ACADEMIC CITY
UNIVERSITY COLLEGE

MOdelling and Simulation of Mechatronic Systems Semester Exam

Academic City University College

Department of Electrical and Computer Engineering

Modelling and Simulation of 5-DOF Robotic Arm

Author:
Nathaniel Awontiirim Asiak

Supervisor:
Michael Mensah

May 6, 2022

Contents

List of Tables

1 Introduction

Automation is one of the most crucial aspects for development in the industrial world. It aids in the reduction of human dependency while also increasing efficiency and production. Automation is used to produce a variety of sophisticated equipment that is utilized on a daily basis, including medical equipment (x-ray machines, radiography, etc.), refrigerators, autos, and so on. Among all of these outputs, one of the most commonly employed in industrial applications is the robotic arm.

A robotic arm is a type of mechanical arm that performs functions comparable to those of a human arm and is usually programmable. The arm can be the sum of the mechanism or part of a larger robot. The manipulator's linkages are joined by joints that allow for either rotational (as in an articulated robot) or translational (linear) movement. The manipulator's linkages can be thought of as a kinematic chain. The end effector, which is equivalent to the human hand, is the end of the manipulator's kinematic chain. Depending on the application, the end effector, or robotic hand, can be built to execute any required operation, such as welding, grabbing, spinning, and so on.

An electric driver (motor) or a pneumatic and hydraulic system are commonly used to drive this arm movement (pistons). These actuators are controlled by a microcontroller (CPU), which is usually programmable and designed to carry out a series of operations in a specific order. The majority of these robotic arms are intended for industrial application, with rapid and reliable performance to aid mass production.

A scientist named George Devol, Jr. created the first robotic arm in the 1950s, before which robotics were mostly the stuff of science fiction and the imagination. For a long time, robotics progressed slowly, with many of the most useful applications involving space exploration [1]. Robotic arms were not completely realized as an aid to industrialization until the 1980s, when they were integrated into automobile and other manufacturing assembly lines.

1.1 Objectives

This project is theoretical based. It discusses the theory of operation of the robotic arm, and additional modelling and simulation of the robotic arm to understand the robotic arm as a mechatronic system. As such, the project includes the following objectives :

1. Model the robotic arm in CAD modelling software (Onshape)

2. Design and simulate the robotic arm in Simscape

3. Analyse the performance of the end-effector

2 Literature Review and Related Work

This chapter discusses previous work and important theories that are relevant to understanding the operation of the robotic arm with 5 degrees of freedom.

2.1 Related Work

Today, a variety of robotic arms are employed in robotics research, each with its own set of capabilities and design criteria. In this part, we'll look at some of the most popular and/or significant robotic arms in recent years.

The Barrett WAM is a cable-driven robot known for its high backdrivability and smooth, fast operation. It has high-speed (3 m/s) operation and 2 mm repeatability.

The Meka A2 arm is series-elastic, intended for human interaction; other, custom-made robots with series-elastic arms include Cog, Domo, Obrero, Twendy-One, and the Agile Arm. The Meka arm and Twendy-One use harmonic drive gearheads, while Cog uses planetary gearboxes, and Domo, Obrero, and the Agile Arm use ball screws; the robots all use different mechanisms for their series elasticity. These arms have lower control bandwidth (less than 5 Hz) due to series compliance, yet that has not appeared to restrict their use in manipulation research.

The PR2 robot developed at Stanford University has a unique system that uses a passive gravity compensation mechanism, so the arms float in any configuration. Because the large mass of the arm is already supported, relatively small motors are used to move the arms and support payloads. These small motors provide human safety, as they can be backdriven easily due to their low gear ratios.

The commercially available robotic arms listed before are all relatively expensive, with end-user purchase prices far exceeding $100,000. However, there are a few low-cost robotic manipulators that have been employed in research. Below are a few of such robotic manipulators.

The arms on the Dynamaid robot are constructed from Robotis Dynamixel robotics servos, which are light and compact. The robot has a human-scale workspace, but a lower payload (1 kg) than the class of arms discussed previously. Its total cost is at least

$3500 USD, which is the price of just the Dynamixel servos.

The KUKA youBot arm is a new 5-DOF arm for robotics research. It has a comparatively small work envelope of just over 0:5 m3, repeatability of 0.1 mm, and a payload of 0.5 kg.

2.2 Literature Review

The operation of a robotic depends on two important concepts in physics; Kinematics and Dynamics. This section discusses the two concepts as they apply to robotic arm manipulators

2.2.1 Kinematics

Kinematics is the analysis of motion without considering forces. Here, we only need geometric properties such as lengths and degrees of freedom of the manipulator bodies. Robot manipulators are frequently made up of many joints. The degrees of freedom in joints are either revolute (rotating) or prismatic (linear) (DOF). As a result, joint positions may be manipulated to place the robot's end effector in 3D space [2].

Forward kinematics is used when you can do the math and find out the position and orientation of any point on the robot if you know the robot's geometry and all of its joint positions. The more common robot manipulation issue, on the other hand, is the inverse. In this type of kinematics, it is important to figure out how to calculate the joint angles required to get the end effector to a given position and orientation. Inverse kinematics is a more challenging problem to solve [2].

It is important to obtain the kinematic diagram of a robotic system before either of the kinematic analysis can be used on it. Figure 2.1 shows how the joints and links are connected in a 5-DOF robotic arm manipulator.

2.2.1.1 Mathematicl modeling of kinematics To define the kinematics of the manipulator axes need to be placed according to the Denavit-Hartenberg method. The basic rules for using this convention are:

- x-axis is parallel to the common normal.
- z-axis is in the direction of the joint axis

The Denavit-Hartenberg parameters corresponded to the mounting configuration of links and determent based on the following rules:

- Rotation around z_{i-1} axis an angle θ_i

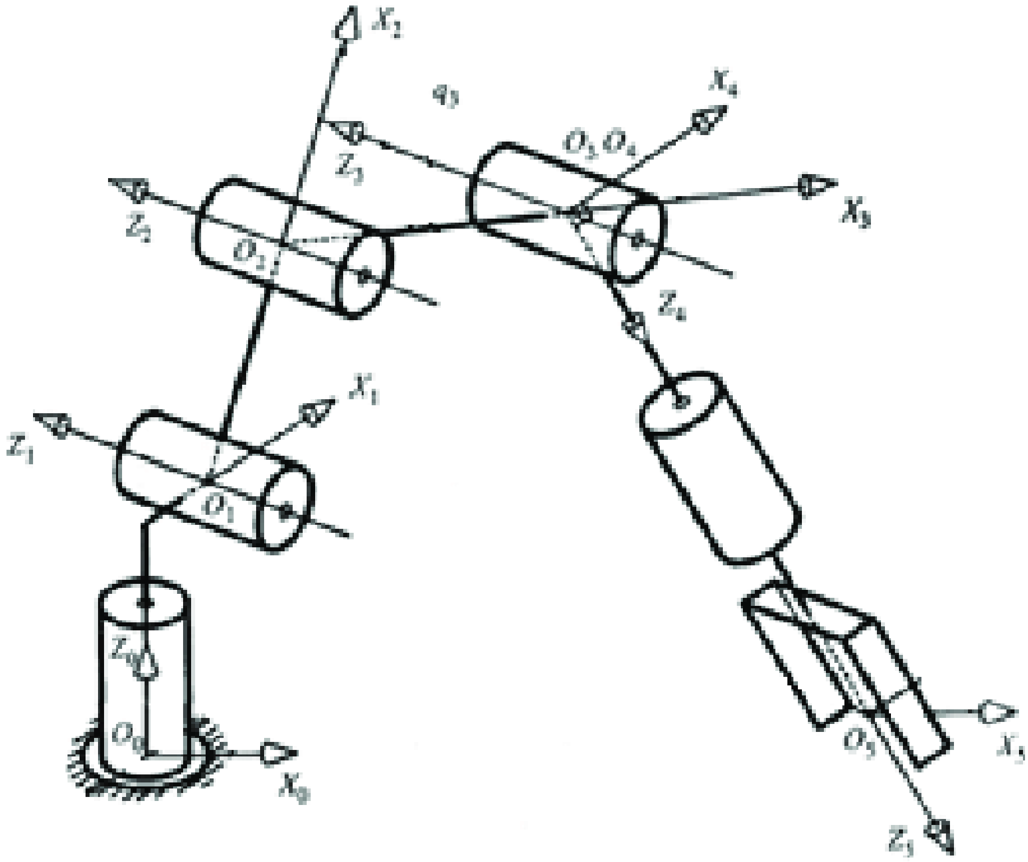


Figure 2.1: Kinematic Scheme of 5-DOF manipulator

- Translation through z_i axis a distance d_i
- Translation through x_i axis a distance a_i.
- Rotation around x_i axis an angle α_i

Table 2.1: D-H symbolic paramters for Kuka youBot Arm

Link	α_i	a_i	d_i	θ_i
1	α_1	a_1	d_2	θ_1
2	α_2	a_2	d_2	θ_2
3	α_3	a_3	d_3	θ_3
4	α_4	a_4	d_4	θ_4
5	α_5	a_5	d_5	θ_5

These parameters can be used to describe the relative position and orientation of links by transition matrixes:

$$A_i = \begin{vmatrix} c_i & -c_{\alpha_i} \cdot s_i & s_{\alpha_i} \cdot s_i & a_i \cdot c_i \\ s_i & c_{\alpha_i} \cdot c_i & -s_{\alpha_i} \cdot c_i & a_i \cdot s_i \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{vmatrix}, \text{ where } c_i = \cos q_i, \quad s_i = \sin q_i, \quad c_{\alpha_i} = \cos \alpha_i, \quad s_{\alpha_i} = \sin \alpha_i.$$

For the Denavit-Hartenberg parameters of the 5 DOF manipulator:

$$A_1 = \begin{vmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{vmatrix} \tag{2.1}$$

$$A_2 = \begin{vmatrix} c_2 & -s_2 & 0 & a_2 \cdot c_2 \\ s_2 & c_2 & 0 & a_2 \cdot s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}, \tag{2.2}$$

$$A_3 = \begin{vmatrix} c_3 & -S_3 & 0 & a_3 \cdot c_3 \\ s_3 & c_3 & 0 & a_3 \cdot s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}, \tag{2.3}$$

$$A_4 = \begin{vmatrix} c_4 & 0 & -s_4 & 0 \\ s_2 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}, \tag{2.4}$$

$$A_5 = \begin{vmatrix} c_5 & -S_5 & 0 & 0 \\ S_5 & c_5 & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ 0 & 0 & 0 & 1 \end{vmatrix} \tag{2.5}$$

Calculation of the homogeneous transformation matrix T_5 that connects the system of end-effector with the base frame of the system is given by;

$$T_5 = A_1 \cdot A_2 \cdot A_3 \cdot A_4 \cdot A_5; \tag{2.6}$$

$$T_5 = \begin{vmatrix} c_1 \cdot c_{234} \cdot c_5 + s_1 \cdot s_5 & -c_1 \cdot c_{234} \cdot s_5 + s_1 \cdot c_5 & -c_1 \cdot s_{234} & c_1 \cdot (-d_5 \cdot s_{234} + a_3 \cdot c_{23} + a_2 \cdot c_2) \\ c_1 \cdot c_{234} \cdot c_5 - s_1 \cdot s_5 & -S_1 \cdot c_{234} \cdot s_5 - c_1 \cdot c_5 & -S_1 \cdot s_{234} & s_1 \cdot (-d_5 \cdot s_{234} + a_3 \cdot c_{23} + a_2 \cdot c_2) \\ -S_{234} \cdot c_5 & s_{234} \cdot s_5 & -c_{234} & d_1 - a_2 \cdot s_2 - a_3 \cdot s_{23} - d_5 \cdot c_{234}) \\ 0 & 0 & 0 & 1 \end{vmatrix}, \tag{2.7}$$

where $c_{ijk} = \cos(q_i + q_j + q_k)$, $\quad s_{ijk} = \sin(q_i + q_j + q_k)$.

Gripper's position is defined by P_5 vector while orientation defined by X_5, Y_5, Z_5 vectors provided by the forward kinematics.

$$p_5 = \begin{pmatrix} c_1 \cdot (-d_5 \cdot s_{234} + a_3 \cdot c_{23} + a_2 \cdot c_2) \\ s_1 \cdot (-d_5 \cdot s_{234} + a_3 \cdot c_{23} + a_2 \cdot c_2) \\ d_1 - a_2 \cdot s_2 - a_3 \cdot c_{23} - d_5 \cdot c_{234} \end{pmatrix} \tag{2.8}$$

$$x_5 = \begin{pmatrix} c_1 \cdot c_{234} \cdot c_5 + s_1 \cdot S_5 \\ S_1 \cdot c_{234} \cdot c_5 + c_1 \cdot s_5 \\ -S_{234} \cdot c_5 \end{pmatrix} \tag{2.9}$$

$$y_5 = \begin{pmatrix} c_1 \cdot c_{234} \cdot S_5 + s_1 \cdot c_5 \\ -S_1 \cdot c_{234} \cdot S_5 - s_1 \cdot c_5 \\ S_{234} \cdot S_5 \end{pmatrix} \tag{2.10}$$

$$z_5 = \begin{pmatrix} -c_1 \cdot S_{234} \\ -S_1 \cdot S_{234} \\ c_{234} \end{pmatrix} \tag{2.11}$$

The solution to the inverse kinematics problem is reduced to a search of the arguments θ_1, θ_2, θ_3, θ_4, θ_5 based on the gripper's position and orientation. The six equations representing the gripper's position are in five unknowns and may have no solution, though it is possible to consider it for some instances [3].

Based on equations 8-11, the solutions for the angles of the joints can be solved as:

$$q_1 = \tan^{-1} \frac{p_{5y}}{p_{5x}} \tag{2.12}$$

$$q_2 = \operatorname{atan2}\left(a \cdot (a_2 + a_3 \cdot c_3) - b \cdot a_3 \cdot s_3, a \cdot a_3 \cdot s_3 + b \cdot (a_2 + a_3 \cdot c_3)\right) \tag{2.13}$$

$$q_3 = \arccos \frac{a^2 + b^2 - a_2^2 - a_3^2}{2 \cdot a_2 \cdot a_3} \tag{2.14}$$

$$\text{where} : a = d_1 - d_5 \cdot c_{234} - p_{5z}, \quad b = p_{5x} \cdot c_1 + p_{5y} \cdot s_1 + d_5 \cdot s_{234}$$

$$q_4 = q_{234} - q_2 - q_3 \tag{2.15}$$

$$q_5 = c_{234} \cdot q_1 - 2\operatorname{atan}\left(x_{5y}, x_{5x}\right) \tag{2.16}$$

Solutions of the forward kinematics and inverse kinematics were obtained in Matlab.

2.2.2 Inverse Dynamics

The branch of mechanics known as inverse dynamics connects the fields of kinematics and kinetics. Forces and moments of force are calculated indirectly from the kinematics and inertial properties of moving things in this procedure. Inverse dynamics can be applied to stationary bodies in theory, although it is most commonly used with moving bodies. It comes from Newton's second law, which divides the total force into known and unknown forces [4]. For many applications with fixed-based robots, the multi-body dynamics is formulated as given in equation 2.17.

In literature, different methods exist to compute the so-called Equations of Motion of a given system, i.e., a closed-form mathematical model of the system dynamics. All such methods are usually based on Newtonian and/or Lagrangian mechanics formulations, but despite the different approaches taken, all methods will result in equivalent descriptions of the dynamic.

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{J}_c(\mathbf{q})^T \mathbf{F}_c \tag{2.17}$$

where : $\mathbf{M}(\mathbf{q}) \in R^{n_q \times n_q}$ Generalized mass matrix (orthogonal).

$\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \quad \in R^{n_q}$ Generalized position, velocity and acceleration vectors.

$\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) \quad \in R^{n_q}$ Coriolis and centrifugal terms.

$\mathbf{g}(\mathbf{q}) \quad \in R^{n_q}$ Gravitational terms.

$\boldsymbol{\tau} \in R^{n_q}$ External generalized forces

$\mathbf{F}_c \in R^{n_c}$ External Cartesian forces (e.g. from contacts).

$\mathbf{J}_c(\mathbf{q}) \quad \in R^{n_c \times n_q}$ Geometric Jacobian corresponding to the external forces.

3 Methodology and Implementation

The main hypothesis of this thesis was to design a 5-DOF robotic manipulator that uses inverse kinematics and some dynamic controls to plan its trajectory. This section describes the techniques and the modeling that was done in this project.

3.1 Block Diagram

The block represents how the various parts of the system are connected to have a fully operational system. The main components are the input control (environment, and the control system), robotic manipulator, and closed feedback loop which returns the positions of the end-effector in order for the inverse kinematics to determine the appropriate joint angles to achieve the desired end-effector position. The environment consists of everything around the manipulator. In this case, it consists of the input and output conveyor belt and the load box which the robot picks up. The control logic is designed using a stateflow diagram in Simulink which allows for supervisory control and trajectory planning. The robotic arm model consists of a 5-DOF arm manipulator modeled in Onshape and imported into Simulink. The manipulator moves, picks up the box from the input belt, and drops it on the output belt.

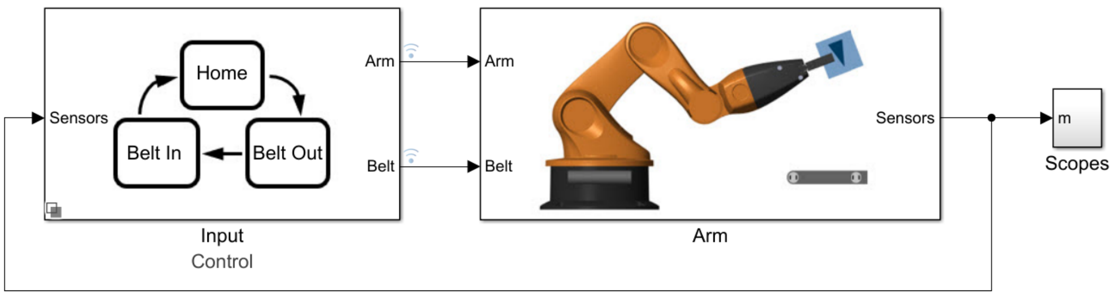


Figure 3.1: Block diagram of Robotic Arm manipulator

3.2 Environment

The environment of a robotic manipulator refers to the space around the agent where the robot accepts percepts through sensors and takes action on it. In this application, the environment is a conveyor belt that rolls a box in and out. Thus, the environment contains two conveyor belts. An input conveyor belt, and an output conveyor belt.

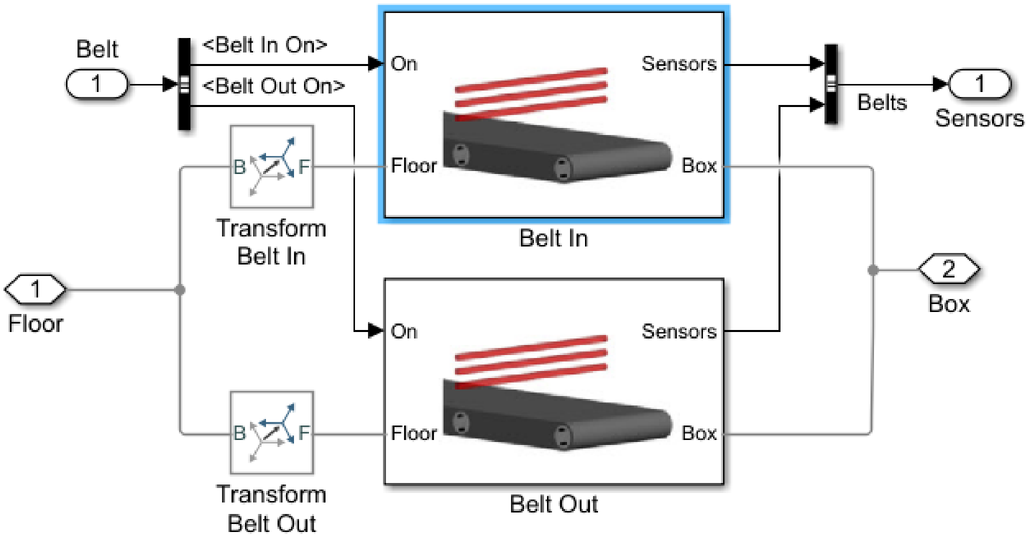
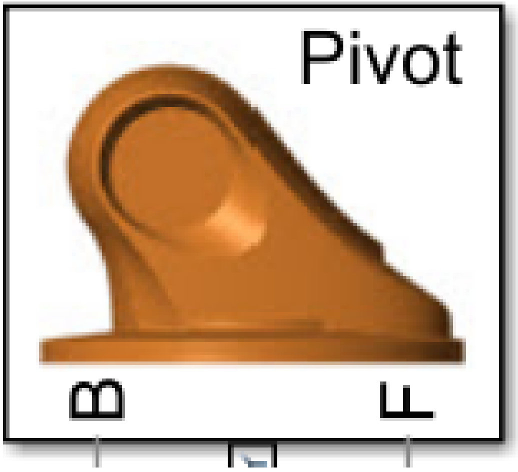


Figure 3.2: Simscape model of manipulator environment

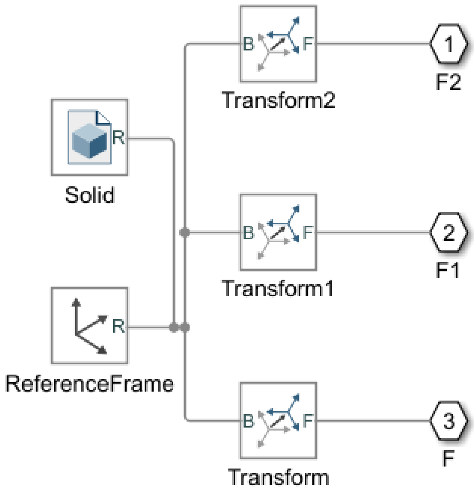
3.3 Robotic Arm Configuration

The robotic arm is a 5-DOF manipulator with 5 revolute joints and 2 prismatic joints. The revolute joints are found at the Pivot, Forearm, Bicep, Wrist, and Gripper. The prismatic joints are located on the two fingers of the gripper.

The joints are driven by five brushless Maxon motors capable of delivering up to 50W of power. These motors are responsible for the dynamics of the system.

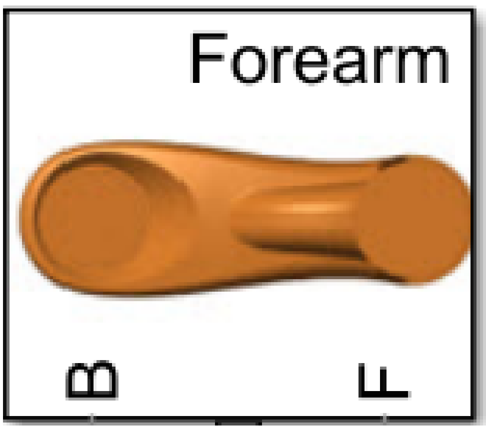


(a) Onshape Pivot Model

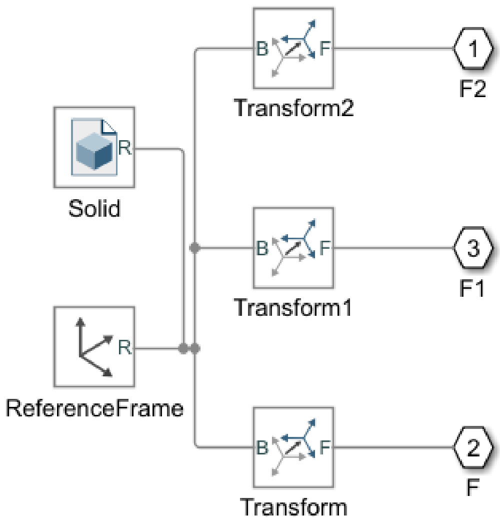


(b) Simscape connection of Pivot

Figure 3.3: Pivot Joint of the Manipulator



(a) Onshape Forearm Model



(b) Simscape connection of forearm

Figure 3.4: forearm Joint of the Manipulator

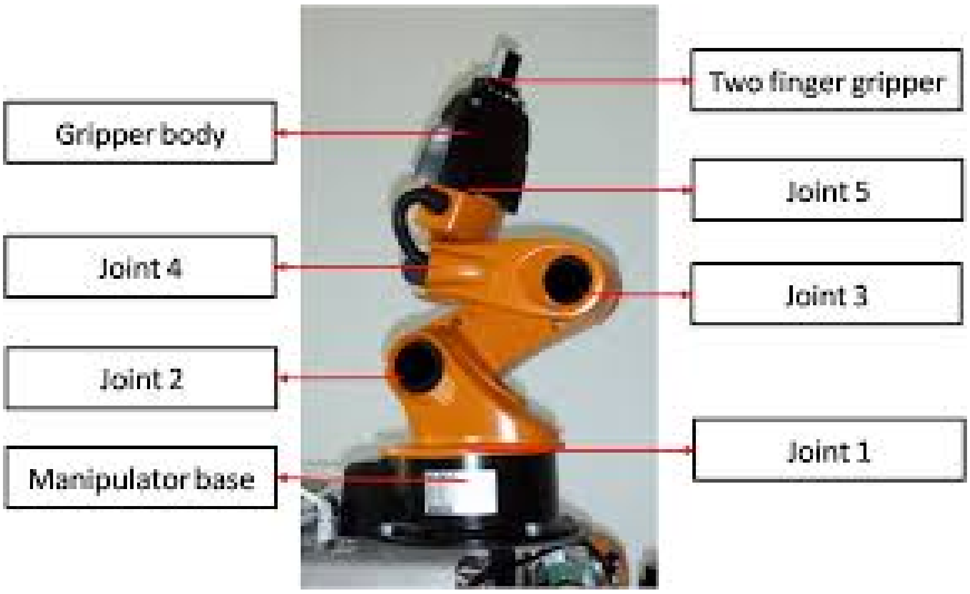


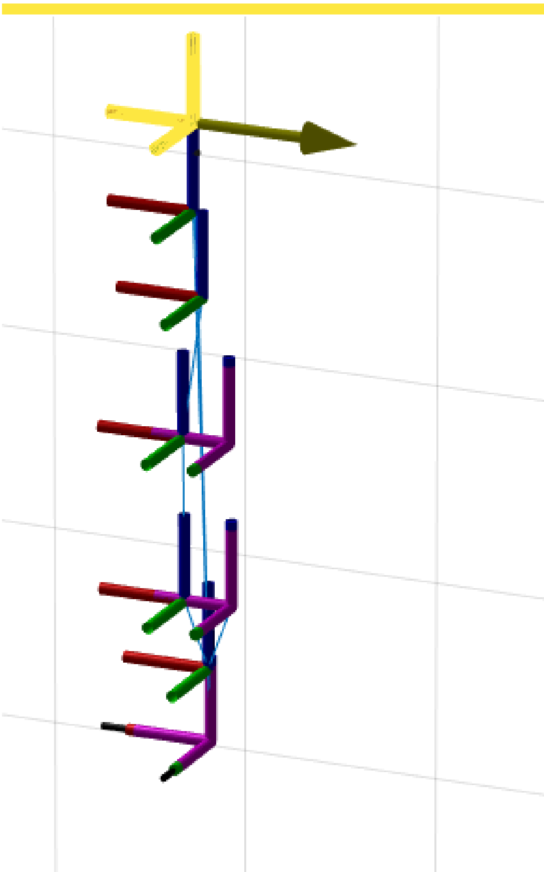
Figure 3.5: KUKA youBot with joints labeled

3.4 Control system

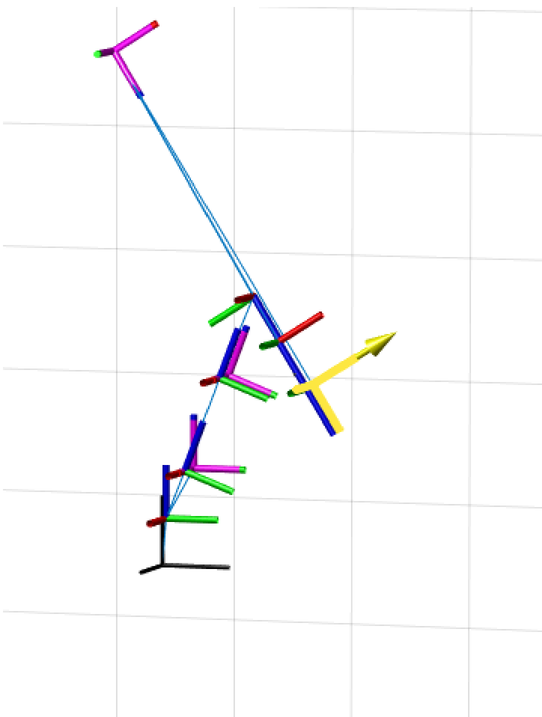
The control system is responsible for the logic that drives the motors of the manipulator to ensure the motion. The system utilizes a closed feedback loop to control the robot's operation. The feedback loop returns whether there is a load on the belts, which then triggers the direction of the robot arm to move based on the trajectory planning done using inverse kinematics. Figure 3.7 shows a snapshot of the control system described above.

It is important to note that, inverse kinematics was used to determine the trajectory of the robot given each state of the environment. Figure 3.7 shows the skeletal view of the robotic arm before inverse kinematics was applied and the position of the robotic arm after inverse kinematics was applied using the code snippet in the appendix.

In this regard, figure 3.8 shows the joint angles of the trajectory for the five revolute joints during the lifting and dropping of the box off the conveyor belts and the translational motion of the prismatic joints at the fingers of the gripper.



(a) Kinematic diagram of robotic arm



(b) Position of robotic arm after Inverse Kinematics

Figure 3.6: Joint angles using Kinematics

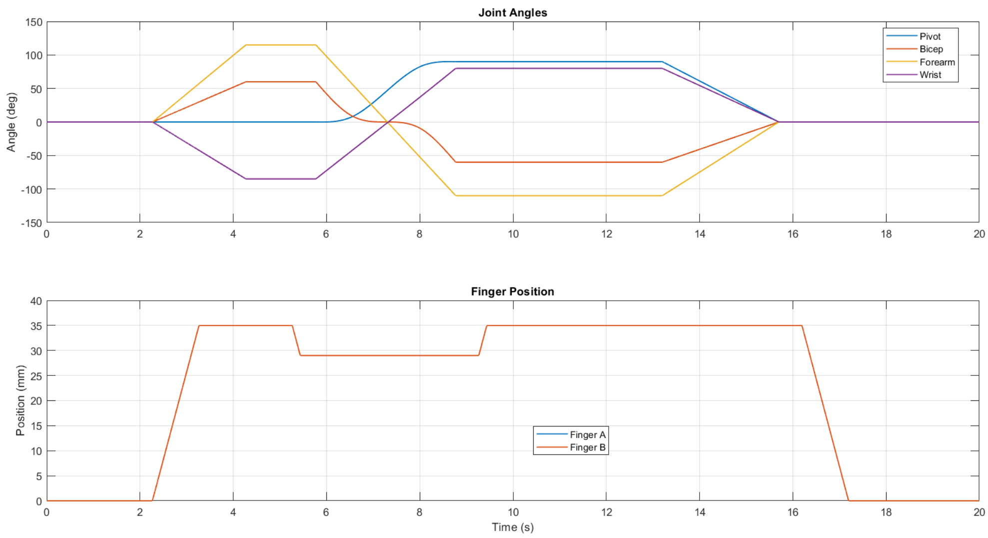


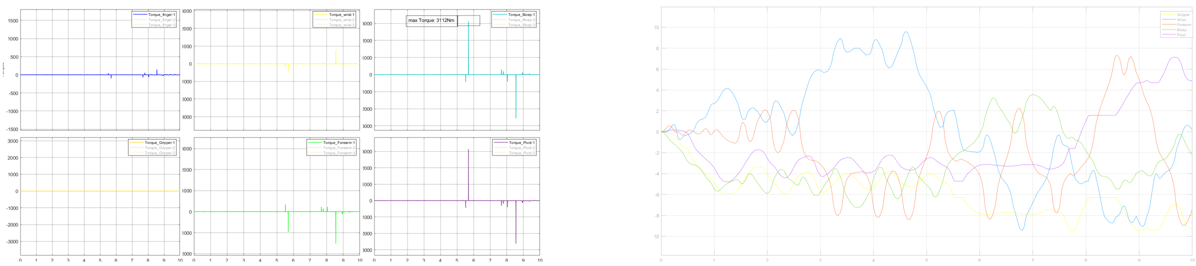
Figure 3.7: Joint angles of youRobot determining through inverse kinematics

4 Results and Discussion

This chapter discusses the results of the simulations run on the robotic arm by comparing the results of torque cost trajectory paths from the unoptimized robotic manipulator versus similar results from the optimized robotic arm.

4.1 Unoptimized Robotic Arm

The unoptimized robotic arm does not consider the dynamics due to the actuators and does not consider the trajectory that produces the optimum power consumption cost. This simulation does not follow any planned trajectory and the arm moves randomly in all directions. Figure 4.1 shows the results of the torque required to move the various arms of the manipulator as well as the positioning of the joints of the manipulator during the operation of the arm. These results show that the maximum torque of 3111Nm is required to move the Bicep arm. The total torque cost as calculated from the Simulink simulation of the five revolute joints is 1.265e+05.



(a) Torque required to move various joints during simulation

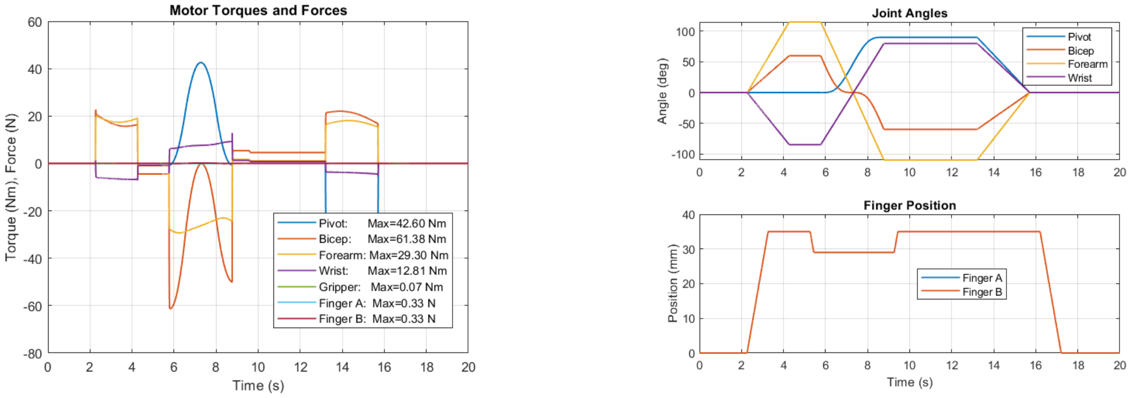
(b) Position of joints of unoptimized robotic arm during simulation

Figure 4.1: Torque and Joint angles of unoptimized robotic arm

4.2 Optimized Robotic Arm

The purpose of optimization is to find the best trajectory that leads to the minimize power consumption in the robotic arm. This essentially means finding the best path for the robotic arm to follow that would reduce the amount of work the motors have to do which means a reduction in the amount of motion(torque) and current required to drive the motors. The optimized model runs on inverse dynamics to calculate the required torques and forces to produce the desired positions of the angles determined by inverse kinematics.

The optimized model also contains the feedback loop to account for disturbances that might result in different angles than the actual numerically computed angles as is usually the case in physically deployed systems. Figure 4.2 shows the torque variation and the position variation of the optimized robotic arm during simulation. The total torque cost of the optimized robotic arm is 127.2Nm. A total maximum torque of 123.66Nm is required to move the Bicep arm.



(a) Torque required to move various joints during simulation

(b) Position of joints of optimized robotic arm during simulation

Figure 4.2: Torque and Joint angles of optimized robotic arm

5 Conclusion

The main aims of this project were to design the Model of the robotic arm in CAD modeling software which was done in Onshape, to analyze the performance of the end-effector with different joint positions, and finally to simulate the operation of the robotic arm in Simulink.

Throughout this thesis, the CAD model of the 5-DOF robotic arm was explained with the various joints and links. It was shown that the 5-DOF Kuka youBot was designed with five revolute joints positioned at the Pivot, Bicep, Forearm, Wrist, and gripper. The two fingers of the gripper were designed with prismatic joints for translational motion.

Inverse kinematics was used through MATLAB to obtain the various joint angles of the imported CAD model of the robotic arm. The inverse kinematics was used to feed back the positions of the joint angles in the control logic system to control the motion of the robot.

The imported CAD model was integrated with an actuation system designed of motors placed at the various joints and then the model was optimized to reduce power consumption.

The results from the optimization show that, by employing inverse dynamics to determine the required forces required to follow a particular trajectory obtained through inverse kinematics, the robotic arm was able to achieve a low power consumption as compared to a robotic arm that was unoptimized in terms of trajectory.

6 Appendix

Appendix

```
mdl = 'robot_arm';
load_system(mdl);
ik = simscape.multibody.KinematicsSolver(mdl);

base = 'robot_arm/World/W';
follower = 'robot_arm/RevoluteWristGripperPivot/B';
addFrameVariables(ik,'Wrist','translation',base,follower);

frameVariables(ik)
targetIDs = ["Wrist.Translation.x";"Wrist.Translation.y";"Wrist.Translation.z"];
addTargetVariables(ik,targetIDs);

jointPositionVariables(ik)
outputIDs = ["j3.Rz.q";"j5.Rz.q";"j7.Rz.q"];
addOutputVariables(ik,outputIDs);

targets = [-0.16,-0.12,0];
[outputVec,statusFlag] = solve(ik,targets)

viewSolution(ik);

guessesIDs=["j5.Rz.q","j7.Rz.q"];
guesses = [90,90];
addInitialGuessVariables(ik,guessesIDs);
[outputVec,statusFlag] = solve(ik,targets,guesses)
```

Figure 6.1: Code Snippet of Inverse Kinematics

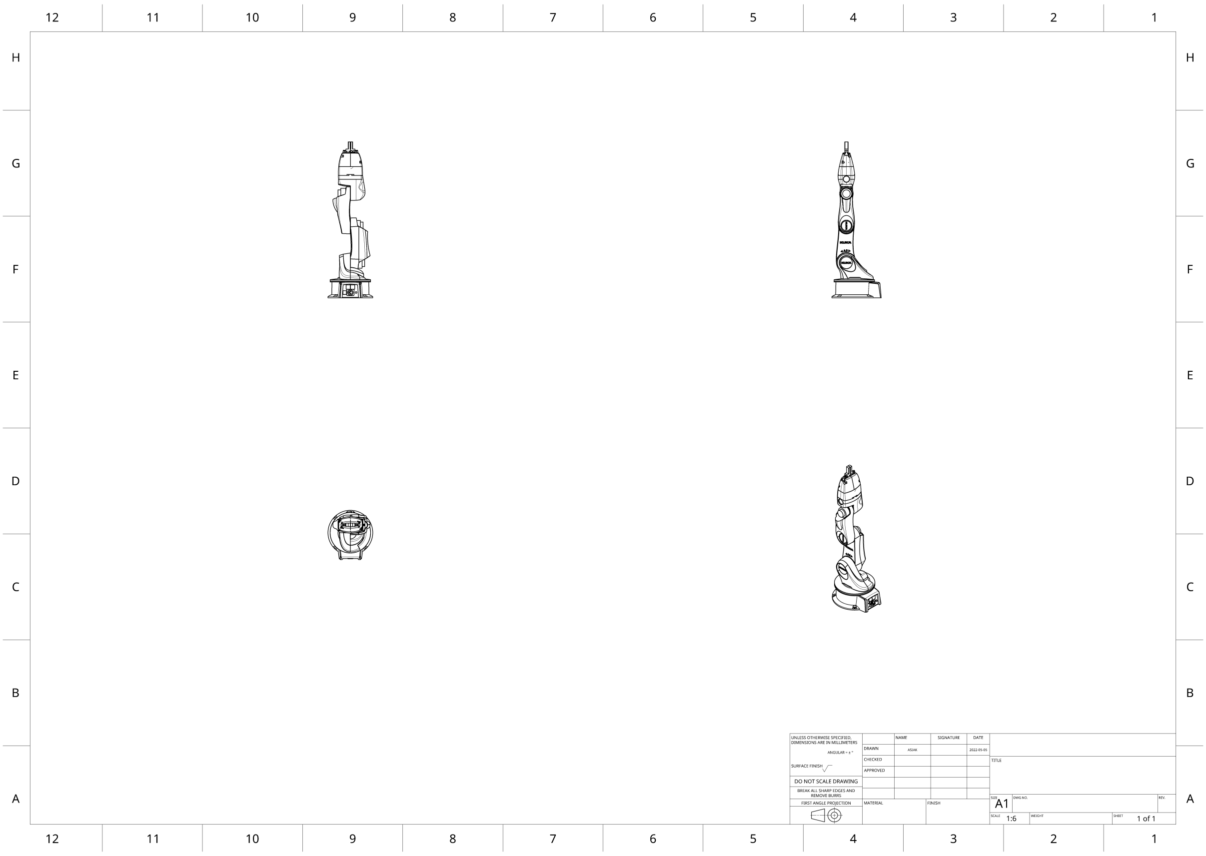


Figure 6.2: Assembly drawing of 5-dof robotic arm

7 References

References

[1] M. E. Moran, "Evolution of robotic arms," vol. 1, pp. 103–111, 2007.

[2] S. Castro, "Robot manipulation, part 1: Kinematics," Apr 2018.

[3] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Springer, 2016.

[4] D. M. Trujillo, H. R. Busby, S. N. Whittlesey, and D. G. E. Robertson, *Two-Dimensional Inverse Dynamics*, p. 103–103. CRC Press, 1997.