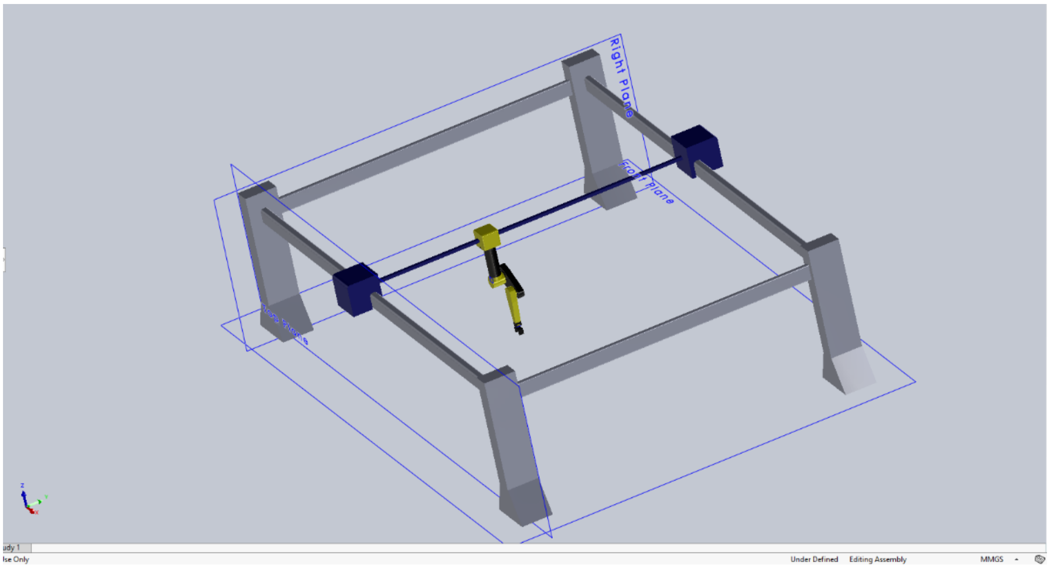


Fig.3: Isometric view of the setup

CHAPTER 5

KINEMATICS

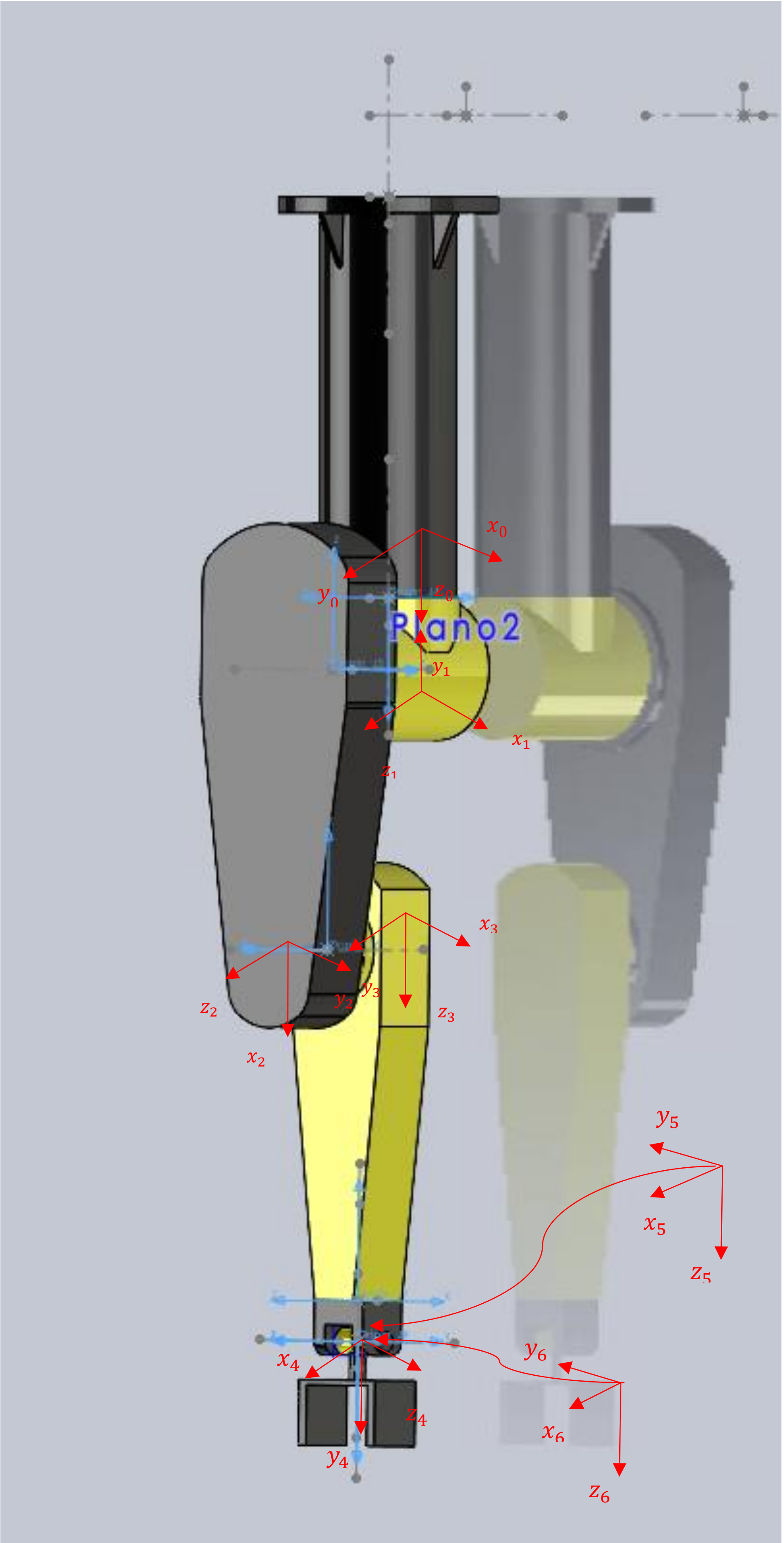
Kinematic analysis of the gantry robot was carried out. For both forward and inverse kinematics analysis, only the 6 DOFs manipulator arm was considered. This made the calculations effortless for inverse kinematics as the Jacobian became a 6X6 matrix.

5.1) DH Parameters:

The DH parameters, were determined for the 6DOF robot manipulator. The orientation of the arm while calculating parameters, is as shown. The base link of the manipulator(black color cylinder) is fixed to the crab, hence do not any joint variable. Spong convention was adopted for solving. The first rotating joint is assigned frame {0} and end effector frame {6}. All the horizontal and vertical distances between the frames were obtained from SolidWorks, after the urdf frame assignment.

The pose of robot for DH parametrisation along with the frame assignment is as shown in the figure.

Fig.4 : Initial pose of arm



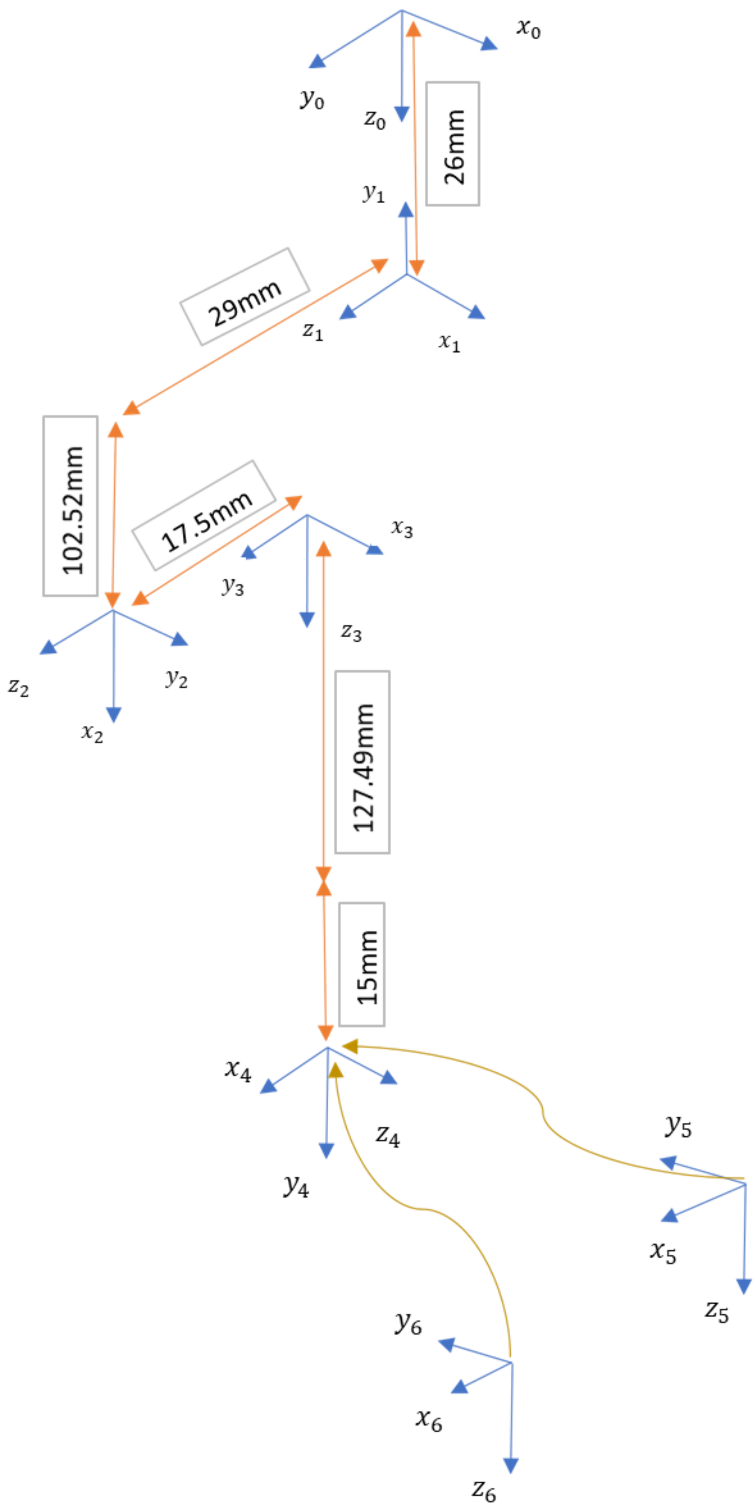


Fig.5: Assignment of frames to joints

Frames	θ_i	d_i	a_i	α_i
0->1	$\dot{\theta}_1$	26mm	0	$-90°$
1->2	$-90° + \dot{\theta}_2$	29mm	102.52mm	$0°$
2->3	$\dot{\theta}_3 + 90°$	-17.5mm	0	$90°$
3->4	$\dot{\theta}_4 + 90°$	142.49mm	0	$90°$
4->5	$\dot{\theta}_5$	0	0	$-90°$
5->6	$\dot{\theta}_6$	0	0	$0°$

5.2) <u>Forward Kinematics</u>:

The transformation between two frames *i-1* and *i* ,is obtained by substituting the values obtained from DH table to the matrix:

$$T_i^{i-1} = Rot_{z,\boldsymbol{\theta}_i} * Trans_{z,d_i} * Trans_{x,a_i} * Rot_{x,\alpha_i}$$

$$T_i^{i-1} = \begin{bmatrix} \cos\boldsymbol{\theta}_i & -\sin\boldsymbol{\theta}_i . \cos\alpha_i & \sin\boldsymbol{\theta}_i . \sin\alpha_i & a_i . \cos\boldsymbol{\theta}_i \\ \sin\boldsymbol{\theta}_i & \cos\boldsymbol{\theta}_i \cos\alpha_i & -\cos\boldsymbol{\theta}_i \sin\alpha_i & a_i \sin\boldsymbol{\theta}_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The final transformation from {0} to {6}, is obtained by multiplication between these matrices:

$$T_6^0 = T_1^0 * T_2^1 * T_3^2 * T_4^3 * T_5^4 * T_6^5$$

After obtaining the transformation, the coordinates of the end-effector in base frame is given by:

$$p^0 = T_6^0 * p^6$$

, where p^6 is $[0 , 0 \ ,0 \ ,1]^T$ and p^0 is $[x , y \ , z \ ,1]^T$

To find the final transformation matrix (transformation from frame 0 to frame 6), we have written a python script and find the final transformation matrix parametrically. The parametric final transformation matrix (output of the program) is shown below. Please note that t1 is $\dot{\theta}_1$, t2 is $\dot{\theta}_2$ and so on.

((-(-sin(t₂)·sin(t₃)·cos(t₁) + cos(t₁)·cos(t₂)·cos(t₃))·sin(t₄) - sin(t₁)·cos

((-(-sin(t₁)·sin(t₂)·sin(t₃) + sin(t₁)·cos(t₂)·cos(t₃))·sin(t₄) + cos(t₁)·cos

((-sin(t₂)·sin(t₃) + cos(t₂)·cos(

$\lfloor$

$(t_4))\cdot\cos(t_5) + (\sin(t_2)\cdot\cos(t_1)\cdot\cos(t_3) + \sin(t_3)\cdot\cos(t_1)\cdot\cos(t_2))\cdot\sin(t_5))\cdot c$

$(t_4))\cdot\cos(t_5) + (\sin(t_1)\cdot\sin(t_2)\cdot\cos(t_3) + \sin(t_1)\cdot\sin(t_3)\cdot\cos(t_2))\cdot\sin(t_5))\cdot c$

$t_3))\cdot\sin(t_5) - (-\sin(t_2)\cdot\cos(t_3) - \sin(t_3)\cdot\cos(t_2))\cdot\sin(t_4)\cdot\cos(t_5))\cdot\cos(t_6) -$

0

$os(t_6) + (-(-\sin(t_2)\cdot\sin(t_3)\cdot\cos(t_1) + \cos(t_1)\cdot\cos(t_2)\cdot\cos(t_3))\cdot\cos(t_4) + \sin($

$os(t_6) + (-(-\sin(t_1)\cdot\sin(t_2)\cdot\sin(t_3) + \sin(t_1)\cdot\cos(t_2)\cdot\cos(t_3))\cdot\cos(t_4) - \sin($

$(-\sin(t_2)\cdot\cos(t_3) - \sin(t_3)\cdot\cos(t_2))\cdot\sin(t_6)\cdot\cos(t_4)$

$t_1)\cdot\sin(t_4))\cdot\sin(t_6) \; -((-(-\sin(t_2)\cdot\sin(t_3)\cdot\cos(t_1) + \cos(t_1)\cdot\cos(t_2)\cdot\cos(t_3))$

$t_4)\cdot\cos(t_1))\cdot\sin(t_6) \; -((-(-\sin(t_1)\cdot\sin(t_2)\cdot\sin(t_3) + \sin(t_1)\cdot\cos(t_2)\cdot\cos(t_3))$

$-((-\sin(t_2)\cdot$

$\cdot\sin(t_4) - \sin(t_1)\cdot\cos(t_4))\cdot\cos(t_5) + (\sin(t_2)\cdot\cos(t_1)\cdot\cos(t_3) + \sin(t_3)\cdot\cos(t$

$\cdot\sin(t_4) + \cos(t_1)\cdot\cos(t_4))\cdot\cos(t_5) + (\sin(t_1)\cdot\sin(t_2)\cdot\cos(t_3) + \sin(t_1)\cdot\sin(t$

$\sin(t_3) + \cos(t_2)\cdot\cos(t_3))\cdot\sin(t_5) - (-\sin(t_2)\cdot\cos(t_3) - \sin(t_3)\cdot\cos(t_2))\cdot\sin($

0

$_1)\cdot\cos(t_2))\cdot\sin(t_5))\cdot\sin(t_6) + (-(-\sin(t_2)\cdot\sin(t_3)\cdot\cos(t_1) + \cos(t_1)\cdot\cos(t_2)\cdot c$

$_3)\cdot\cos(t_2))\cdot\sin(t_5))\cdot\sin(t_6) + (-(-\sin(t_1)\cdot\sin(t_2)\cdot\sin(t_3) + \sin(t_1)\cdot\cos(t_2)\cdot c$

$t_4)\cdot\cos(t_5))\cdot\sin(t_6) - (-\sin(t_2)\cdot\cos(t_3) - \sin(t_3)\cdot\cos(t_2))\cdot\cos(t_4)\cdot\cos(t_6)$

$os(t_3))\cdot cos(t_4) + sin(t_1)\cdot sin(t_4))\cdot cos(t_6)$ $-(-(-sin(t_2)\cdot sin(t_3)\cdot cos(t_1) + cos$

$os(t_3))\cdot cos(t_4) - sin(t_4)\cdot cos(t_1))\cdot cos(t_6)$ $-(-(-sin(t_1)\cdot sin(t_2)\cdot sin(t_3) + sin$

$(-sin(t_2$

$(t_1)\cdot cos(t_2)\cdot cos(t_3))\cdot sin(t_4) - sin(t_1)\cdot cos(t_4))\cdot sin(t_5) + (sin(t_2)\cdot cos(t_1)\cdot co$

$(t_1)\cdot cos(t_2)\cdot cos(t_3))\cdot sin(t_4) + cos(t_1)\cdot cos(t_4))\cdot sin(t_5) + (sin(t_1)\cdot sin(t_2)\cdot co$

$)\cdot sin(t_3) + cos(t_2)\cdot cos(t_3))\cdot cos(t_5) + (-sin(t_2)\cdot cos(t_3) - sin(t_3)\cdot cos(t_2))\cdot si$

0

$s(t_3) + sin(t_3)\cdot cos(t_1)\cdot cos(t_2))\cdot cos(t_5)$ $-11.5\cdot sin(t_1) + 142.49\cdot sin(t_2)\cdot cos(t$

$s(t_3) + sin(t_1)\cdot sin(t_3)\cdot cos(t_2))\cdot cos(t_5)$ $142.49\cdot sin(t_1)\cdot sin(t_2)\cdot cos(t_3) + 102$

$n(t_4)\cdot sin(t_5)$ $-142.49\cdot sin(t_2)\cdot sin$

$_1)\cdot cos(t_3) + 102.5\cdot sin(t_2)\cdot cos(t_1) + 142.49\cdot sin(t_3)\cdot cos(t_1)\cdot cos(t_2)$

$.5\cdot sin(t_1)\cdot sin(t_2) + 142.49\cdot sin(t_1)\cdot sin(t_3)\cdot cos(t_2) + 11.5\cdot cos(t_1)$

$(t_3) + 142.49\cdot cos(t_2)\cdot cos(t_3) + 102.5\cdot cos(t_2) + 26$

1

5.3) Inverse Kinematics:

Inverse kinematics is the determination of joint angles for a given end-effector position. For this project,we have used the Jacobian method to calculate the inverse kinematics as the manipulator has 6 DOF, Jacobian matrix will be 6X6 square matrix making the Jacobian inverse possible to calculate.

Jacobian Matrix, gives the relation between joint velocity and cartesian end effector velocity (For velocity kinematics). The manipulator used in this project has all the joints are of revolute type.

$$\dot{X} = \begin{bmatrix} J_v \\ J_w \end{bmatrix} \dot{q}_{6\times 1} \tag{1}$$

Where X is the position of the end-effector

q is the set of joint angles,

J_v is the linear velocity Jacobian and J_w is the angular velocity Jacobian.

General form of Jacobian for i^{th} joint,

$$J_i = \begin{bmatrix} J_v \\ J_w \end{bmatrix} = \begin{bmatrix} z_{i-1} \times (o_n - o_{i-1}) \\ z_{i-1} \end{bmatrix} \tag{2}$$

Effective Jacobian is given by,

$$J = [J_1 \quad J_2 \quad J_3 \quad J_4 \quad J_5 \quad J_6] \tag{3}$$

To calculate (2), z_i is the first three elements in third column of T_i^0 matrix.

And o_i is the first three elements of last column of T_i^0 matrix.

Jacobian for the first joint (i=1) is given by,

$$J_1 = \begin{bmatrix} z_0 \times (o_6 - o_0) \\ z_0 \end{bmatrix}$$

Where $z_0 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$ (third column of identity matrix)

And $o_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ (origin coordinate)

Therefore,

$$(3) \Rightarrow J = \begin{bmatrix} z_0 \times (o_6 - o_0) & z_1 \times (o_6 - o_1) & z_2 \times (o_6 - o_2) & z_3 \times (o_6 - o_3) & z_4 \times (o_6 - o_4) & z_5 \times (o_6 - o_5) \\ z_0 & z_1 & z_2 & z_3 & z_4 & z_5 \end{bmatrix}$$

The relation used for determining joint angles are,

$$q_{current} = q_{previous} + \dot{q}_{current} * (\Delta t)$$

$$q_{current} = q_{previous} + \left\{ J_{inv} * \frac{(\Delta x)}{(\Delta t)} \right\} * (\Delta t) \qquad From(1)$$

$$q_{current} = q_{previous} + J_{inv} * (\Delta x)$$

To find the inverse kinematics, we have used the Jacobian method mentioned above. We have found the Jacobian matrix parametrically using python program. The output of the program and hence the parametric Jacobian matrix is shown below. Please note that t1 is $\dot{\theta}_1$, t2 is $\dot{\theta}_2$ and so on.

$-142.49 \cdot \sin(t_1) \cdot \sin(t_2) \cdot \cos(t_3)$ - $102.5 \cdot \sin(t_1) \cdot \sin(t_2)$ - $142.49 \cdot \sin(t_1) \cdot \sin($

$-11.5 \cdot \sin(t_1) + 142.49 \cdot \sin(t_2) \cdot \cos(t_1) \cdot \cos(t_3) + 102.5 \cdot \sin(t_2) \cdot \cos(t_1) + 14$

2.

0

0

0

1

$t_3) \cdot \cos(t_2)$ - $11.5 \cdot \cos(t_1)$

$49 \cdot \sin(t_3) \cdot \cos(t_1) \cdot \cos(t_2)$

$-(142.49 \cdot \sin(t_1) \cdot \sin(t_2) \cdot \cos(t_3) + 102.5 \cdot \sin(t_1) \cdot s$

$(-142.49 \cdot \sin(t_2) \cdot \sin(t_3) + 142.49 \cdot \cos(t_2) \cdot \cos(t_3) +$

$(-142.49 \cdot \sin(t_2) \cdot \sin(t_3) + 142.49 \cdot \cos(t_2) \cdot \cos(t_3) +$

$in(t_2) + 142.49 \cdot sin(t_1) \cdot sin(t_3) \cdot cos(t_2) + 11.5 \cdot cos(t_1)) \cdot sin(t_1) - (-11.5 \cdot sin(t$

$-sin(t_1)$

$cos(t_1)$

0

$102.5 \cdot cos(t_2)) \cdot cos(t_1)$

$102.5 \cdot cos(t_2)) \cdot sin(t_1)$

$_1) + 142.49 \cdot sin(t_2) \cdot cos(t_1) \cdot cos(t_3) + 102.5 \cdot sin(t_2) \cdot cos(t_1) + 142.49 \cdot sin(t_3) \cdot$
c

$os(t_1) \cdot cos(t_2)) \cdot cos(t_1) \quad -(142.49 \cdot sin(t_1) \cdot sin(t_2) \cdot cos(t_3) + 142.49 \cdot sin(t_1) \cdot sin$

$(-142.49 \cdot sin(t_2) \cdot sin(t_3) + 142.49 \cdot cos(t_2) \cdot cos(t_3)) \cdot cos(t_1)$

$(-142.49 \cdot sin(t_2) \cdot sin(t_3) + 142.49 \cdot cos(t_2) \cdot cos(t_3)) \cdot sin(t_1)$

$(t_3) \cdot cos(t_2) - 17.5 \cdot cos(t_1)) \cdot sin(t_1) - (17.5 \cdot sin(t_1) + 142.49 \cdot sin(t_2) \cdot cos(t_1) \cdot$

$-sin(t_1)$

$\cos(t_1)$

0

$(-142.49\cdot\sin$

$-(-142.49\cdot\mathrm{si}$

$\cos(t_3) + 142.49\cdot\sin(t_3)\cdot\cos(t_1)\cdot\cos(t_2))\cdot\cos(t_1)\ \ -(\sin(t_1)\cdot\sin(t_2)\cdot\cos(t_3) +$

$(t_2)\cdot\sin(t_3) + 142.49\cdot\cos(t_2)\cdot\cos(t_3))\cdot(\sin(t_1)\cdot\sin(t_2)\cdot\cos(t_3) + \sin(t_1)\cdot\sin($

$\mathrm{n}(t_2)\cdot\sin(t_3) + 142.49\cdot\cos(t_2)\cdot\cos(t_3))\cdot(\sin(t_2)\cdot\cos(t_1)\cdot\cos(t_3) + \sin(t_3)\cdot\cos$

$\sin(t_1)\cdot\sin(t_3)\cdot\cos(t_2))\cdot(142.49\cdot\sin(t_2)\cdot\cos(t_1)\cdot\cos(t_3) + 142.49\cdot\sin(t_3)\cdot\cos$

$\sin(t_2)\cdot\cos$

$\sin(t_1)\cdot\sin$

$-\mathrm{si}$

$t_3)\cdot\cos(t_2)) - (-\sin(t_2)\cdot\sin(t_3) + \cos(t_2)\cdot\cos(t_3))\cdot(142.49\cdot\sin(t_1)\cdot\sin(t_2)\cdot\mathrm{co}$

$(t_1)\cdot\cos(t_2)) + (-\sin(t_2)\cdot\sin(t_3) + \cos(t_2)\cdot\cos(t_3))\cdot(142.49\cdot\sin(t_2)\cdot\cos(t_1)\cdot\mathrm{c}$

$(t_1)\cdot\cos(t_2)) + (142.49\cdot\sin(t_1)\cdot\sin(t_2)\cdot\cos(t_3) + 142.49\cdot\sin(t_1)\cdot\sin(t_3)\cdot\cos(t$

$(t_1)\cdot\cos(t_3) + \sin(t_3)\cdot\cos(t_1)\cdot\cos(t_2)$

$(t_2)\cdot\cos(t_3) + \sin(t_1)\cdot\sin(t_3)\cdot\cos(t_2)$

$\mathrm{n}(t_2)\cdot\sin(t_3) + \cos(t_2)\cdot\cos(t_3)$

$\mathrm{s}(t_3) + 142.49\cdot\sin(t_1)\cdot\sin(t_3)\cdot\cos(t_2))$

$os(t_3) + 142.49\cdot sin(t_3)\cdot cos(t_1)\cdot cos(t_2))$

$_2))\cdot(sin(t_2)\cdot cos(t_1)\cdot cos(t_3) + sin(t_3)\cdot cos(t_1)\cdot cos(t_2))$

$-(-(-sin(t_2)\cdot sin(t_3)\cdot$

$-(-(-sin(t_1)\cdot sin(t_2)\cdot$

0

0

0

$cos(t_1) + cos(t_1)\cdot cos(t_2)\cdot cos(t_3))\cdot sin(t_4) - sin(t_1)\cdot cos(t_4))\cdot sin(t_5) + (sin(t$

$sin(t_3) + sin(t_1)\cdot cos(t_2)\cdot cos(t_3))\cdot sin(t_4) + cos(t_1)\cdot cos(t_4))\cdot sin(t_5) + (sin(t$

$(-sin(t_2)\cdot sin(t_3) + cos(t_2)\cdot cos(t_3))\cdot cos(t_5) + (-sin(t_2)\cdot cos(t_3) - sin(t_3$

$_2)\cdot cos(t_1)\cdot cos(t_3) + sin(t_3)\cdot cos(t_1)\cdot cos(t_2))\cdot cos(t_5)\ \ -(-(-sin(t_2)\cdot sin(t_3)\cdot co$

$_1)\cdot sin(t_2)\cdot cos(t_3) + sin(t_1)\cdot sin(t_3)\cdot cos(t_2))\cdot cos(t_5)\ \ -(-(-sin(t_1)\cdot sin(t_2)\cdot si$

$)\cdot cos(t_2))\cdot sin(t_4)\cdot sin(t_5)$

0

0

0

$s(t_1) + cos(t_1)\cdot cos(t_2)\cdot cos(t_3))\cdot sin(t_4) - sin(t_1)\cdot cos(t_4))\cdot sin(t_5) + (sin(t_2)$

$n(t_3) + \sin(t_1)\cdot\cos(t_2)\cdot\cos(t_3))\cdot\sin(t_4) + \cos(t_1)\cdot\cos(t_4))\cdot\sin(t_5) + (\sin(t_1)$

$(-\sin(t_2)\cdot\sin(t_3) + \cos(t_2)\cdot\cos(t_3))\cdot\cos(t_5) + (-\sin(t_2)\cdot\cos(t_3) - \sin(t_3)\cdot$

$\cdot\cos(t_1)\cdot\cos(t_3) + \sin(t_3)\cdot\cos(t_1)\cdot\cos(t_2))\cdot\cos(t_5)$

$\cdot\sin(t_2)\cdot\cos(t_3) + \sin(t_1)\cdot\sin(t_3)\cdot\cos(t_2))\cdot\cos(t_5)$

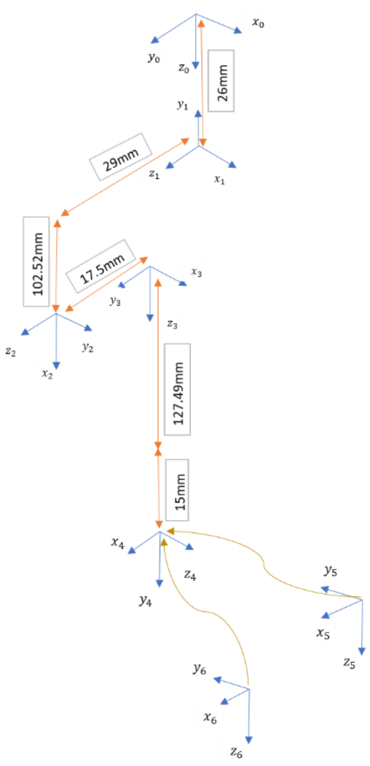
$\cos(t_2))\cdot\sin(t_4)\cdot\sin(t_5)$

CHAPTER 6

FORWARD KINEMATICS VALIDATION

- Validation method: Geometrical validation
- Number of validations: 5
- Validation Process:
  - Geometrically find the position of end effector for known value of set of thetas
  - Write a program to find the position of end effector using DH parameter for the above theta value
  - Compare the result of program with the result obtained from geometrical method

1. 1st validation: $(\dot{\theta}_1,\ \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4,\ \dot{\theta}_5,\ \dot{\theta}_6)$ = (0, 0, 0, 0, 0, 0)
   - Position of end effector using geometry:



$$\Rightarrow \quad \text{P06} = \begin{bmatrix} 0 \\ 11.5 \\ 271.01 \\ 1 \end{bmatrix}$$

Fig.6 : Frames assigned

- Position of end effector from DH parameter (output of python program):

$$\text{T06} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 11.5 \\ 0 & 0 & 1 & 270.99 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow \quad \text{P06} = \begin{bmatrix} 0 \\ 11.5 \\ 270.99 \\ 1 \end{bmatrix}$$

2. 2nd Validation: ($\dot{\theta}_1$, $\dot{\theta}_2$, $\dot{\theta}_3$, $\dot{\theta}_4$, $\dot{\theta}_5$, $\dot{\theta}_6$) = (0, $\pi/2$, 0, 0, 0, 0):
   - Position of end effector using geometry: It is obvious from the figure shown in the first validation that P06 can be given as below:

$$P06 = \begin{bmatrix} 245.01 \\ 11.5 \\ 26 \\ 1 \end{bmatrix}$$

   - Position of end effector from DH parameter (output of python program):

$$T06 = \begin{bmatrix} 0 & 0 & 1 & 244.99 \\ 1 & 0 & 0 & 11.5 \\ 0 & 1 & 0 & 26.0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow \quad P06 = \begin{bmatrix} 244.99 \\ 11.5 \\ 26.0 \\ 1 \end{bmatrix}$$

3. 3nd Validation: ($\dot{\theta}_1$, $\dot{\theta}_2$, $\dot{\theta}_3$, $\dot{\theta}_4$, $\dot{\theta}_5$, $\dot{\theta}_6$) = (0,0 , $\pi/2$, 0, 0, 0):
   - Position of end effector using geometry: It is obvious from the figure shown in the first validation that P06 can be given as below:

$$P06 = \begin{bmatrix} 142.49 \\ 11.5 \\ 128.52 \\ 1 \end{bmatrix}$$

   - Position of end effector from DH parameter (output of python program):

$$T06 = \begin{bmatrix} 0 & 0 & 1 & 142.49 \\ 1 & 0 & 0 & 11.5 \\ 0 & 1 & 0 & 128.5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow P06 = \begin{bmatrix} 142.49 \\ 11.5 \\ 128.5 \\ 1 \end{bmatrix}$$

4. 4nd Validation: ($\dot{\theta}_1$, $\dot{\theta}_2$, $\dot{\theta}_3$, $\dot{\theta}_4$, $\dot{\theta}_5$, $\dot{\theta}_6$) = (0, 0, 0, $\pi/2$, 0, 0):
   - Position of end effector using geometry: It is obvious from the figure shown in the first validation that P06 can be given as below:

$$P06 = \begin{bmatrix} 0 \\ 11.5 \\ 271.01 \\ 1 \end{bmatrix}$$

   - Position of end effector from DH parameter (output of python program):

$$T06 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 11.5 \\ 0 & 0 & 1 & 270.99 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow \quad P06 = \begin{bmatrix} 0 \\ 11.5 \\ 270.99 \\ 1 \end{bmatrix}$$

5. 5nd Validation: $(\dot{\theta}_1,\ \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4,\ \dot{\theta}_5,\ \ \dot{\theta}_6)$ = ($\pi/2$, 0, 0, 0, 0, 0):
    - Position of end effector using geometry: It is obvious from the figure shown in the first validation that P06 can be given as below:

$$P07 = \begin{bmatrix} -11.5 \\ 0 \\ 271.01 \\ 1 \end{bmatrix}$$

    - Position of end effector from DH parameter (output of python program):

$$T06 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 11.5 \\ 0 & 0 & 1 & 270.99 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow \quad P07 = \begin{bmatrix} -11.5 \\ 0 \\ 270.99 \\ 1 \end{bmatrix}$$

CHAPTER 7

INVERSE KINEMATICS VALIDATION

For Inverse kinematics validation, we have moved the end effector of the robot to draw a circle of the radius 200 mm. We have used inverse of Jacobian to find the value of joint angle at each iteration during drawing the circle. The initial configuration of robot is as follow:

$(\dot{\theta}_1,\ \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4,\ \dot{\theta}_5,\ \ \dot{\theta}_6)$ = $(0,\ \pi/2,\ \ \text{-}\pi/4,\ \pi/4,$ 0, 0)

Corresponding initial position of the end effector is as follow:

(x, y, z)= $(203.255, 11.5, 126.75)$

The circle will be drawn in Z = 126.75 mm plane and centre of the circle is (x, y) ≅ $(0,11.5)$ and radius is 200 mm.

Initial pose of the robot is as follow.

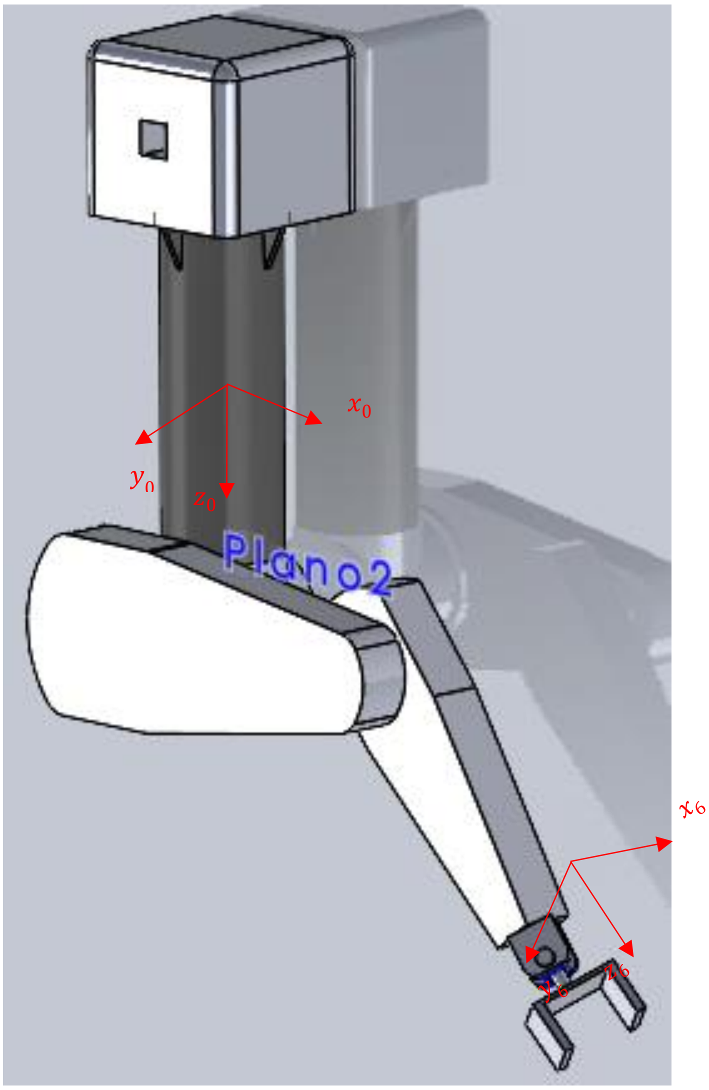


Fig.7: Initial pose of the robot while drawing the circle

We have used the python program to draw the position of the end effector while drawing the circle. The below figure shows the position of the end effector while drawing the circle.

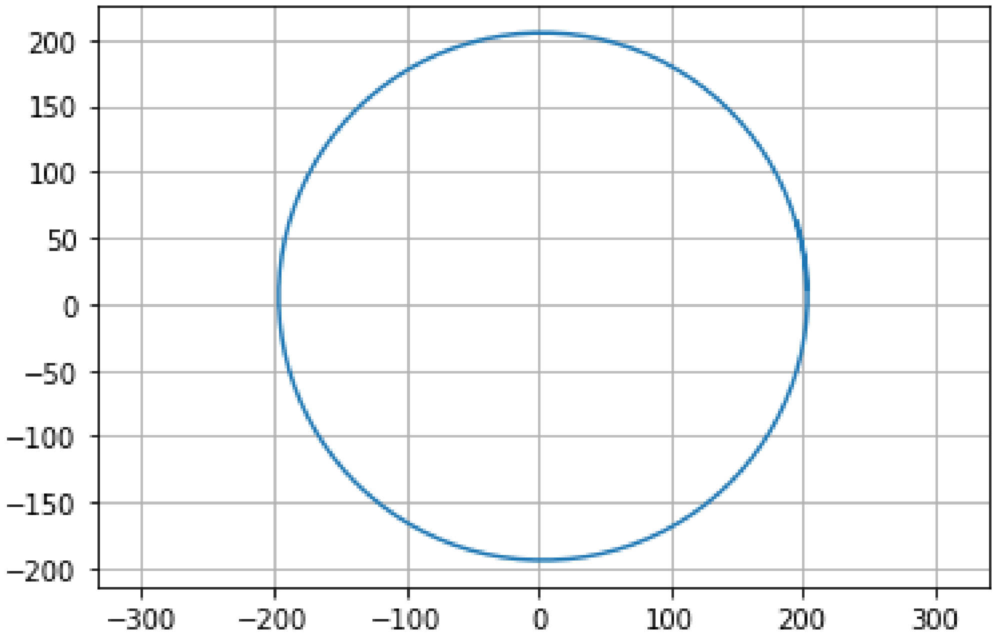


Fig.8 :2D figure of the end effector position

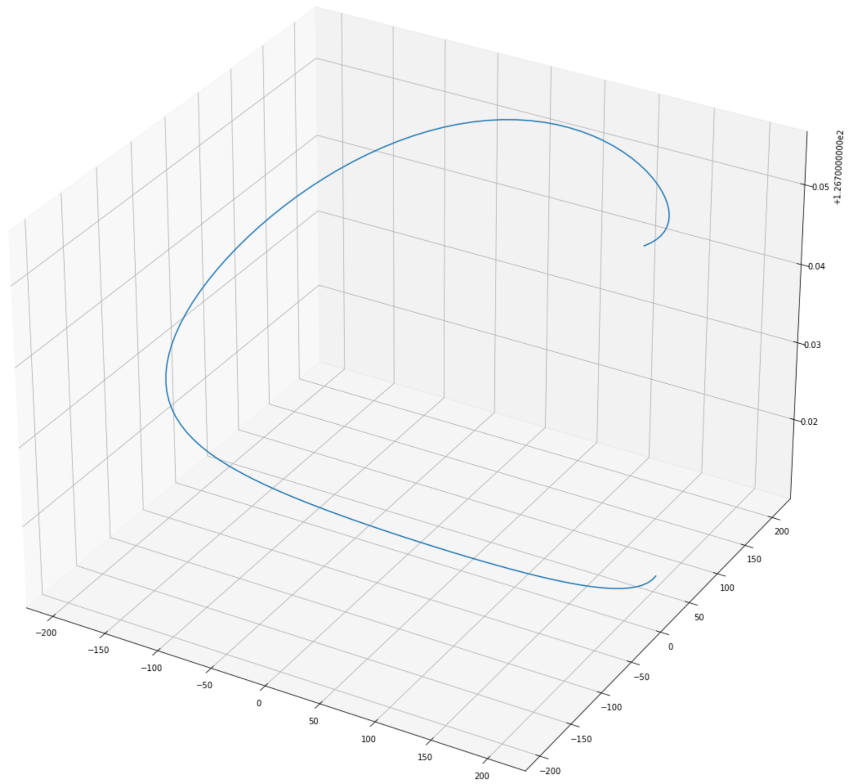


Fig.9: 3D figure of the end effector position

From 2D figure, we can say that the centre and radius of the trajectory followed by robot while drawing circle is same as the centre and radius of the circle to be drawn.

However, from the 3D figure, we can say that trajectory is not exactly in the Z = 126.75 mm plane. It has a deviation of about 0.05 mm at some points. This deviation is due to the fact that while following trajectory, robot has encounter singularities.

CHAPTER 8

WORKSPACE STUDY

In robotics, the workspace of a robot manipulator is defined as the set of points that can be reached by its end-effector. For our project, the workspace was computed considering the crab to be at a fixed position. The arms's workspace was plotted using the end effector position from the final transformation matrix. The first two joint angles ($\boldsymbol{\theta_1}, \boldsymbol{\theta_2}$) were varied to their rotation limit for computing the maximum reachable workspace. Using this logic, a python script was developed and is submitted along with the package.

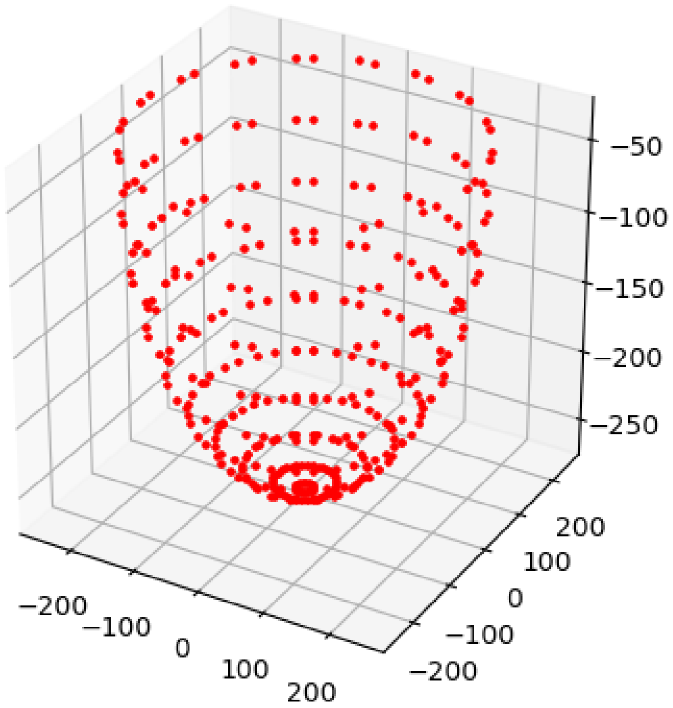


Fig.10 : Workspace of the manipulator

CHAPTER 9

ASSUMPTIONS

The following assumptions were considered for the project.:

- All the links were considered as rigid bodies and resist bending when a load is applied.
- Frictional forces were neglected. Work done by the friction is zero.
- There is enough distance between the adjacent stacks such that no collision takes place when a material is picked or placed by robot.
- All the path planning details are available to the robot. i.e., no path planning will be performed by the robot.
- No expansion or compression of the slotted rail occurred during the robot operation.

CHAPTER 10

CONTROL METHOD

Controller used : joint_trajectory_controller

We have used Joint_trajectory_controller for the control of the gantry robot. Joint_trajectory_controller is used for executing joint-space trajectories on a group of joints. Since the gantry robot has multiple joints and the purpose of the gantry robot is to do pick and place operation, robot has to follow a specific trajectory using group of joints to achieve the intended task. Hence, Joint_trajectory_controller is used for the control of gantry robot.

Moreover, this controller is templated to work with multiple hardware interface types. Currently joints with position, velocity and effort interfaces are supported. It is also possible to support new hardware interfaces, or alternative mappings to an already supported interface (e.g. a proxy controller for generating effort commands).

Apart from the above-mentioned advantages, this controller has also offered following advantages.

1. Realtime-safe implementation.
2. Proper handling of wrapping (continuous) joints.
3. Robust to system clock changes: Discontinuous system clock changes do not cause discontinuities in the execution of already queued trajectory segments.

CHAPTER 11

Gazebo and RViz Visualisation

The simulation link of the video is provided:

https://drive.google.com/file/d/1DO3e0nfKYU_HJ2XyD0Ok-1KXF0CXa9xn/view?usp=sharing

CHAPTER 12

PROBLEMS FACED

1. Errors were encountered with the initially designed model, during urdf conversion and also while launching in gazebo. The initial idea had 2 frame stands and the girder was to be placed on top the slots of these frames. This violated the single parent-single child relation, as the girder (child) had 2 parents(frame stands). To overcome this issue, the frames were joined together and redesigned. And also mating was applied to only one of the slotted rails.

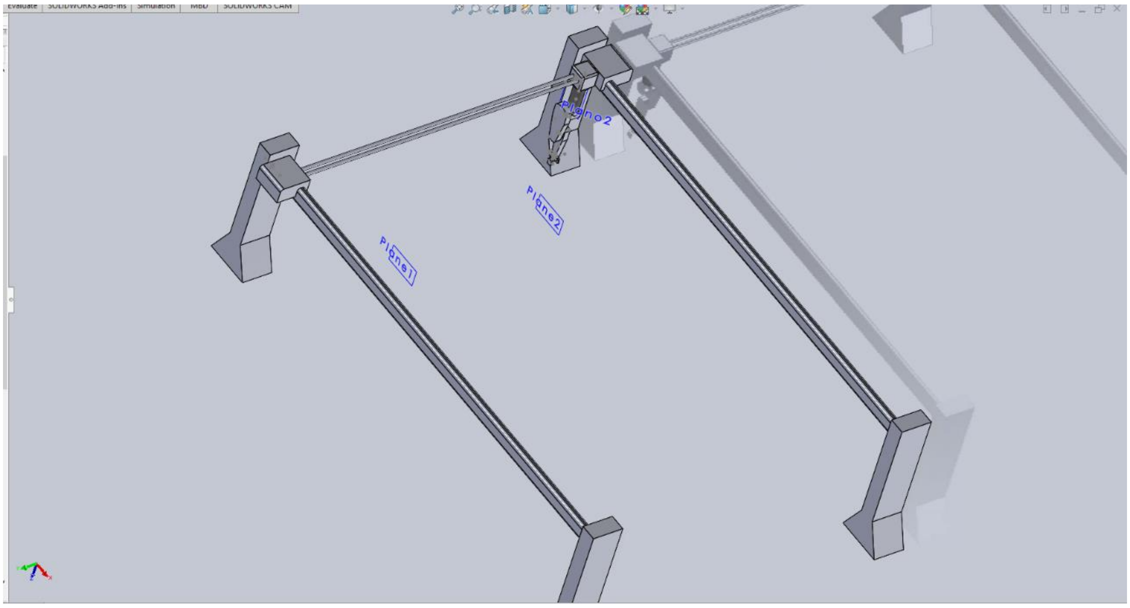


Fig.11 : Initial Design

2. Initially when the model was launched in Gazebo, it appeared as if the mates were broken and most of the links were out of order. To overcome this, the base link was designed in various orientations and also some parts were subjected to minute design changes so that additional mating constraints can be added.
3. Torque limit and velocity limit were zero in urdf, so robot was not moving.
4. It was very difficult to tune the PID gains of the manipulator.

CHAPTER 13

LESSONS LEARNED

Following are the outcomes, we learnt from this project:

1. Studied the forward and inverse kinematics of a higher DOF robot used in the real world.
2. Understood the process of modeling robots using Solidworks.
3. Exporting a Solidworks file into URDF and implementation of a proper parent-child tree structure was achieved.
4. Learnt the process of integrating various sensors and controllers to the model for the simulation.
5. Integration of the controller was studied and also how to approach gain tuning process.
6. Understood the process of simulating the robot in a real-world environment.
7. Trajectory planning of the manipulator using Moveit.
8. Learnt how to use joint trajectory controller.
9. Forward and inverse validation techniques studied.
10. Path planning of the robot using inverse kinematics, was briefly studied.

CHAPTER 14

CONCLUSION

- A gantry robot for pick and place operation has been proposed and modelled.
- The forward and inverse kinematics of the manipulator arm have been carried out.
- Validation of forward kinematics was done using geometric, we have found out the end effector position using geometry for known joint angle and was compared with values from the transformation matrix.
- For inverse kinematics validation, we have moved our end effector to a known position and a circle of 200mm radius was plotted using inverse Jacobian matrix.
- The gantry robot is moved in the gazebo environment to mimic pick and place operation.

CHAPTER 15

FUTURE WORK

These are our ambitious ideas we hope to implement in the future:

1) Implementation of a gripper for pick and place operation
2) Carry out the forward and inverse kinematics validation for the whole gantry robot.
3) Implementing path planning for manipulator using inverse kinematics.
4) And also achieve pick and place operation using inverse kinematics.

CHAPTER 16

REFERENCES

[1] https://www.robots.com/faq/what-are-gantry-robots
[2] https://www.robots.com/robots/fanuc-m-900ia-200p
[3] FANUC-M-900iA-Series-Large-Payload-Robot-Brochure
[4] https://www.slideshare.net/kajolpanchal2/guntry-girder
[5] http://wiki.ros.org/joint_trajectory_controller
[6] http://wiki.ros.org/moveit
[7] https://moveit.ros.org/
[8] https://github.com/ros-planning/moveit