ENPM662- INTRODUCTION TO ROBOT MODELLING

Final Project Report

Modeling and Analysis of 6-DOF Robotic Arm for Palletizing

Submitted to- Dr. Sean W. Gart

Akshay Kurhade

UID-116914529

INDEX

Table of Contents

References

Introduction

Robotic arms with multiple degrees of freedom, such as UR-10, have become popular research platforms in robotics and mobile industrial automation. Such robotic arms can perform complex motions, including the grasping, precisely orienting the work object, precision movements. The robotic arm must keep its balance, self-collisions and collisions with obstacles in the working environment must be avoided and, if applicable, the trajectory of the end-effector must follow the constrained motion of a manipulated object in Cartesian space. The design of complex dynamic motions is achievable only using robot kinematics, which is an application of geometry to the study of arbitrary robotic chains. This report studies the problems of forward and inverse kinematics for the 'UR-10e Robotic Arm by Universal Robots'. The forward kinematics allow system developers to map any configuration of the robot from its own joint space to the three-dimensional physical space, whereas the inverse kinematics provide closed form solutions to finding joint configurations that drive the end effectors of the robot to desired points in the three-dimensional space. The proposed solution was made feasible through a decomposition into independent problems (learning the environment, interacting with other system components, co-ordination),

Fig. 1: Universal Robots UR-10e

I. Motivation

Palletizing is a basic requirement of any supply chain management. There are several industrial grade solutions available for palletizing which work well in pre-defined factory/warehouse structure. However, it becomes a difficult task to do such operations in limited space and dynamic locations. The supply chains must rely on human labor in such instances leaving room both for human error and risk of hazard. Ur-10e has a reachable radius of 1300mm/51.2" with a payload of 10kgs which is sufficient for most of the standard palletizing operations. At the same time, it has a footprint of just 190mm diameter making it easy to mount on mobile platforms, thus creating a compact palletizing solution. Ur-10 is a 6-R arm with all joints having no rotation constraints. Also, the robot is light-weight, fast and precise.

Fig. 2: Palletizing in operation using UR-10

II. UR-10

a) Why UR-10 specifically?

UR-10 is a 6-R robotic arm developed by Universal Robots. It has no revolution constraints, a very small footprint and flexible mounting options. It can also handle payloads up to 10kg/22lbs with a precision of ±0.05mm. The UR10e adheres to advanced collaborative standards making it Industry 4.0 ready.

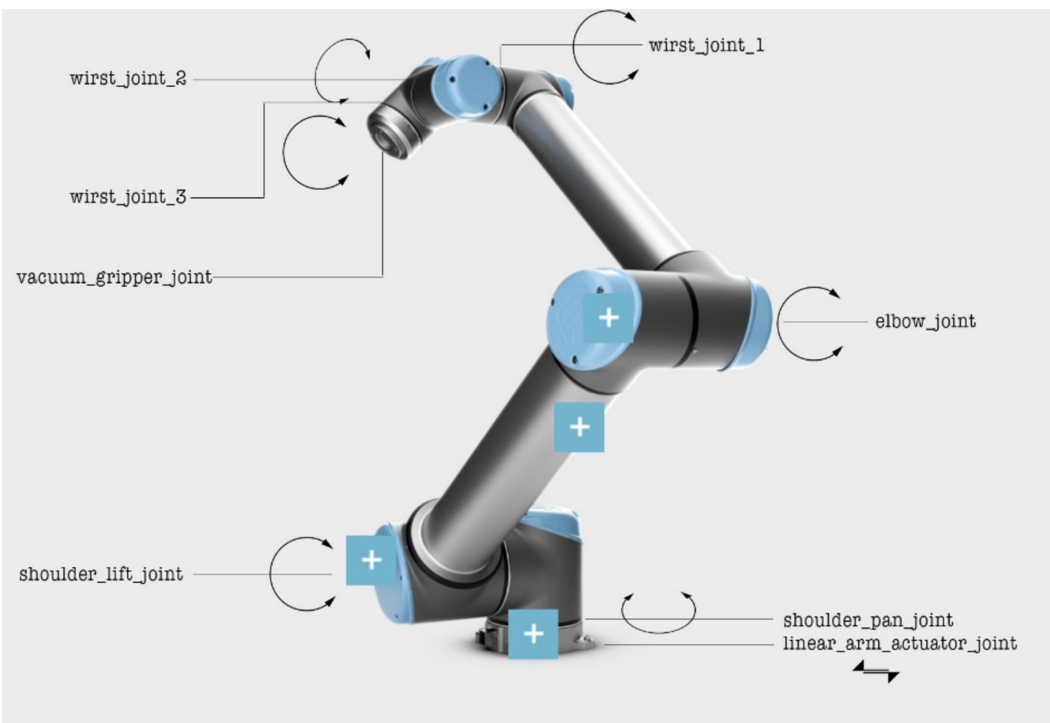


Fig. 3: UR-10 Joint Description

b) Technical Specifications-

Payload	10 kg / 22lbs
Reach	1300 mm / 51.2 in
Joint ranges	360 for all joints
Speed	Base and Shoulder joints: Max 120 "/s. All other joints: Max 180 "/s. Tool: Approx. 1 m/s / Approx. 39.4 in/s.
Repeatability	± 0.1mm / ± 0.0039 in (4 mils)
Footprint	Ø190mm
Degrees of freedom	6 rotating joints
Control box size (W x H x D)	475mm x 423mm x 268mm / 18.7 in x 16.7 in x 10.6 in
Control box I/O ports	16 digital in, 16 digital out, 2 analogue in, 2 analogue out
Tool I/O ports	2 digital in, 2 digital out, 2 analogue in
I/O power supply	24 V 2A in control box and 12 V/24 V 600mA in tool
Communication	TCP/IP 100 Mbit: IEEE 802.3u, 100BASE-TX Ethernet socket & Modbus TCP
Programming	PolyScope graphical user interface on 12” touchscreen with mounting
Noise	Comparatively noiseless
IP classification	IP54
Power consumption	Approx. 350W using a typical program
Collaboration operation	Collaborative operation according to ISO 10218-1:2011
Temperature	The robot can work in a temperature range of 0-50 "C
Power supply	100-240 VAC, 50-60 Hz
Cabling	Cable between robot and control box (6m / 236 in) Cable between touchscreen and control box (4.5m / 177 in)

c) Axis Movement:

Axis/Joint	Working Range	Maximum Speed
Base	± 360	±120°/Sec
Shoulder	± 360	±120°/Sec
Elbow	± 360	±180°/Sec
Wrist 1	± 360	±180°/Sec
Wrist 2	± 360	±180°/Sec
Wrist 3	± 360	±180°/Sec

Pose Repeatability: +/- 0.05 mm, with payload, per ISO 9283

d) Workspace and singularities-

The workspace of the UR10e robot extends 1300 mm from the base joint. The workspace can be extended by attaching the arm to a linear prismatic joint at the base, increasing the height constraint if any. It is important to consider the cylindrical volume directly above and directly below the robot base when choosing a mounting place for the robot. Moving the tool close to the cylindrical volume should be avoided because it causes the joints to move fast even when the tool is moving slowly, which causes the robot to work inefficiently and makes it difficult to conduct a risk assessment.

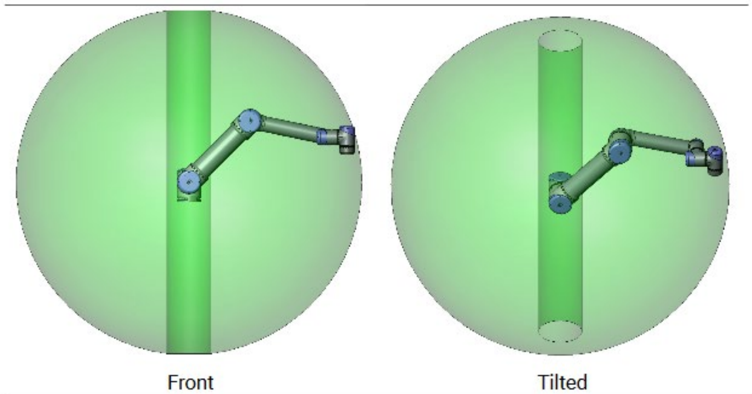


Fig. 4: UR-10 workspace

Singularities-

Singularities will occur when two or more concurrent joints align each other. Certain areas of the workspace should receive attention regarding singularity hazards, due to the physical properties of the robot arm. One area is defined for radial motions, when the wrist 1 joint is at a distance of at least 1100mm from the base of the robot. The other area is within 300mm of the base of the robot, when moving in the tangential direction

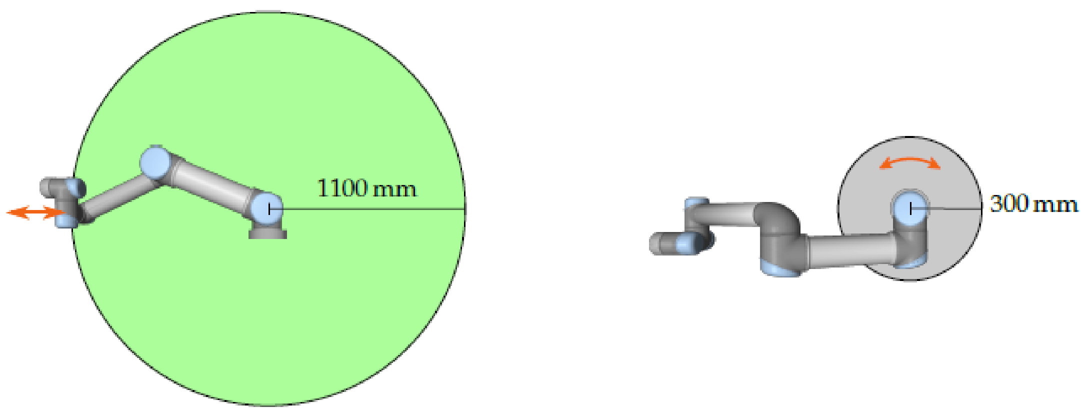


Fig. 5: Singularities and avoidable regions

III. Grasping-

For the scope of the project a RobotiQ-85 gripper was used in two-finger grab configuration. The gripper design enables both internal and external parallel gripping, as well as encompassing grip mode. The gripper is also equipped with automatic part detection and position feedback

Fig. 6: RobotiQ-85 gripper

Stroke	85 mm 3.3 in
Grip Force	20 to 235 N 4.5 to 50 lbf
Form-fit Grip Payload	5 kg 11lbs
Gripper Weight	0.9 kg 2lbs
Closing speed	20 to 150 mm/s 0.8 to 5.9 in/s

IV. Robotic Arm Kinematics-

A robot kinematic chain is an articulated manipulator that interacts with the environment and is typically described as an assembly of robotic links connected by (rotary) joints. The joints rotate and control the relative angular positioning of the links of the manipulator. Not all combinations of joints' positions in the chain are valid, because some combinations lead to collisions between the links of the chain or with some fixed item of the environment, such as the floor or a wall. All the valid combinations of joint values form the joint space. The term degrees of freedom (DOF) refers to the number of joints in a kinematic chain; clearly, more DOF imply more flexibility in the motion of the chain.

a) Forward Kinematics-

In order to find the forward kinematics, we use the Denavit-Hartenberg method which is the most common method for deriving the manipulator kinematics that lead to the derivation of the Forward Kinematic equations. In Denavit-Hartenberg method, the homogeneous transformation matrix is represented by a product of four basic transformations. These transformations are two translations and two rotations parametrized by the four Denavit-Hartenberg parameters. Which can be represented as,

$$T_i^{i-1} = Rot_{zi}(\theta_i)Trans_{zi}(d_i)Trans_{xi}(a_i)Rot_{xi}(\alpha_i)$$

$$\begin{bmatrix} cos\,\theta_i & -sin\,\theta_i & 0 & 0 \\ sin\,\theta_i & cos\,\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos\,\alpha_i & -sin\,\alpha_i & 0 \\ 0 & sin\,\alpha_i & cos\,\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} cos\,\theta_i & -sin\,\theta_i\,cos\,\alpha_i & sin\,\theta_i\,sin\,\alpha_i & a_i\,cos\,\theta_i \\ sin\,\theta_i & cos\,\theta_i\,cos\,\alpha_i & -cos\,\theta_i\,sin\,x_i & a_i\,sin\,\theta_i \\ 0 & sin\,\alpha_i & cos\,\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

i) D-H Parameters for UR-10:

Frame assignment-

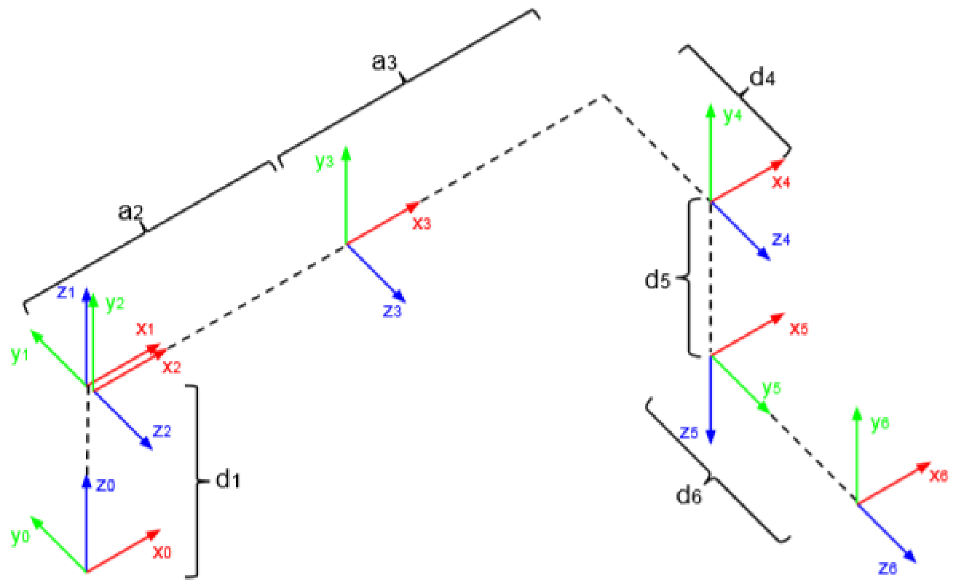


Fig. 7: UR-10 DH frame assignment

Based on the above DH- frame assignment, the D-H parameter table was constructed as follows-

Kinematics	θ	α	a	d
Joint 1	q_1	$\frac{\pi}{2}$	0	d_1
Joint 2	q_2	0	a_2	0
Joint 3	q_3	0	a_3	0
Joint 4	q_4	$\frac{\pi}{2}$	0	d_4
Joint 5	q_5	$-\frac{\pi}{2}$	0	d_5
Joint 6	q_6	0	0	d_6

ii) Homogenous Transformation Matrices-

HTM's were generated using the D-H parameter table

$$T_1^0 = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_2^1 = \begin{bmatrix} C_2 & -S_2 & 0 & a_2C_2 \\ S_2 & C_2 & 0 & a_2S_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_3^2 = \begin{bmatrix} C_3 & -S_3 & 0 & a_3C_3 \\ S_3 & C_3 & 0 & a_3S_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_4^3 = \begin{bmatrix} C_4 & 0 & S_4 & 0 \\ S_4 & 0 & -C_4 & 0 \\ 0 & 1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_5^4 = \begin{bmatrix} C_5 & 0 & -S_5 & 0 \\ S_5 & 0 & C_5 & 0 \\ 0 & -1 & 0 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_6^5 = \begin{bmatrix} C_6 & -S_6 & 0 & 0 \\ S_6 & C_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The multiplication of the equations is done in MATLAB to obtain the Transformation matrix T_6^0 which gives the position and orientation of the end effector as a function of the joint angles.

$$T_6^0 = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 = \begin{bmatrix} R_{11} & R_{12} & R_{13} & p_x \\ R_{21} & R_{22} & R_{23} & p_y \\ R_{31} & R_{32} & R_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Where,

$$R_{11} = c_6(s_{15} + c_5(c_4(c_{123} - c_1 s_{23}) - s_4(c_{12}s_3 + c_{13}s_2))) - s_6(c_4(c_{12}s_3 + c_{13}s_2) + s_4(c_{123} - c_1 s_{23}))$$

$$R_{12} = -s_6(s_{15} + c_5(c_4(c_{123} - c_1s_{23}) - s_4(c_{12}s_3 + c_{13}s_2))) - c_6(c_4(c_{12}s_3 + c_{13}s_2) + s_4(c_{123} - c_1s_{23}))$$

$$R_{13} = c_5s_1 - s_5(c_4(c_{123} - c_1s_{23}) - s_4(c_{12}s_3 + c_{13}s_2)$$

$$R_{21} = c_6(c_5(c_4(c_{23}s_1 - s_{123}) - s_4(c_2s_{13} + c_3s_{12})) - c_1s_5) - s_6(c_4(c_2s_{13} + c_3s_{12}) + s_4(c_{23}s_1 - s_{123}))$$

$$R_{22} = -c_6(c_4(c_2s_{13} + c_3s_{12}) + s_4(c_{23}s_1 - s_{123})) - s_6(c_5(c_4(c_{23}s_1 - s_{123}) - s_4(c_2s_{13} + c_3s_{12})) - c_1s_5)$$

$$R_{23} = -c_{15} - s_5(c_4(c_{23}s_1 - s_{123}) - s_4(c_2s_{13} + c_3s_{12}))$$

$$R_{31} = s_6(c_4(c_{23} - s_{23}) - s_4(c_2s_3 + c_3s_2)) + c_5c_6(c_4(c_2s_3 + c_3s_2) + s_4(c_{23} - s_{23}))$$

$$R_{32} = c_6(c_4(c_{23} - s_{23}) - s_4(c_2s_3 + c_3s_2)) - c_5s_6(c_4(c_2s_3 + c_3s_2) + s_4(c_{23} - s_{23}))$$

$$R_{33} = -s_5(c_4(c_2s_3 + c_3s_2) + s_4(c_{23} - s_{23}))$$

And,

$$p_x = d_4s_1 + d_6(c_5s_1 - s_5(c_4(c_{123} - c_1s_{23}) - s_4(c_{123} + c_{13}s_2))) + d_5(c_4(c_{12}sl + c_{13}s_2) + s_4(c_{123} - c_1s_{23})) + a_2c_{12} - a_3c_1s_{23} + a_3c_{123}$$

$$p_y = d_5(c_4(c_2s_{13} + c_3s_{12}) + s_4(c_{23}s_1 - s_{123})) - c_1d_4 - d_6(c_{15} + s_5(c_4(c_{23}s_1 - s_{123}) - s_4(c_2s_{13} + c_3s_{12}))) + a_2c_2s_1 - a_3s_{123} + a_3c_{23}s_1$$

$$p_z = d_1 + a_2s_2 - d_5(c_4(c_{23} - s_{23}) - s_4(c_2s_3 + c_3s_2)) + a_3c_2s_3 + a_3c_3s_2 - d_6s_5(c_4(c_2s_3 + c_3s_2) + s_4(c_{23} - s_{23}))$$

b) Inverse Kinematics

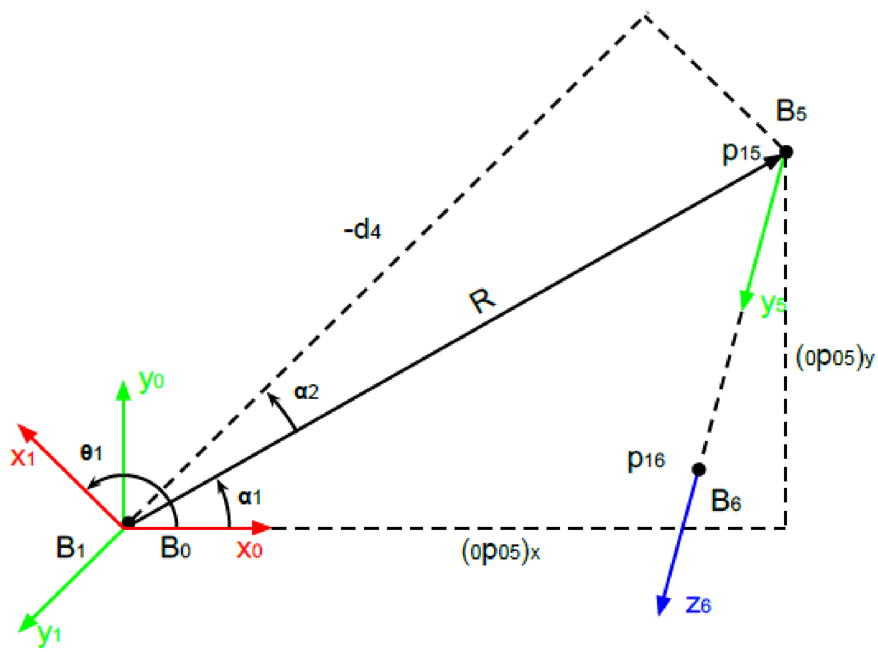
Inverse kinematics problem is to find set of joint configurations such as to achieve the desired position of the tip/tool.

i.e. $Q = \{q_i\}$ where, $q_i = (\theta_1^i, \theta_2^i, \theta_3^i, \theta_4^i, \theta_5^i, \theta_5^i) \in (0,2\pi)^6$

which will satisfy the transformation matrices obtained earlier

i.e. $T_6^0 = \begin{bmatrix} R_{11} & R_{12} & R_{13} & p_x \\ R_{21} & R_{22} & R_{23} & p_y \\ R_{31} & R_{32} & R_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$

For finding θ_1, Consider the arm in top-view X-Y plane.



$$T_5^1 = T_5^1$$

$$\{(T_1^0)^{-1} T_6^0 (T_6^5)^{-1}\}_{LHS} = \{T_2^1 T_3^2 T_4^3 T_5^4\}_{RHS}$$

Referring to the X-Y plane diagram, y-coordinate of the position of this frame is,

$$\{p_{15}^{1}\}_{LHS} = (p_x - d_6 z_x)(-s_1) + (p_y - d_6 z_y)(c_1)$$

from transformation matrices,

$\{p_{15}^{1}\}_{\mathrm{LHS}} = -d4$

i.e. $(p_x - d_6 z_x)(-s_1) + \left(p_y - d_6 z_y\right)(c_1) = -d4$

solution to the above equation can be found using,

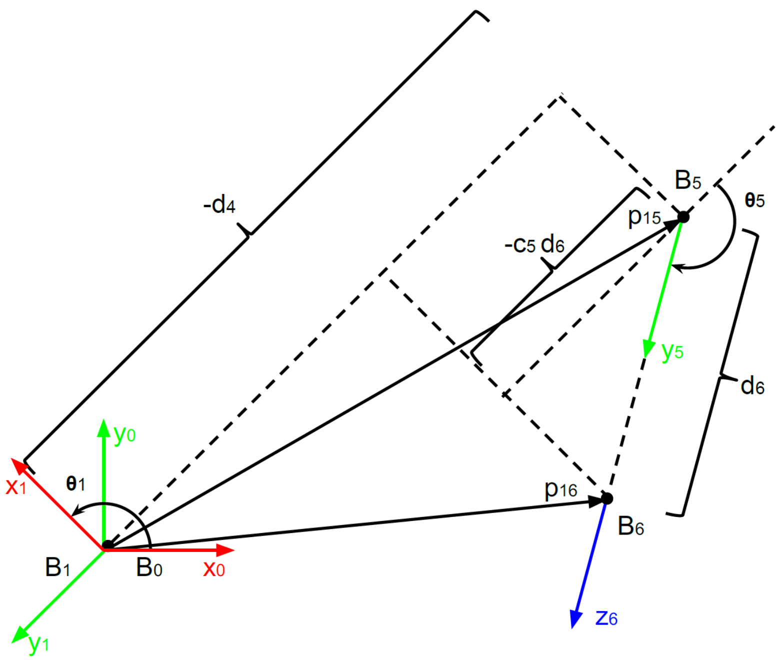
$$tan\, \alpha_1 = \frac{({}_0p_{o5})_y}{({}_op_{05})_x}$$

$$cos\, \alpha_2 = \frac{d_4}{\sqrt{({}_op_{o5})_x^2 + ({}_op_{o5})_y^2}}$$

$$\theta_1 = \alpha_1 + \alpha_2 + \pi/2$$

$$= a\, tan\, 2\, (({}_0p_{05})_y, ({}_0p_{05})_x) \pm cos^{-1}\frac{d_4}{R} + \pi/2$$

Use of atan2 is essential because, for any valid inverse kinematic solution there can be two shoulder joint configurations," left" or "right". Atan2 ensures the right direction.



Given a particular θ_1, we can solve for θ_5 using transformation from frame 1 to frame 6,

i.e. $\{(T_1^0)^{-1}T_6^0\}_{LHS} = \{T_2^1T_3^2T_4^3T_5^4T_6^5\}_{RHS}$

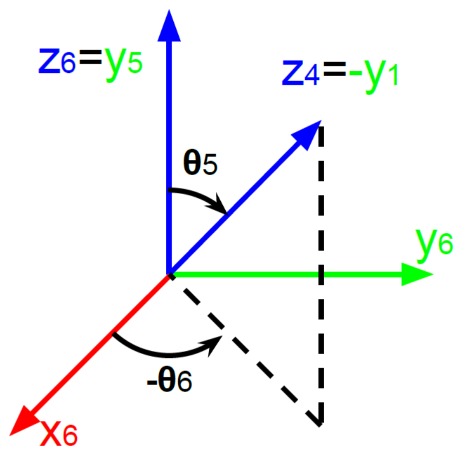
y-coordinate of the position of this frame is,

$[p_{16}^1]_{LHS} = p_x(-s_1) + p_y(c_1)$

$[p_{16}^1]_{RHS} = -d_4 + c_5d_6$

$$\theta_5 = \pm\, cos^{-1}\frac{p_xs_1 - p_yc_1 - d_4}{d_6}$$

Again, there are two solutions for θ_5 , which similarly correspond to "up" or "down" wrist position. This occurs due to the fact that the joint sum θ_{234}can allow T_5^1 to achieve orientations where y_5^1 is pointing in the same direction, but that $z_5^1$is pointing in the opposite direction. This flip can then be reversed very simply by the 6th joint. This joint has a solution so long as the argument of $\cos^{-1}$ has magnitude not greater than 1.



For θ_6, looking at the y_6^1 axes,

$$[y_6^1]_{LHS} = \begin{bmatrix} -x_x s_1 + x_y c_1 \\ -y_x s_1 + y_y c_1 \\ -z_x s_1 + z_y c_1 \end{bmatrix}$$

$$[y_6^1]_{RHS} = \begin{bmatrix} -c_6 s_5 \\ s_6 s_5 \\ -c_5 \end{bmatrix}$$

Which gives,

$$\theta_6 = a\tan 2\,\left(\frac{-y_x s_1 + y_y c_1}{s_5}, \frac{-(-x_x s_1 + x_y c_1)}{s_5}\right)$$

This solution does not hold true when either numerators are 0, or $s_5 = 0$. Similarly, $\theta_2, \theta_3, \theta_4$ were found which had multiple non-exhaustive IK solutions for a given desired location and hence computationally not feasible on paper.

c) Individual Joint Torque Calculation

Jacobian matrices are widely used in robotics. It basically defines the dynamic relation between two different representations of a system. i.e. it forms a relationship between the end-effector position and orientation with the joint angles.

The Jacobian for this system relates how movement of the elements of J_q causes movement of the elements of the arm. Hence it is also known as a transform matrix for velocity. The Jacobian for UR-10 is given as,

$$J = \begin{bmatrix} J_{v1} & J_{v2} & J_{v3} & J_{v4} & J_{v5} & J_{v6} \\ R_0^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & R_1^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & R_2^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & R_3^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & R_4^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & R_5^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{bmatrix}$$

Where, '*' represents the cross product and $J_{v1}, J_{v2}, J_{v3}, J_{v4}, J_{v5}, J_{v6}$ are as follows:

$$J_{v1} = \left[R_0^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} * (o_6^0 - o_0^0) \right] \quad J_{v2} = \left[R_1^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} * (o_6^0 - o_1^0) \right]$$

$$J_{v3} = \left[R_2^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} * (o_6^0 - o_2^0) \right] \quad J_{v4} = \left[R_3^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} * (o_6^0 - o_3^0) \right]$$

$$J_{v5} = \left[R_4^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} * (o_6^0 - o_4^0) \right] \quad J_{v6} = \left[R_5^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} * (o_6^0 - o_5^0) \right]$$

We know that $T = J^T F$ where T is the torque and J is the Jacobean matrix derived earlier. Thus, by giving the desired torque wrench requirement we can calculate the individual joint torques of our robotic arm.

Based on the frame assignment and DH parameters the Jacobian matrix elements were calculated as below,

$$J_{v1} = \begin{bmatrix} c_1 d_4 - a_2 c_2 s_1 + a_3 s_{123} - a_3 c_{23} s_1 \\ d_4 s_1 + a_2 c_{12} - a_3 c_1 s_{23} + a_3 c_{123} \\ 0 \end{bmatrix}$$

$$J_{v2} = \begin{bmatrix} -c_1(a_2 sl + a_3 c_2 s_3 + a_3 c_3 s_2) \\ -s_1(a_2 s_2 + a_3 c_2 s_3 + a_3 c_3 s_2) \\ c_1(d_4 s_1 + a_2 c_{12} - a_3 c_1 s_{23} + a_3 c_{123}) - s_1(c_1 d_4 - a_2 c_2 s_1 + a_3 s_{123} - a_3 c_{23} s_1) \end{bmatrix}$$

$$J_{v3} = \begin{bmatrix} -c_1(a_3 c_2 s_3 + a_3 c_3 s_2) \\ -s_1(a_3 c_2 s_3 + a_3 c_3 s_2) \\ c_1(d_4 s_1 - a_3 c_1 s_{23} + a_3 c_{123}) - s_1(c_1 d_4 + a_3 s_{123} - a_3 c_{23} s1) \end{bmatrix}$$

$$J_{v4} = \begin{bmatrix} -c_1(a_3 c_2 s_3 + a_3 c_3 s_2) \\ -s_1(a_3 c_2 s_3 + a_3 c_3 s_2) \\ c_1(d_4 s_1 - a_3 c_1 s_{23} + a_3 c_{123}) - s_1(c_1 d_4 + a_3 s_{123} - a_3 c_{23} s_1) \end{bmatrix}$$

$$J_{v5} = \begin{bmatrix} (c_4(c_2 s_{13} + c_3 s_{12}) + s_4(c_{23} s_1 - s_{123}))(a_3 c_2 s_3 + a_3 c_3 s_2) - \\ (c_4(c_{23} - s_{23}) - s_4(c_2 s_3 + c_3 s_2))(c_1 d_4 + a_3 s_{123} - a_3 c_{23} s_1) \\ \\ -(c_4(c_{123} + c_{13} s_2) + s_4(c_{123} - c_1 s_{23}))(a_3 c_2 s_3 + a_3 c_3 s_2) - \\ (c_4(c_{23} - s_{23}) - s_4(c_2 s_3 + c_3 s_2))(d_4 s_1 - a_3 c_1 s_{23} + a_3 c_{123}) \\ \\ -(c_4(c_{12} s_3 + c_{13} s_2) + s_4(c_{123} - c_1 s_{23}))(c_1 d_4 + a_3 s_{123} - a_3 c_{23} s_1) - \\ (c_4(c_2 s_{13} + c_3 s_{12}) + s_4(c_{23} s_1 - s_{123}))(d_4 s_1 - a_3 c_1 s_{23} + a_3 c_{123}) \end{bmatrix}$$

$$J_{v6} = \begin{bmatrix} -(a_3c_2s_3 + a_3c_3s_2)(c_{15} + s_5(c_4(c_{23}s_1 - s_{123}) - s_4(c_2s_{13} + c_3s_{12}))) - \\ s_1(c_4(c_2s_3 + c_3s_2) + s_4(c_{23} - s_{23}))(c_1d_4 + a_3s_{123} - a_3c_{23}s_1) \\ \\ -(c_5s_1 - s_5(c_4(c_{123} - c_{123}) - s_4(c_{123} + c_{13}s_2)))(a_3c_2s_3 + a_3c_3s_2) - \\ s_5(c_4(c_2s_3 + c_3s_2) + s_4(c_{23} - s_{23}))(d_4s_1 - a_3c_1s_{23} + a_3c_{123}) \\ \\ (c_{15} + s_5(c_4(c_{23}s_1 - s_{123}) - s_4(c_2s_{13} + c_3s_{12})))(d_4s_1 - a_3c_1s_{23} + a_3c_{123}) - \\ (c_5s_1 - s_5(c_4(c_{123} - c_1s_{23}) - s_4(c_{12}s_3 + c_{13}s_2)))(c_1d_4 + a_3s_{123} - a_3c_{23}s_1) \end{bmatrix}$$

$$J_{w1} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad J_{w2} = \begin{bmatrix} s_1 \\ -c_1 \\ 0 \end{bmatrix} \qquad J_{w3} = \begin{bmatrix} s_1 \\ -c_1 \\ 0 \end{bmatrix}$$

$$J_4 = \begin{bmatrix} s_1 \\ -c_1 \\ 0 \end{bmatrix}$$

$$J_{w5} = \begin{bmatrix} c_4(c_{12}s_3 + c_{13}s_2) + s_4(c_{123} - c_1s_{23}) \\ c_4(c_2s_{13} + c_3s_{12}) + s_4(c_{23}s_1 - s_{123}) \\ s_4(c_2s_3 + c_3s_2) - c_4(c_{23} - s_{23}) \end{bmatrix}$$

$$J_{w6} = \begin{bmatrix} c_5s_1 - s_5(c_4(c_{123} - c_1s_{23}) - s_4(c_{12}s_3 + c_{13}s_2)) \\ -c_{15} - s_5(c_4(c_{23}s_1 - s_{123}) - s_4(c_2s_{13} + c_3s_{12})) \\ -s_5(c_4(c_2s_3 + c_3s_2) + s_4(c_{23} - s_{23})) \end{bmatrix}$$

V. Model Validation and Simulation

The forward kinematics and inverse kinematics equations are determined theoretically and practically for verification and the required motion is pre-planned by feeding the input the robot controller. We have used CoppeliaSim (previously VRep) integrated with MATLAB to perform extensive calculations and achieve the proposed application designed around UR-10.

a) Simulation Environment-

CoppeliaSim is a robot simulator with integrated development environment, which is based on a distributed control architecture: each object/model can be individually controlled via an embedded script, a plugin, a ROS, a remote API client, or a custom solution. CoppeliaSim is versatile and ideal for multi-robot applications. Controllers can be written in C/C++, Python, Java, Lua, Matlab or Octave. CoppeliaSim is provides a platform for fast algorithm development, factory automation simulations, fast prototyping and verification, robotics related education, remote monitoring, safety double-checking, as digital twin.

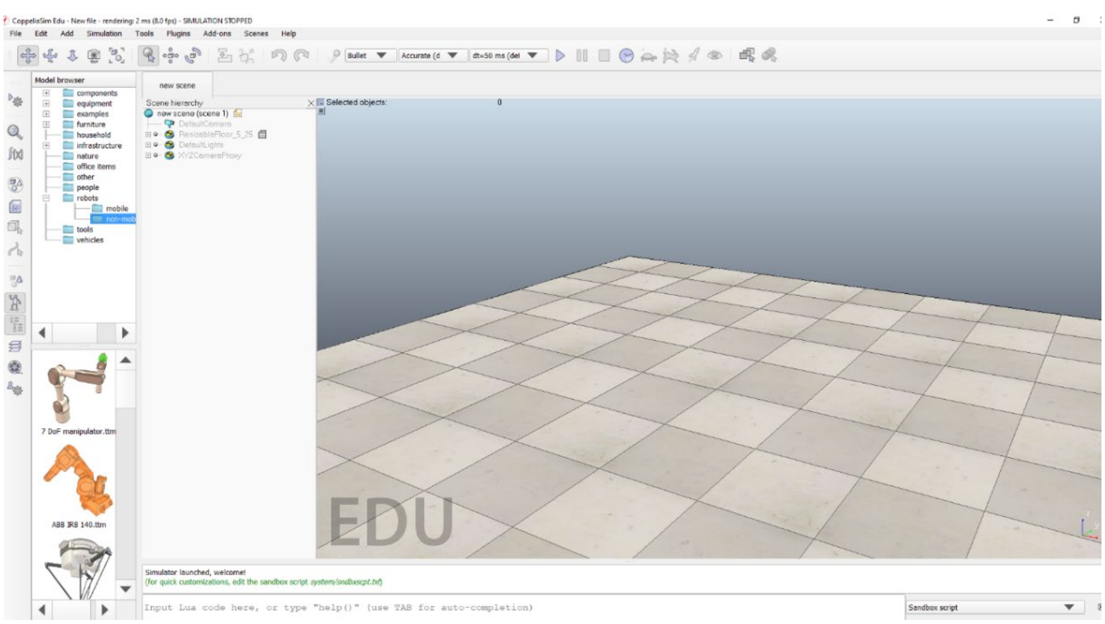


Fig. 8: CoppeliaSim GUI

b) Proposed scene recreation-

We recreated the proposed palletizing operation in the simulation environment.

The Primary Components in the scene are-

- Conveyor Belt (to keep the boxes moving)
- A proximity sensor (to detect the presence of a box at predefined location)
- Pallet (Standard size= 48in X 4in)
- UR-10 6-DOF arm (Mounted on a fixed table but may be mounted on a mobile platform)
- Box (For validation, size 0.1mmx0.1mmx01mm, weight-200gm)

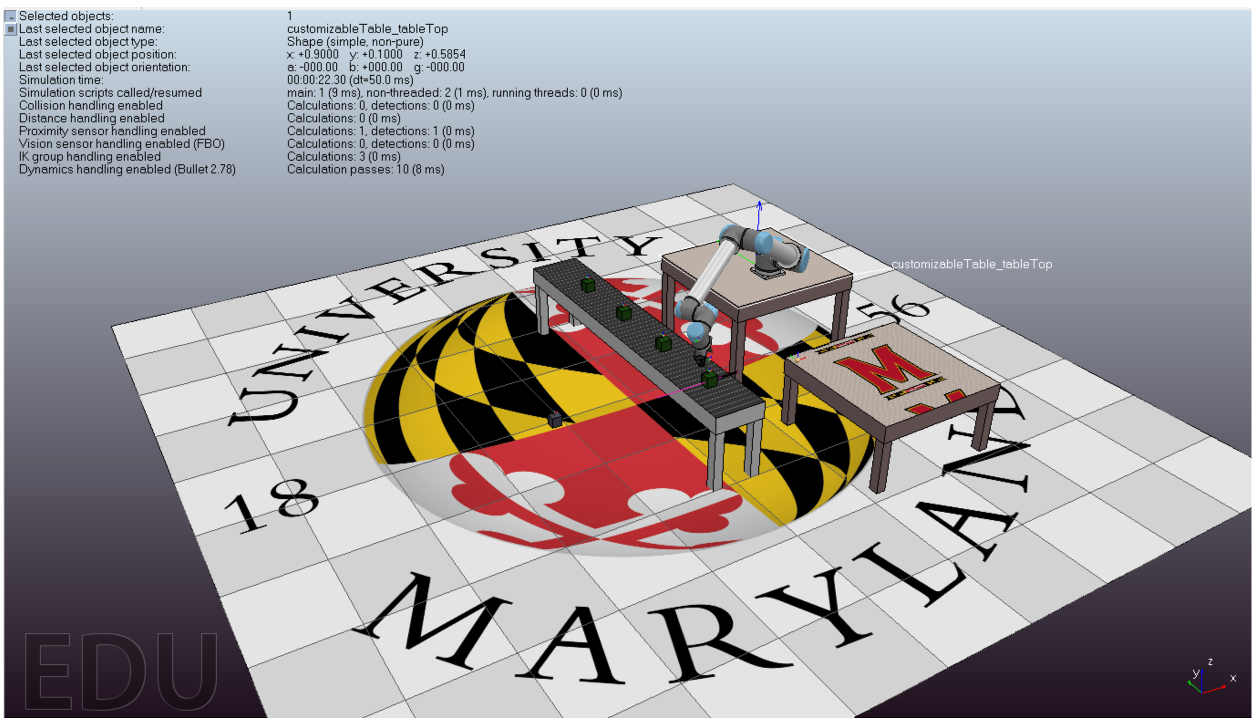


Fig. 9: Proposed Palletizing Operation

Recreated in CoppeliaSim

c) Workflow of the proposed system-

1) The designed workflow is as follows-
2) The robotic arm (UR-10) initializes its joints to an optimized position for the application. It is also set in the Inverse Kinematics mode in simulation environment.
3) The conveyor is initialized with the assumption of least friction and high efficiency.
4) The cube size(0.1mx0.1mx0.1m) boxes are dropped on the conveyor. Assumption: The boxes are pre-oriented and uniform size.
5) The proximity sensor detects the presence of the box at the position installed. A NC type ray proximity sensor is used for validation. However, it can easily be interchanged without affecting the system.
6) The inverse kinematics to the position of the box are calculated and the arm moves to picking position.
7) The ROBOTIQ-85 gripper opens and picks the box
8) The script calculates the drop location based on previous drop location/ in case of first box it is dropped at the origin of the pallet
9) Go to step 1 or break if pallet is full.

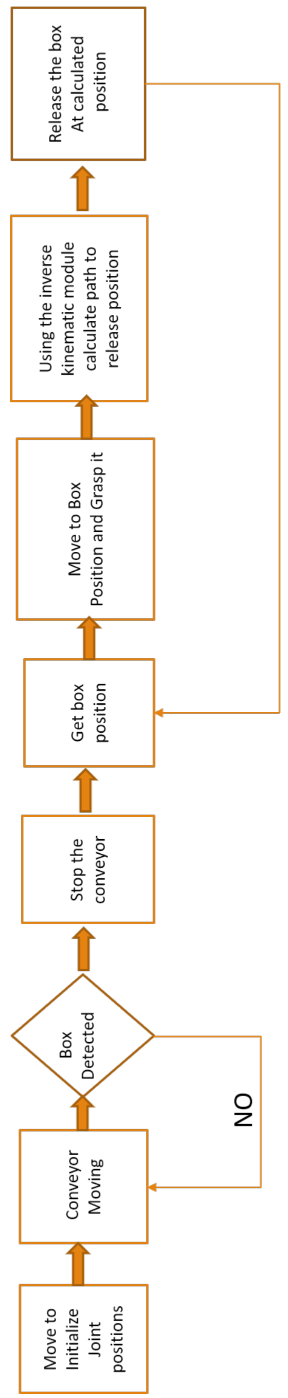


Fig. 10: Workflow of the system

d) Validation-

Calculations of Forward and Inverse kinematics were theoretically possible, but manually simulating the inverse kinematic position resulted in unexpected behavior. The integrated inverse kinematics module had issues like, failure to detect environmental obstacles. Grasping the object was a task we could not implement as the arm kept colliding with work object.

Validation Video link

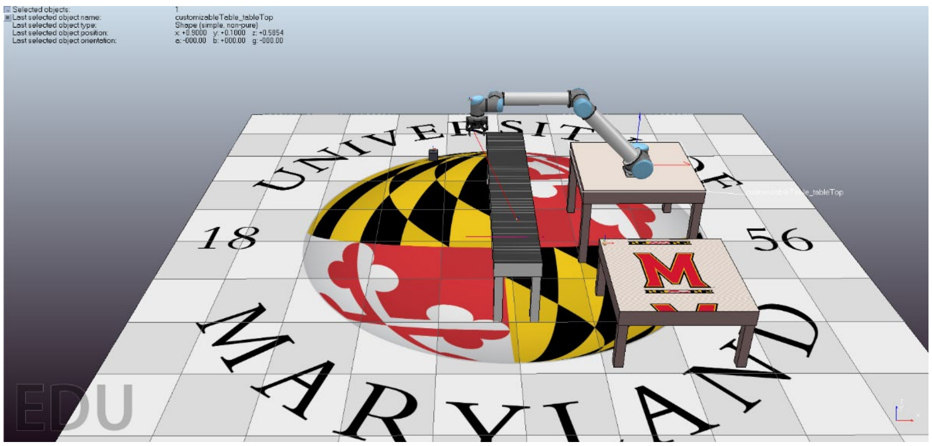


Fig. 11: Initial joint configuration

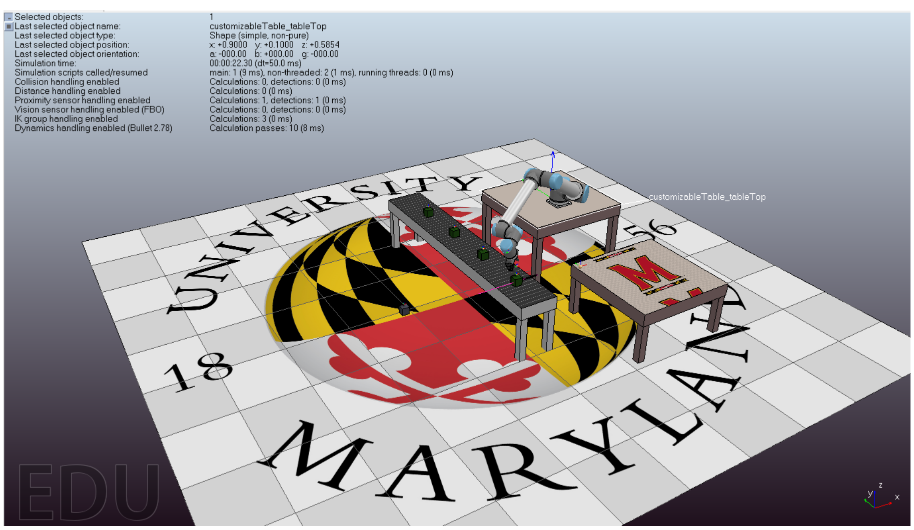


Fig. 12: Grasping Position

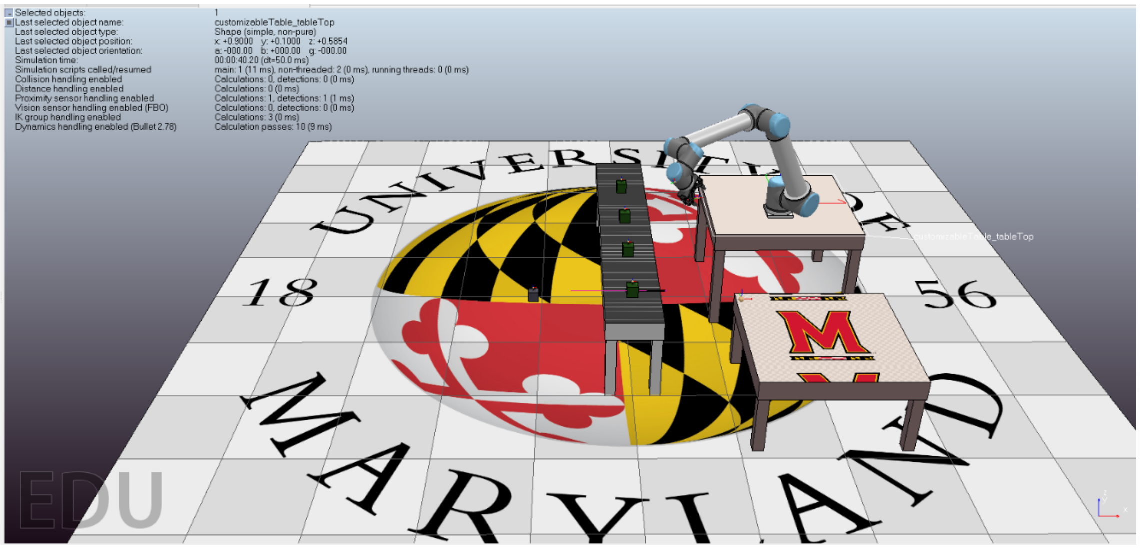


Fig. 13: Object Travel

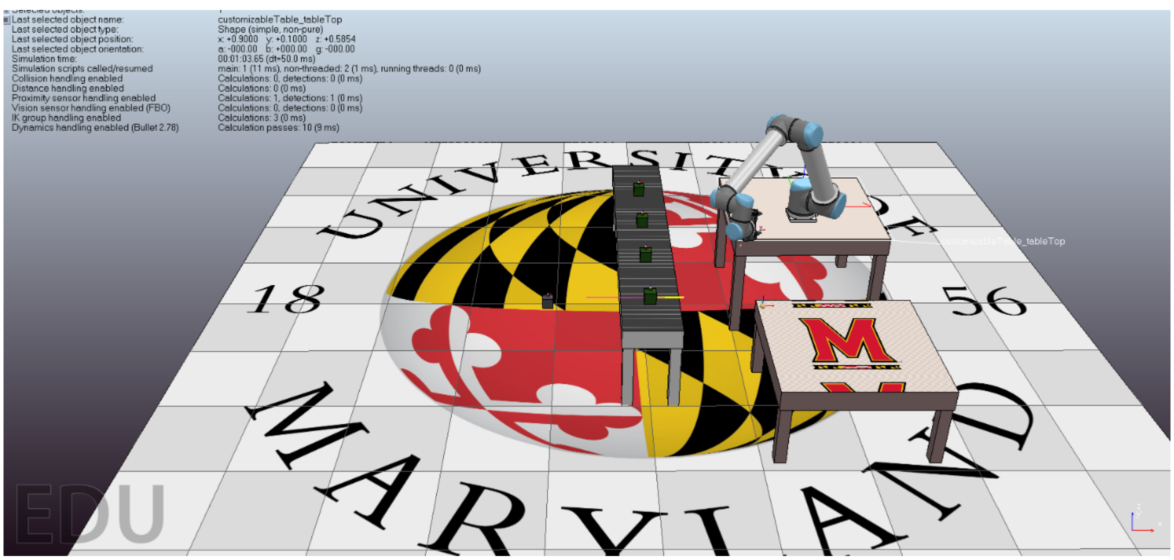


Fig. 14: Object Travel

VI. Limitations and Future Scope

- The path is manually calculated for the scope of the project, efficient path planning algorithms could be implemented.
- The grasping is done using a mechanical movement gripper, which is both not an optimized solution and impractical in supply chain operations.
- Co-ordinated robotic arms could be implemented, limiting functions of one to pick and other to place, maximizing the output.
- Environmental obstacles detection would be an essential feature in mobile applications.
- The boxes are assumed to be aligned, which may not be a real-life scenario, reorienting the boxes would be one of the necessary improvements.

Conclusion-

My approach to kinematics of a 6-DOF robotics arm is based on standard principled methods for studying robot kinematic chains. The project was demanding and difficult to complete as it involved many unseen aspects. However, that resulted in the learning and derivations of Inverse position kinematics and Forward position kinematics of the UR-10 as the major and the primary outcome. I conducted an extensive literature survey on the existing methods of solving the inverse kinematics for 6-DOF redundant manipulators. The forward kinematic equations were derived using the Denavit-Hartenberg convention. But apart from that it also resulted in many sub outcomes, familiarizing me with current industry standards, common issues involved with real world interfaces, design-implementation trade-offs, which are also important for the robotics system implementation. The project outcomes were seen in the derivation results and the videos of simulations.

References

1. *Robot Dynamics and Control*: Mark W. Spong, M. Vidyasagar.
2. https://www.universal-robots.com/products/ur10-robot/
3. http://www.coppeliarobotics.com/coppeliaSim
4. https://www.youtube.com/channel/UCX6c67E5PTUq5W8n8POeO_w
5. J. Zou and D. Tu, "The development of six D.O.F. robot arm for intelligent robot," *2011 8th Asian Control Conference (ASCC)*, Kaohsiung, 2011, pp. 976-981.
6. Q. Liu, D. Yang, W. Hao and Y. Wei, "Research on Kinematic Modeling and Analysis Methods of UR Robot," *2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC)*, Chongqing, China, 2018, pp. 159-164.