DESIGN AND SIMULATION OF A ROBOTIC ARM SYSTEM FOR SAFE RETRIEVAL OF AMMUNITION FROM JAMMED MACHINE GUN

CAPSTONE PROJECT (MEE4099)

A PROJECT REPORT

Submitted in partial fulfilment of the requirement for the award of the Degree of

BACHELOR OF TECHNOLOGY

IN

MECHANICAL ENGINEERING

By

J Chamala Vaishnavi (18BME1008)

Naveen Raj Srinivasan (18BME1011)

Nishanth Rajkumar (18BME1024)

Under the Guidance of

DR. DAVIDSON JEBASEELAN

SCHOOL OF MECHANICAL ENGINEERING

VELLORE INSTITUTE OF TECHNOLOGY, CHENNAI - 600127

(APRIL 2022)

CERTIFICATE

This is to certify that the Project work titled "**DESIGN AND SIMULATION OF A ROBOTIC ARM SYSTEM FOR SAFE RETRIEVAL OF AMMUNITION FROM JAMMED MACHINE GUN**" that is being submitted by **J CHAMALA VAISHNAVI, NAVEEN RAJ SRINIVASAN, NISHANTH RAJKUMAR** is in partial fulfilment of the requirements for the award of **Bachelor of Technology**, is a record of bona fide work carried out under my guidance. The contents of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for award of any degree or diploma and the same is certified.

Dr. Davidson Jebaseelan

Guide

The thesis is satisfactory / unsatisfactory

Internal Examiner   **External Examiner**

Approved by

Head of Department

ACKNOWLEDGEMENT

J Chamala Vaishnavi (18BME1008)

Naveen Raj Srinivasan (18BME1011)

Nishanth Rajkumar (18BME1024)

ISO 9001 : 2015
Certified Estt

Combat Vehicles Research & Development Establishment (CVRDE)
Defence Research & Development Organisation
Government of India, Ministry of Defence
Avadi, Chennai - 600 054

Human Resource Development Division

CERTIFICATE

This is to certify that the under mentioned **B.Tech Mechanical Engineering** student(s), from **Vellore Institute of Technology, Chennai - 600 127**, has/have successfully completed the Project Work titled, **"DESIGN AND SIMULATION OF A ROBOTIC ARM SYSTEM FOR SAFE RETRIEVAL OF AMMUNITION FROM JAMMED MACHINE GUN "** in this organization from 15-12-2021 to 21-04-2022 under the guidance of **Dr. N. BABU, Scientist 'F'** of this Establishment.

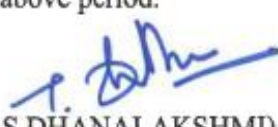|	
Mr. NAVEEN RAJ SRINIVASAN	**Reg No: 18BME1011**
Ms. CHAMALA VAISHNAVI J	**Reg No: 18BME1008**
Mr. NISHANTH R	**Reg No: 18BME1024**

Certified also that the student(s) was/were regular during the above period.

No: CV/7.2/HR/MP/Stu-Pro/36/UG/2022/002
Place: Avadi
Date :21 Apr 2022

(Dr. S DHANALAKSHMI)
Scientist 'F'
For Director

ISO 9001 : 2015
Certified Estt

Combat Vehicles Research & Development Establishment (CVRDE)
Defence Research & Development Organisation
Government of India, Ministry of Defence
Avadi, Chennai - 600 054

Human Resource Development Division

C E R T I F I C A T E

This is to certify that the under mentioned **B.Tech Mechanical Engineering** student(s), from **Vellore Institute of Technology, Chennai - 600 127**, has/have successfully completed the Project Work titled, **"DESIGN AND SIMULATION OF A ROBOTIC ARM SYSTEM FOR SAFE RETRIEVAL OF AMMUNITION FROM JAMMED MACHINE GUN "** in this organization from 15-12-2021 to 21-04-2022 under the guidance of **Dr. N. BABU, Scientist 'F'** of this Establishment.

Mr. NAVEEN RAJ SRINIVASAN	Reg No: 18BME1011
Ms. CHAMALA VAISHNAVI J	Reg No: 18BME1008
Mr. NISHANTH R	Reg No: 18BME1024

Certified also that the student(s) was/were regular during the above period.

No: CV/7.2/HR/MP/Stu-Pro/36/UG/2022/001
Place: Avadi
Date :21 Apr 2022

(Dr. S DHANALAKSHMI)
Scientist 'F'
For Director

ISO 9001 : 2015
Certified Estt

Combat Vehicles Research & Development Establishment (CVRDE)
Defence Research & Development Organisation
Government of India, Ministry of Defence
Avadi, Chennai - 600 054

Human Resource Development Division

CERTIFICATE

This is to certify that the under mentioned **B.Tech Mechanical Engineering** student(s), from **Vellore Institute of Technology, Chennai - 600 127**, has/have successfully completed the Project Work titled, **"DESIGN AND SIMULATION OF A ROBOTIC ARM SYSTEM FOR SAFE RETRIEVAL OF AMMUNITION FROM JAMMED MACHINE GUN "** in this organization from 15-12-2021 to 21-04-2022 under the guidance of **Dr. N. BABU, Scientist 'F'** of this Establishment.

Mr. NAVEEN RAJ SRINIVASAN	**Reg No: 18BME1011**
Ms. CHAMALA VAISHNAVI J	**Reg No: 18BME1008**
Mr. NISHANTH R	**Reg No: 18BME1024**

Certified also that the student(s) was/were regular during the above period.

No: CV/7.2/HR/MP/Stu-Pro/36/UG/2022/003
Place: Avadi
Date :21 Apr 2022

(Dr. S DHANALAKSHMI)
Scientist 'F'
For Director

LIST OF TABLES

LIST OF FIGURES

EXECUTIVE SUMMARY

Heavy Machine Guns are an essential fixture in modern warfare and military combat situations due to their destructive capabilities. The NSVT Utyos 12.7mm Machine Gun is one such equipment of Soviet origin, which has been the gold standard for weapons in this segment all over the world. Despite the largely successful usage of these high calibre weapons, there have also been some incidents wherein the machine gun operation has been hampered due to jammed ammunition. The repair of a jammed gun which is ready to fire the stuck bullet at any moment, is unpredictable and is likely to result in an explosion. Therefore, to save lives, and to safely retrieve these misfired ammunition rounds, a robotic solution has been sought by the Ministry of Defence. This dissertation documents the conceptualization and analysis of a robotic arm system aimed towards that objective. The system was made to be capable of the multiple complicated operations to be performed in sequence for retrieving the ammunition. This spatial robotic arm system has 8 links and 7 degrees of freedom. It has two arms functioning independent of each other, where either arm is dedicated entirely to one set of the required operations. The right arm, equipped with a two-fingered end effector performs heavier external operations regarding disassembly of the gun. The left arm, culminating in a three-fingered end effector is focused towards internal operations or bullet retrieval, where precision is emphasized on. The system also includes a stereo-vision camera, according to whose inputs, the bullet is detected, sought and extracted. Our project further focuses on motion simulation of this system using MATLAB. The Denavit-Hartenberg parameters are calculated to arrive at the transformation matrix, based on which forward and inverse kinematics, trajectory, velocity, acceleration and singularities are calculated and documented. These kinematic studies are validated using MSC ADAMS, in addition to the computation of dynamic properties as in torques and forces necessitated /by the complete operation cycle of the mechanism. Using all this data, a viable solution has been posed to address the problem, and is intended to support further development and manufacture.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Background

NSVT Utyos is a 12.7mm calibre heavy machine gun of Soviet origin. It has a rate of fire of 700–800 rounds per minute. It is effective against both airborne and ground targets, and is used in numerous weapon platforms. It is typically deployed on a height-adjustable tripod with stock and pistol grip, as seen in Figure 1, or on an anti-aircraft mount for usage on vehicles or buildings, as illustrated in Figure 2. A loaded ammunition belt employed in NSVT with 100 rounds weighs 5 kg. This gun is being produced and used in large quantities, in more than 30 countries across the globe. In India, it is license-manufactured at Ordnance Factory Tiruchirappalli.

Figure 1: NSVT on ground

Figure 2: NSVT mounted on Tank

1.2 Problem Statement

The high rate of fire from the NSVT occasionally results in the problem of ammunition being jammed during operation in the battlefield, as reported by different users globally. The various complex mechanisms that occur inside the gun during the firing cycle were studied, and it was inferred that there were certain phenomena that could possibly result in the aforementioned problem. Such an evaluation was necessary to develop an understanding of the causes of the issue before proceeding towards formulating a solution for it.

Once the trigger of the gun is pulled, the cocking lever gets released from its taut position, and the firing bolt of the gun - which is initially held back - strikes the ammunition seated in the belt, pushing it forward to a certain distance, and placing it in the firing chamber. Once this happens, a small protrusion on the face of the firing bolt pierces the dorsal surface of the ammunition, leading to combustive release of compressed propellant gases, which are responsible for the projectile being thrust out of the gun, and towards the target (Figure 3).
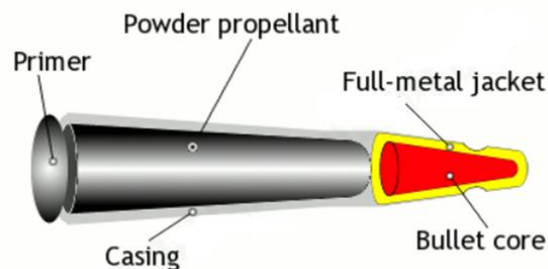
Figure 3: The different parts of a bullet

From the point of triggering to the ejection of the projectile, there are several processes that could contribute towards the jamming of the bullet in various positions inside the gun, and the consequent inability of it to be ejected. This situation renders the weapon unusable, and hence manual intervention is needed by the operator to rectify it. The complication here is that, by doing so, the operator is exposed to the chance of explosion of the bullet when the chamber is opened or when the belt is pulled out. This may occur due to the inadvertent activation of primer in the ammunition, which has not yet been consumed. Therefore, the most effective solution for this would be an equipment that retrieves the ammunition from the gun

by itself, thus avoiding human involvement and performing this complicated task efficiently.

This problem statement has been posted by the Ministry of Defence, India, as part of their series of Technology Development Fund (TDF) Projects (Figure 4).
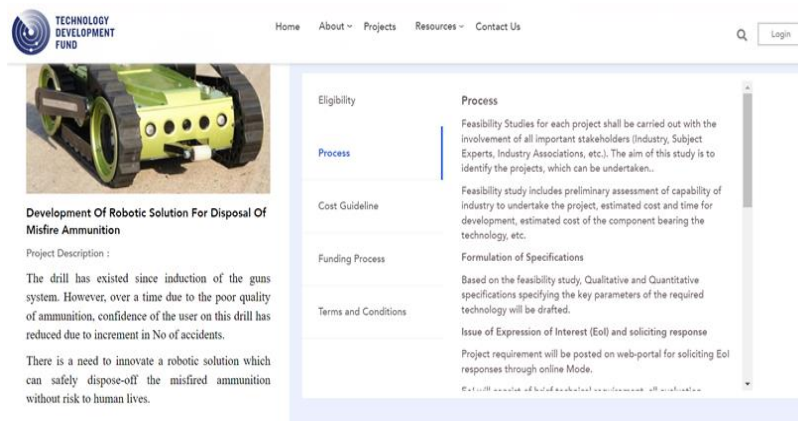
Figure 4: Problem Statement as posted on the TDF website

The possibilities of ammunition jamming have been found to occur by means of four major mechanisms, as discussed below with the help of illustrations:

Case 1:
The most straightforward type of obstruction happens when the bullet is misfired, but still remains in the ammunition belt. This may be a result of misaligned belt assembly or firing bolt failure. This case is the easiest to be solved, as the removal of the belt would also result in removal of the erroneous ammunition (Figure 5).
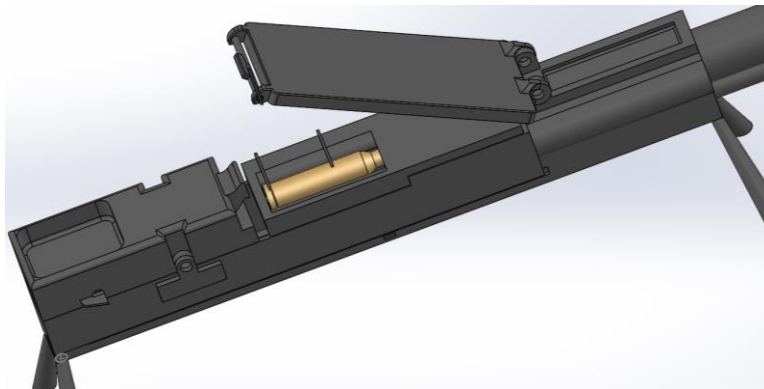
Figure 5: Ammunition jammed in ammunition belt

Case 2:

In this case, the ammunition is successfully ejected from the belt, but is jammed in a part of the breech, on its way to the chamber. This mode of jamming entails opening of the breech cover and ammunition tray before gaining access to the bullet (Figure 6).
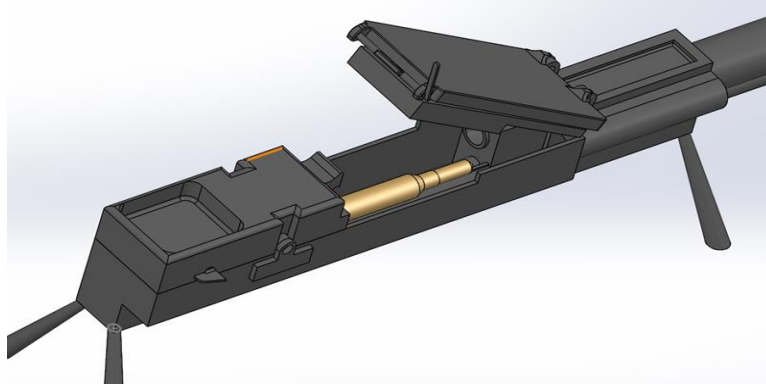
Figure 6: Ammunition jammed in a part of the breech

Case 3:

Here, the ammunition is successfully ejected, and pushed across the breech onto the mouth of the chamber, but does not get seated properly due to misalignment of its axis to the axis of the chamber. The bullet may either be stuck in a diagonal position, or may have entered the chamber partially, with a portion of itself remaining exposed (Figure 7).
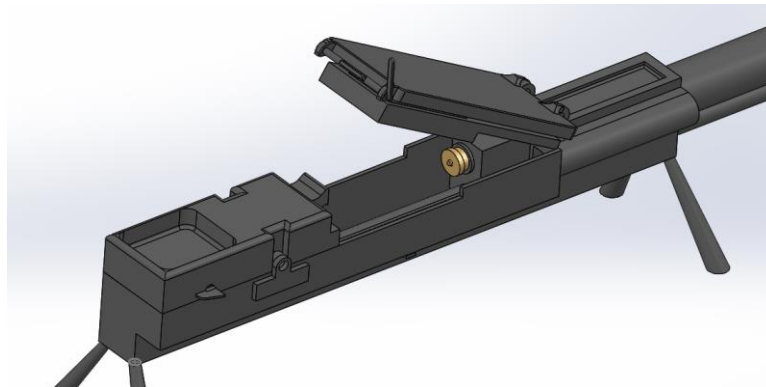
Figure 7: Ammunition jammed partially in the firing chamber

Case 4:

The most complicated case is where the bullet completes its travel from the ammunition belt to the firing chamber perfectly, but does not get fired from that point. Unlike the previous case, the bullet is seated in the chamber properly, but the firing bolt fails to eject it. The chamber houses a slot matching the profile of the bullet, so that it remains in a loose fit, but just enough to move without significant hindrance due to friction (Figure 8).
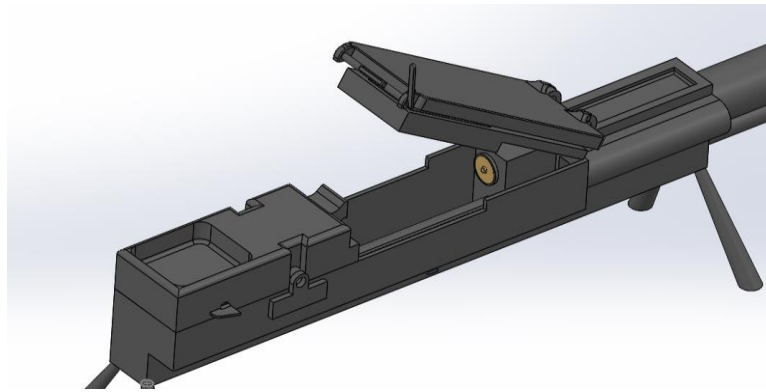
Figure 8: Ammunition jammed inside the firing chamber

All the cases listed above have been accounted for, in the course of building a solution towards the requirement. A specialised robotic arm system has been formulated to that end, which will be discussed further in this report.

CHAPTER 2

INFERENCES FROM EXTANT LITERATURE

In the conceptualisation stage, various literature sources were studied to obtain valuable working knowledge and insights pertaining to the machine gun and the objective. A number of technical papers and repositories had been explored to collect data on the architecture and working principles of the NSVT 12.7 mm machine gun. The Small Arms and Machine Guns SA/E 510 No.2 report titled "Gun, Machine (MAG TK 71), 7.62 mm 5A and 6A" provided detailed insight on the components used in the MAG TK 71 gun, which was largely analogous to our application. A report titled "Clearing of Live Ammunition from Guns", published by the Chief of the Bureau of Naval Weapons, in the United States explains the precautions and procedures to be followed in removing live ammunition from naval gun barrels. Information contained in this publication had been obtained from past experience and tests. Exact compliance with the procedures outlined in this report was not directed, but the reason for departure was advised to be justified carefully by the activity concerned. This was highly beneficial towards establishing the requirements and considerations to be employed while building an application to perform that task artificially.

Furthermore, the principles of robotic arm form and function were studied, along with how they are employed in various fields to fulfil tasks. A robotic arm is a type of mechanical arm, with similar functions to a human arm. The arm may be the sum total of the mechanism or may be part of a more complex robot. The links of such a manipulator are connected by joints allowing either rotational motion or translational displacement. The links of the manipulator can be considered to form a kinematic chain. The terminus of the kinematic chain of the manipulator is called the end effector, and it is analogous to the human hand. Robotic arms are machines that are programmed to execute a specific task or job quickly, efficiently, and extremely accurately. Generally, motor-driven, they are most often used for the rapid, consistent performance of precise, heavy and/or highly repetitive procedures over extended periods of time, and are especially valued in the industrial production, manufacturing, machining and assembly sectors.
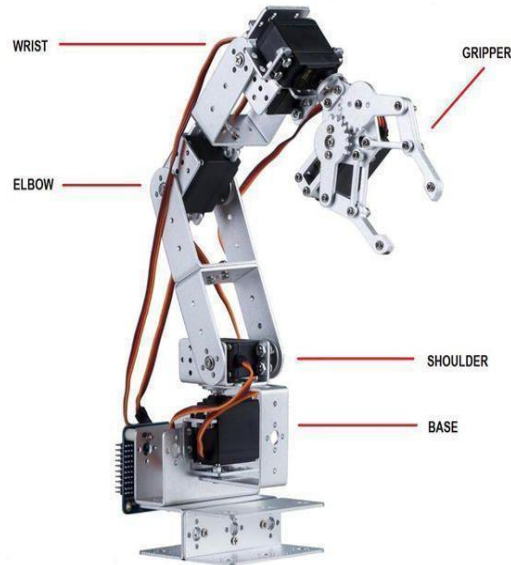
Figure 9: A typical robotic arm

A typical robotic arm as the one seen in Figure 9, is made up of "n" number of metal segments, joined by "n-1" number of joints. A computer controls the robot by rotating individual step motors connected to each joint. Unlike ordinary motors, step motors move in exact increments. This allows the computer to move the arm very precisely, repeating exactly the same movement over and over again. The robot uses motion sensors to make sure it moves just the right amount.

An object in space has six degrees of freedom (DOF), in translatory motion along the X, Y, and Z-axes (3 DOF), and rotary motion about the X, Y, and Z-axes (3 DOF), as illustrated in Figure 10.
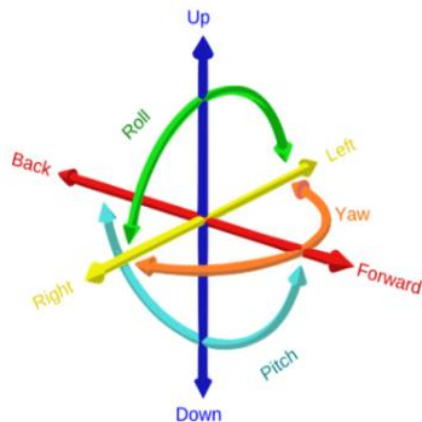
Figure 10: DOF of an object in space

Although a rigid body has 6 DOF, due to the formation of linkage one or more DOF is lost due to the presence of constraint on the body. The total number of constraints cannot be zero as the body has to be fixed at some place to make the linkage possible.

A robotic arm with six joints closely resembles a human arm -- it has the equivalent of a shoulder, an elbow and a wrist. Typically, the shoulder is mounted to a stationary base structure rather than to a movable body. This type of robot has six degrees of freedom, meaning it can pivot in six different ways. A human arm, by comparison, has seven degrees of freedom which are explained in Figure 11.

The shoulder has 3 DOF: Shoulder pitch, shoulder roll and should yaw
Elbow has 1 DOF: Elbow
The wrist has 3 DOF: Wrist pitch, Wrist roll, and Wrist yaw
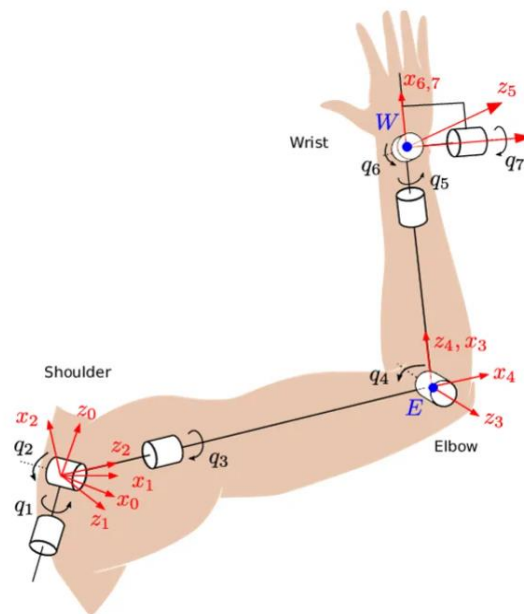
Figure 11: 7 DOF of human arm

A robotic arm's job is to move an end effector from place to place. Robotic hands often have built-in pressure sensors that tell the computer how hard the robot is gripping a particular object, and keeps the robot from dropping it.

Some robotic arm patents were referred to, in order to understand the range of possibilities involved in creating such an appliance.

The patent titled Sprung worm gripper for a robotic device featured a finger having a worm gear coupled to its base end (Figure 12). The device also includes an actuator having a motor and a shaft, wherein the shaft is configured to rotate a worm coupled to the worm gear, and the actuator is mounted on a carriage such that the actuator is configured to slide along an axis. The device also includes a spring having first and second ends, wherein the first end is coupled to the motor and the second end is fixed. Further, the actuator is configured to rotate the shaft relative to the motor by a first amount to move the finger toward an object, and when the finger is in contact with the object and is prevented from further movement, further rotate the shaft relative to the motor to slide the actuator along the axis.
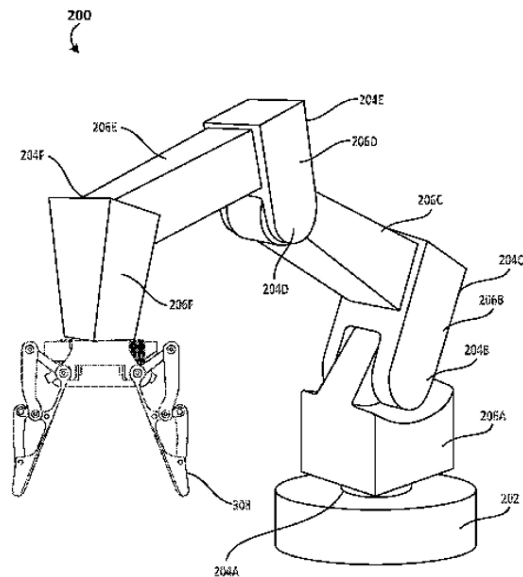
Figure 12: Robotic device featured on the patent titled Sprung worm gripper

The patent titled Specimen holding robotic arm end effector is an apparatus specially adapted for handling semiconductor or electric solid state devices during manufacture or treatment It is a robotic arm end effector for catching a specimen. It is directed to an apparatus and method for handling specimens, which substantially reduces damage and specific contamination of the wafer backside.

Another interesting patent was the Configurable robotic surgical system with virtual rail and flexible endoscope (Figure 13). Systems and methods for moving or manipulating robotic arms are provided. A group of robotic arms are configured to form a virtual rail or line between the end effectors of the robotic arms. The robotic

arms are responsive to outside force such as from a user. When a user moves a single one of the robotic arms, the other robotic arms will automatically move to maintain the virtual rail alignments. The virtual rail of the robotic arm end effectors may be translated in one or more of three dimensions. The virtual rail may be rotated about a point on the virtual rail line. The robotic arms can detect the nature of the contact from the user and move accordingly. Holding, shaking, tapping, pushing, pulling, and rotating different parts of the robotic arm elicits different movement responses from different parts of the robotic arm.
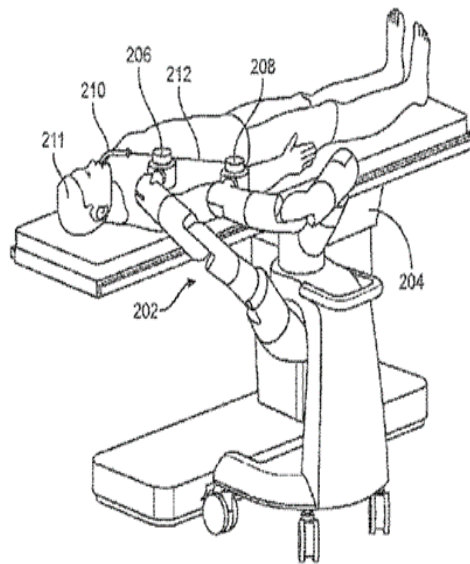
Figure 13: Configurable robotic surgical system

Robot arm structure, workpiece processing system, and system including robot arm structure and sensor array, relates to a precision arm mechanism capable of moving back and forth along a straight line including a straight line extending in a non-radial direction and capable of moving along a desired non-linear path. The device can be tilted from the normal vertical axis to correct workpiece misalignment. It can pick up and deliver a workpiece that is tilted from the normal operating plane of the apparatus. It is suitable for positioning various objects such as panels, computer hard disks, cassettes, etc. for processing and use.
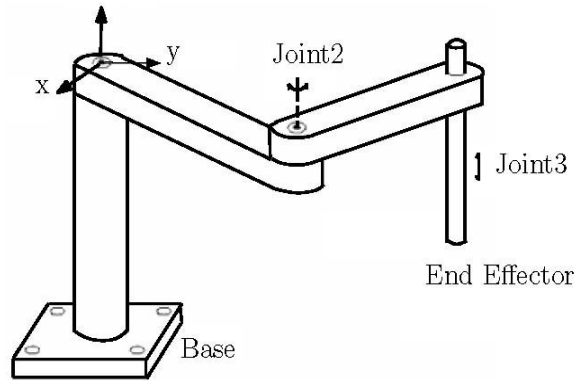
Figure 14: SCARA Robot

SCARA Robots are a popular option for small robotic assembly applications, which can be seen in Figure 14. SCARA is an acronym for Selective Compliance Articulated Robot Arm, meaning it is compliant in the X-Y axis, and rigid in the Z-axis. The SCARA configuration is unique and designed to handle a variety of material handling operations. The SCARA's structure consists of two arms joined at the base and the intersection of first and second arms. Two independent motors use inverse kinematics and interpolation at the joints to control the SCARA's X-Y motion. The final X-Y location at the end of arm two is a factor of the joint angles, length of first arm, and length of second arm.
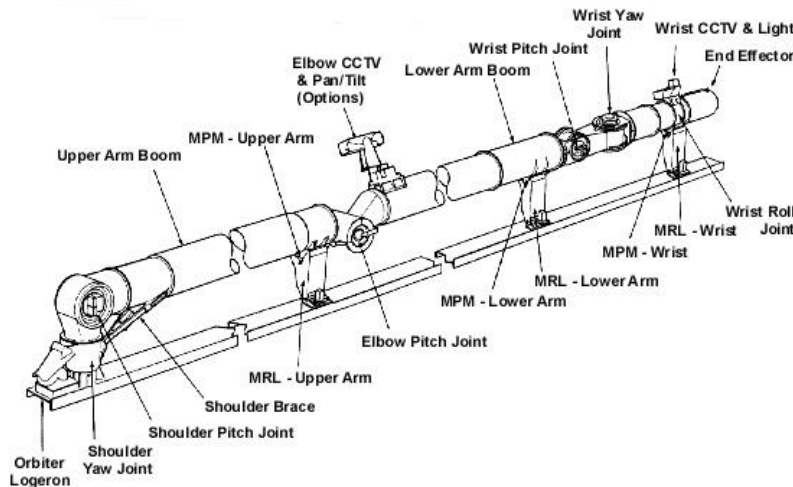
Figure 15: Canadarm

The Shuttle Remote Manipulator System or Canadarm, shown in Figure 15, was a robotic arm for the deployment/retrieval of space hardware from the payload bay of a NASA orbiter. It consists of a shoulder, elbow and wrist joint separated by an upper

and lower arm boom giving it a total of six degrees-of-freedom (shoulder pitch and yaw, elbow pitch and wrist pitch yaw and roll). Each joint is made up of motor driven gearboxes that allow the basic structure of the arm to articulate much like the human arm. There are two motors in the shoulder joint which allow the whole arm to pitch (up and down motion) and yaw (side to side motion). One in the elbow joint to allow the lower arm to pitch and three in the wrist joint to allow the tip of the arm to pitch, yaw and roll (rotating motion). The motors are equipped with their own brakes and joint motor speed control. Each motor also incorporates a device called an encoder, which accurately measures joint angles. Thus, each joint is capable of moving independently at different speeds and in different directions. The End Effector or mechanical hand allows the arm to capture stationary or free flying payloads by providing a large capture envelope and a mechanism/structure capable of soft docking. Its elbow and wrist joint cameras provided visual inspection of the shuttle and its payload. The robotic arm could be operated manually by an astronaut at the controls or programmed to function automatically.

CHAPTER 3

PROPOSED SOLUTION

3.1 Design Considerations

After extensive study on the problem statement concerning the heavy machine gun and its firing mechanism, we learned how a jammed ammunition round is to be retrieved manually. This objective requires several tasks to ensure the gun is made to be in a condition where retrieval is possible in the first place, and to seek out ammunition according to the different cases of jamming as explained earlier. The sequence of operations involved in the process is as follows:

1. Cocking
2. Safety Lever Operation
3. Breech Cover Opening
4. Belt Retrieval
5. Cartridge Tray Opening
6. Ammunition Extraction

So, the proposed robotic arm system should not only be capable of performing these tasks, but should also execute them in sequential order. The necessity for each of these tasks and plan to accomplish will be explained further in this dissertation. Before that, the design of the robotic arm system will be illustrated and delineated in the following section.

3.2 Concept Design

In order to arrive at the design requirements of the system, we initially had to build up an accurate, life-sized model of the NSVT machine gun. Following which, it was decided that the system needs to be placed on the right side of the machine gun when viewed from the position of the operator (behind the gun). This is because the ammunition belt is fed into the gun from the right to the left, and hence the empty ammunition links after firing would free-fall and get stored in a dedicated collecting pouch placed on the left side of the gun. Keeping the system on the right precludes removal of the pouch, thus simplifying the setup for the user. The system is deemed

to carry two separate arms independent of one another, because the operations required to be performed involve ones entailing precision, and others entailing strength. Besides, a strong base is also required for cocking and for holding up the whole apparatus. Once these properties were fixed, further detailing and features were incorporated into the concept, and a Computer-Aided Design (CAD) model was created using SolidWorks, as illustrated in Figure 16, by referring to extant designs of robots used elsewhere and by picking out ones with form and function similar to those we need.

The system has seven degrees of freedom in total, distributed among two custom-designed arms, viz. Distal Left and Vicinal Right arm (with relation to the gun), working in tandem with a T-joint to which the links are attached to, and a base link to which the T-joint is attached to. The whole system is mounted on a carriage which is the bottom-most link, and has linear motion by means of a lead screw. Table 1 contains a detailed summary of all the links and joints that make up the robotic arm system. The material for the bulk of the link bodies is chosen to be the Aluminium 7075 t6 alloy, on account of its high formability, high yield strength - even comparable to that of steel - toughness and resistance to fatigue. It is also commonly used as building material in robots. The total mass of the system comes out to about 6.865 kg.
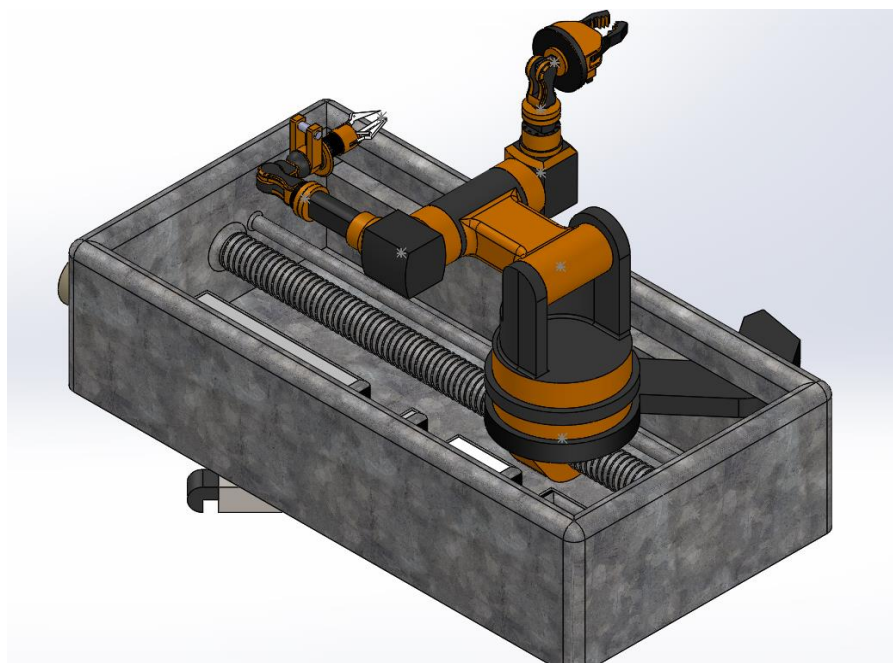
Figure 16: Isometric View of the Proposed model

Table 1: Summary of Links and Joints

Link	Name	Pre-Link	Suc-Link	Joints involved	Mass (g)
1	Substructure Unit	-	2	1. Prismatic (with 1) (+/-X)	4397.67
2	Carriage	1	3	1. Prismatic (with 1)(+/-X) 2. Rotary (with 2)(+Y)	564.86
3	Base Support	2	4	1. Rotary (with 2)(+Y) 2. Rotary (with 4)(+Z) Waist	506.36
4	Torso - T-Joint	3	5R,5L	1. Rotary (with 3)(+Z) Waist 2. Rotary (with 5L,5R)(+Z) Shoulder	524.64
5R	Upper Arm - Antepenultimate Link R	4	6R	1. Rotary (with 4)(+Z) Shoulder 2. Rotary (with 6R)(+Z) Elbow	415.91
5L	Upper Arm - Antepenultimate Link L	4	6L	1. Rotary (with 4)(+Z) Shoulder 2. Rotary (with 6L)(+Z) Elbow	288.62
6R	Forearm - Penultimate Link R	5R	7R	1. Rotary (with 5R)(+Z) Elbow 2. Rotary (with 7R)(+Z) Wrist	23.42
6L	Forearm - Penultimate Link L	5L	7L	1. Rotary (with 5L)(+Z) Elbow 2. Rotary (with 7L)(+X) Wrist	41.66
7R	Tool - Vicinal End Effector	6R	-	-	40.65
7L	Tool - Distal End Effector	6L	-	-	59.60

3.3 Part Descriptions

3.3.1 Internal Arm

The distal arm (or internal operation arm or left arm) of the system comprises an upper arm and a lower arm, as seen in Figure 17, culminating in a mutable end effector, having the options of equipping either a mechanical gripper or a vacuum-based gripper. This arm is characterised to perform the precise operations involving ammunition retrieval from the interior of the gun. The lower arm has two cameras mounted on its top surface. The lower arm could be rotated about the roll axis, for the cameras to be held at a position on-looking the gun, in a stereoscopic setup, to perceive a three-dimensional view. This image input is analysed by the accompanying processing unit, at the commencement of every step, in order to plot the path that each link needs to traverse to perform the required operation.
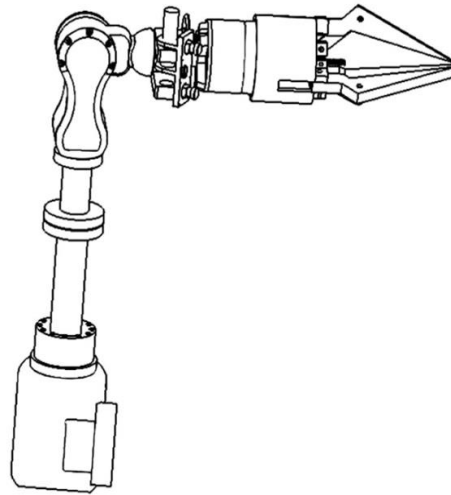
Figure 17: Internal Arm

3.3.2 External Arm

The vicinal arm (or external operation arm or right arm) of the system comprises an upper arm and a short lower arm, as illustrated in Figure 18, culminating in a three-fingered gripper. This arm is intended to perform the operations which enable the gun to be disassembled and ready for ammunition retrieval. The upper arm is joined with the T-joint, and the lower arm is essentially a housing for the motor which renders the gripper capable of rotation about the pitch axis. The operation of this arm does not entail any visual output from the camera because it manipulates the parts

of the gun in safety lever, breech cover and ammunition tray, whose positions are well-defined and can be programmed readily into it.
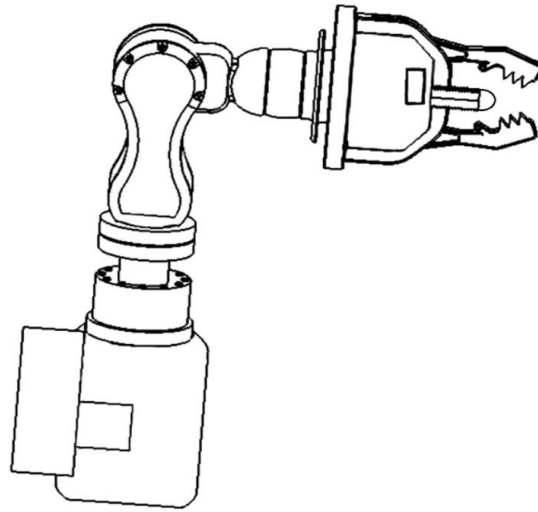
Figure 18: External Arm

3.3.3 External Arm Gripper

The gripper for external operation is tasked with turning on the safety switch. It is attached to a forearm link in a wrist joint that facilitates rotation about the pitch axis, and is capable of pushing the switch inward and applying a twisting moment to it, while retaining the push force. This is done as a result of the gripper having three fingers – one in the middle, and one either side of it (Figure 19). The middle finger is driven by a linear actuator, and acts as a rack to one pinion gear on either side. Those pinions are idlers to impart the desired directions of rotation to gears that are coupled with a lateral finger each. This amounts to the forward motion of the middle finger effecting inward movement of the lateral fingers in a direction about the roll axis. This would mean that the safety switch is gripped and pushed in one motion. Once the gripping and pushing of the safety switch is done, then in that same state, the entire gripper is rotated clockwise to ensure that the gun is safe for ammunition extraction.
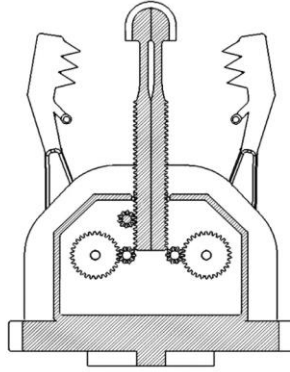
Figure 19: External Arm Gripper designed to facilitate the safety lever manipulation

3.3.4 Internal Arm Grippers

Figure 20: Vacuum cup end effector
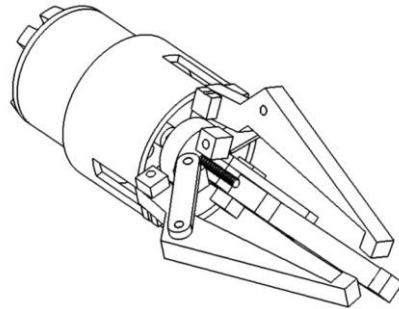
Figure 21: 3-fingered gripper of Internal Arm
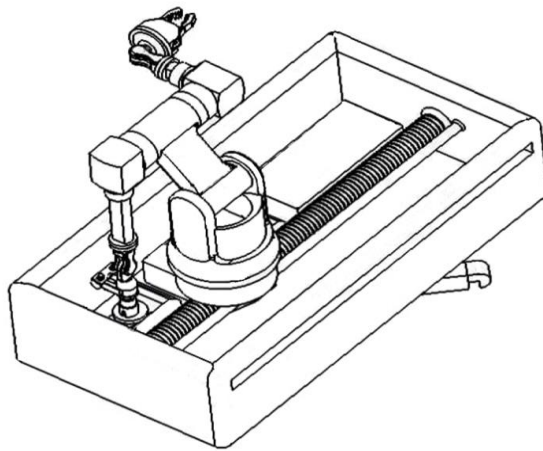
Figure 22: Tool Changing Unit

The gripper for internal operation is tasked with ammunition retrieval from multiple locations. It is attached to a forearm link in a wrist joint that facilitates rotation about the roll axis. There are two possible end effectors that the forearm could equip itself with, a vacuum-based gripper (Figure 20), and a mechanical gripper (Figure 21). The switching between the two grippers is done by means of a magnetic clutch mechanism (Figure 22) at the end of the forearm, which could be powered on to equip a gripper, and powered off to let go of it, once it reaches the gripper's designated slot on the substructure unit. The mechanical gripper could perform extraction of the ammunition belt, and the bullet from the breech or from a position in the chamber where it is able to hold onto the bullet. In case the bullet is found deep inside the chamber with no protrusion that could mechanically be grabbed onto, the vacuum gripper comes into play, which would then create a vacuum on the hind face of the bullet, using which it could be dragged out.

The pressure created by the vacuum-based gripper is calculated as follows:

$$Mass\ of\ the\ bullet = 0.05\ kg$$

$$Force\ required = 0.5\ N$$

$$Diameter\ of\ the\ bullet = 0.02\ m$$

$$Force(F) = Pressure(P)\ x\ Area(A)$$

$$P = \frac{F}{A}$$

$$P = \frac{0.5\ x\ 4}{\pi\ x\ (0.02)^2}$$

$$P = 1591\ Pa$$

Since the pressure required for the operation is well under the atmospheric pressure value, a plug-and-play type of gripper could be used which generates the pressure using its internal valve mechanism. Higher pressure requirements could be handled by incorporating a vacuum generator within the apparatus.

3.3.5 T-Joint
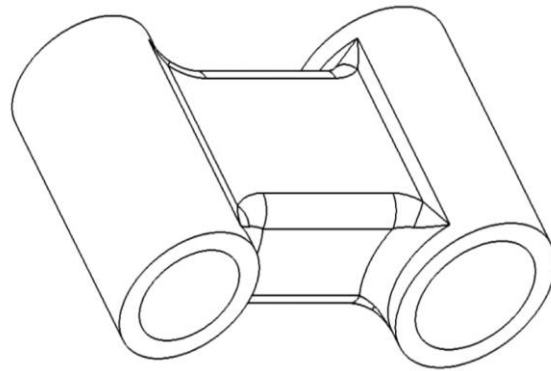
Figure 23: T-Joint

The T-Joint (Figure 23) is the interface between the basal links and the arms. It essentially acts as the part from which the system bifurcates to form the two arms. On its dorsal side, it is driven by one motor, and on the ventral side, it is coupled to two different motors, which drive the two arms individually. It is capable of rotation about the pitch axis.

3.3.6 Base Support
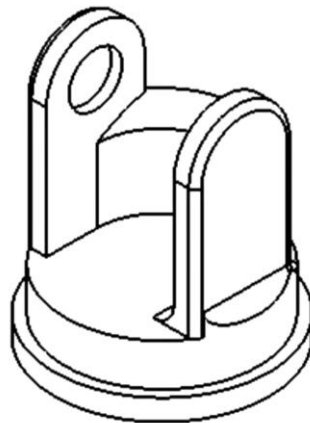
Figure 24: Base Support

The Base Support (Figure 24) connects the carriage to the T-joint. It is the first revolute joint of the system, and the only joint capable of rotation about the yaw axis. It does not contribute to the external arm operations, since all those parts lie in the same plane, but it helps extensively in the internal ammunition retrieval tasks.

3.3.7 Carriage
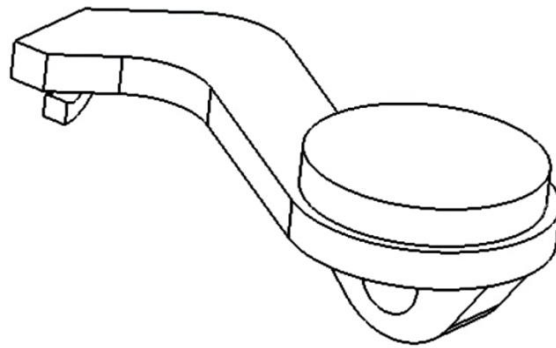
Figure 25: Carriage

The Carriage (Figure 25) is the only prismatic link in the system. It is involved in every operation because the nature of the workspace is that all critical points are widely-spaced from one another. It translates on a lead screw setup driven by a linear actuator. It solely performs the operation of cocking by means of the cocking arm that projects towards the gun.

3.3.8 Substructure Unit
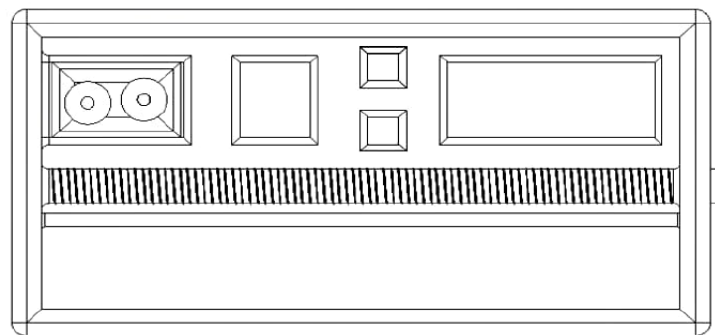
Figure 26: The different compartments in the Substructure Unit
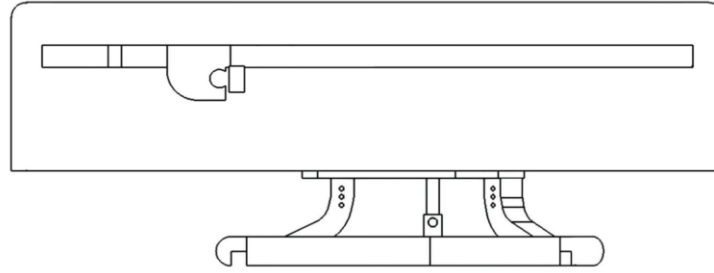
Figure 27: Tripod Stand

The substructure unit houses the entire system, as could be seen in Figure 26, through the lead screw fitted onto it, to which the carriage is coupled. Thus, the substructure bears the weight of the entire system, along with additional components such as the battery pack and the control units. It also has dedicated slots on it for each of the mutable left arm grippers to be placed, when not in use. It has a foldable tripod stand (Figure 27) underneath which is to be placed on the ground, with a height-adjustment lever helping the user to adjust the height according to the mount of the gun, so that once fixed, the lead screw axis is in line with the cocking pin of the gun. Moreover, the standard operating procedure for assembly also requires that the cocking arm of the carriage is made to butt with the cocking pin, so that cocking happens effectively. A slot is given on the arm matching the profile of the pin to bring about this contact. This also ensures that the rest of the arm is in tune with the gun, so no further calibration is needed, and the calculated input kinematic and dynamic values bring about the operations required to be done.

3.3.9 Stereo Vision Camera
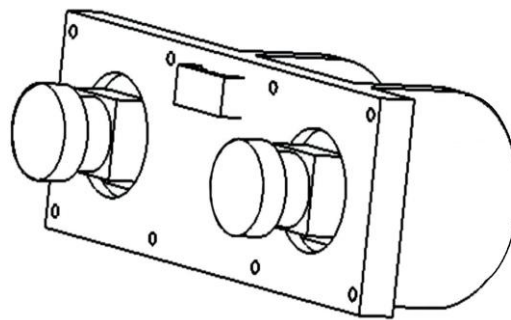
Figure 28: Stereo Vision Camera

A stereo vision camera (Figure 28) has two or more lenses with a separate image sensor or film frame for each lens. This allows the camera to simulate human binocular vision, and therefore gives it the ability to perceive depth, and capture three-dimensional images. The human vision perceives depth by using stereo-disparity which refers to the difference in image location of an object seen by the left and right eyes, resulting from the eyes' horizontal separation. Stereo-disparity in a camera can be found by using two 2D images taken from different positions and the correlation between the images can be used to create a depth profile. This camera is strategically positioned on the link preceding the end effector on the left arm, so that it would always have the best vantage point from the home position to capture images. These images are then used to identify the presence of the bullet in a certain region of the gun, and if identified, the position and orientation in terms of coordinates. This data will be used by the motor controller to actuate the joints.

3.3.10 Controller

The controller is the "brain" of the robotic arm and makes the different parts of the robot operate together. It works as a computer and runs a set of instructions through programs written for its operations. It has an image processing unit, which receives input from the stereo camera and cross-references it with a predefined catalogue containing the profile of the bullet which needs to be sought, to identify its position and orientation. This would then be used for inverse kinematics computation which gives the joint displacements, using which eventually the power to be supplied to each joint motor to bring about the necessary motion is calculated and distributed accordingly. The system is provided with an electric drive, which would be controlled by DC motors with encoders to provide closed-loop feedback for better control. Thus, the controller used has to be a DC motor controller.

3.3.11 Contact Sensor

The slot in the cocking arm, created especially to be in touch with the cocking pin of the gun, is equipped with a contact sensor. This is because the slot being in full contact with the pin is what would ensure the proper setup of the system with relation to the gun, so that all operations can be carried out smoothly without errors due to misalignment. A push button type of sensor is used here, which acts like a switch

which completes a circuit to provide indication to the user that the contact is established and the system is ready for operation.

3.4 Operation Descriptions

The operations to be done, and how the robotic arm system has been conceptualised and designed towards facilitating those, would be explained in sequence in this section. The flow diagram shown below (Figure 29) comprehensively represents the flow of all the operations involved in the entire working of the system. The methodology behind the kinematic and dynamic calculations and basis for them will be explained in the further chapters of this dissertation.
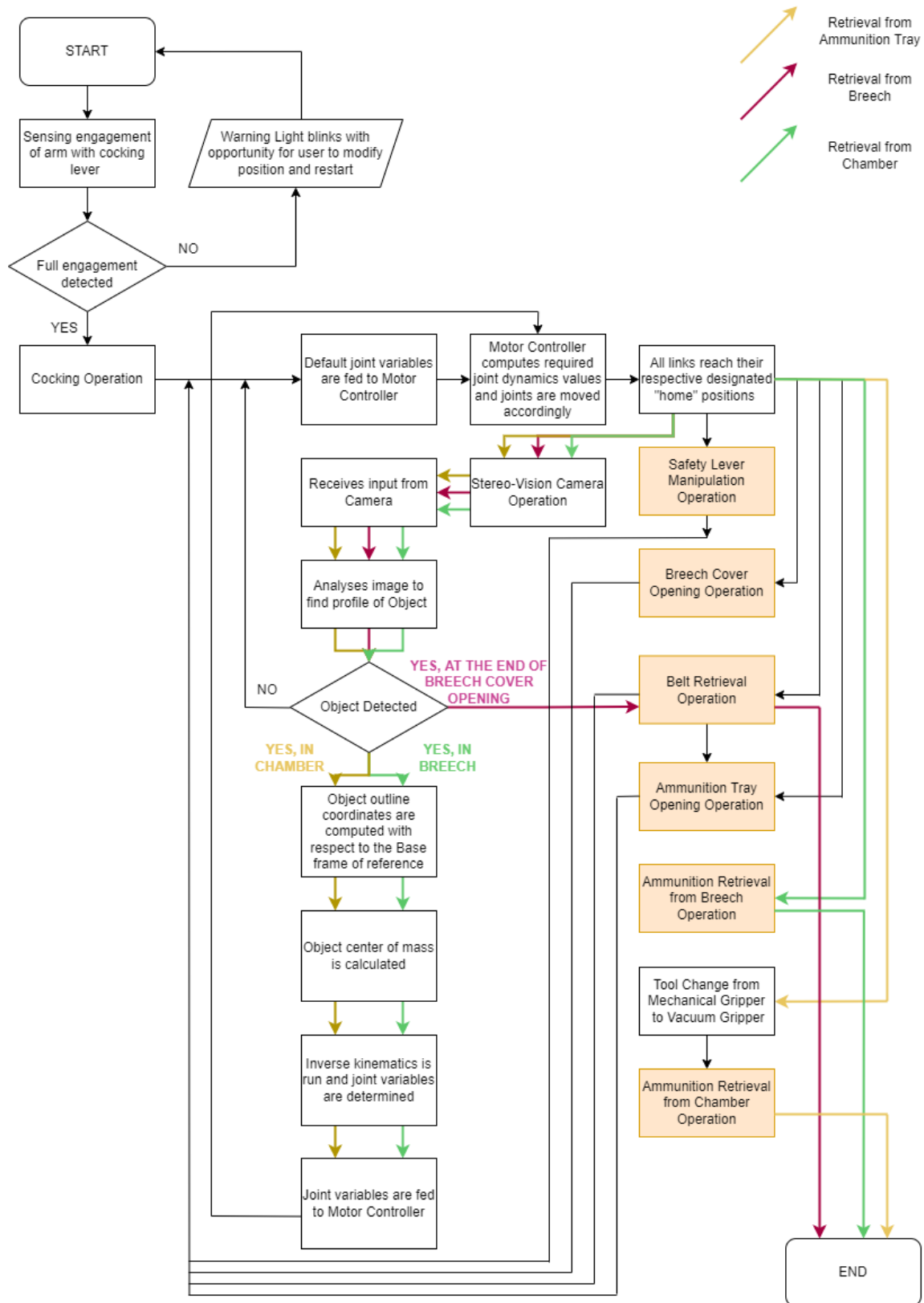
Figure 29: Workflow Diagram

3.4.1 Cocking

The cocking lever is essentially the part of the firing bolt that moves through a slot and is exposed outside for handling. In order for the bullet to be fired, the lever has to be pulled horizontally backward, up until it is locked in position at the maximum compression state of the spring that it is coupled with. This is because the bolt has to be in position to move forward and strike the bullet, on activation of the trigger, followed by release of the cocking lever. This process is done manually at the beginning, and would be automated for further continuous firing through a solenoid that gets tipped off by the gas recirculation as a result of the ejection of projectile from the preceding bullet.

However, in case the bullet gets stuck in between before the fulfilment of ejection, then the cocking lever would have reached the end of the slot again, and would not have come back into the coiled position. This creates a two-fold difficulty, in which neither is the breech free for some sort of intervention to attempt to dislodge the bullet; nor is such an activity safe owing to the impending possible release of the bolt that is coiled to an extent - depending on the erstwhile passage of the bullet before jamming, - and is only halted by the misfiring of the bullet, which would mean that the intervention would end up being counterproductive. Therefore, the foremost operation to be done by the system has to be to cock the lever back, regardless of the position it is in, to obviate any sort of disturbance posed by the firing bolt to the retrieval process.
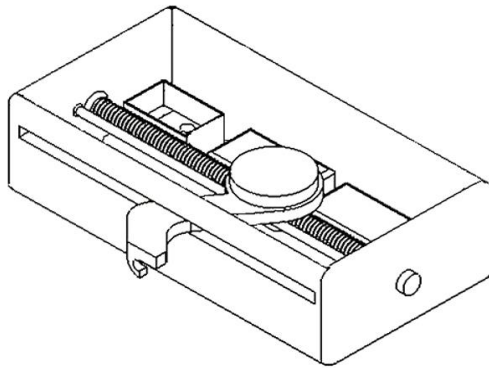
Figure 30: Cocking

This system features a carriage at the inferior end. The carriage is driven between its extreme positions by a ball screw stepper motor linear actuator. A protrusion arises from the carriage along the direction of its pitch axis, which culminates in a vertical plate, which transmits the motion of the carriage to the cocking lever, and cocks the gun, , as illustrated in Figure 30. Once cocking is done, the carriage would move back and forth as necessitated by the further operations, with the plate being redundant.

3.4.2 Safety Lever Operation

To further ensure that the firing bolt stays restrained, without being released even by the action of the trigger, the gun carries a mechanical safety switch. This needs to be turned on to exhaustively arrest the release of the firing bolt, thus making it safe for the ammunition retrieval operation.
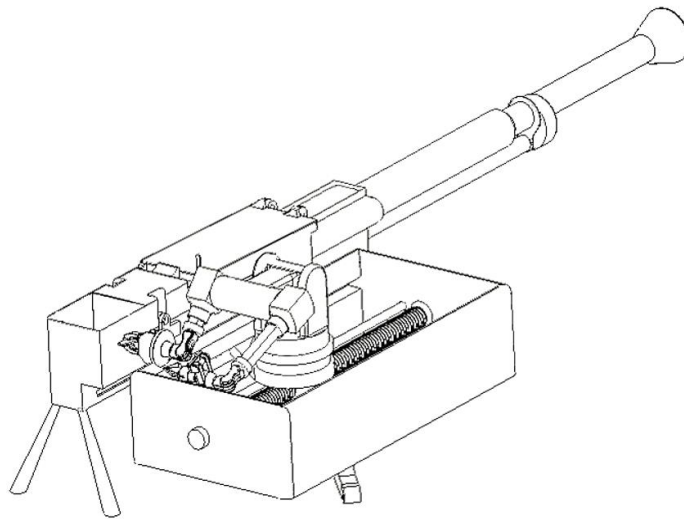
Figure 31: Turning Safety mode ON

The system features an arm that is vicinal to the machine gun, comprising an end effector that operates this switch (Figure 31).

3.4.3 Breech Cover Opening

Once the safety has been made sure of, the next operation is to open the feed cover and access the cartridge tray, which is the first possible point of recovery

for jammed ammunition. To open the feed cover, the feed cover latch needs to be rotated about the pitch axis of the system, and an amount of thrust needs to be imparted to it in the upward direction, up to a certain minimum distance, beyond which it could be let loose to rest in its maximum open position, providing access to the interior of the gun, for ammunition retrieval.
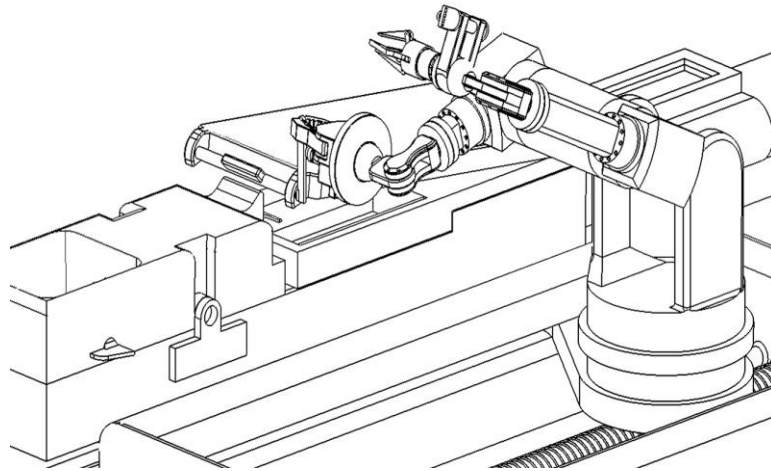
Figure 32: External Arm of the Robot opens the breech cover

The gripper for external operation is tasked with opening the feed cover latch (Figure 32), and has three fingers capable of pushing the switch inward and applying a twisting moment to it while retaining the push force.
At this juncture, the forearm of the gripper is driven at the point of its waist to pull the gripper - along with the latch - in the upward direction, up to a certain level of height, beyond which the gripper relaxes its grasp on the latch by reverse driving of its internal gear system, so that the feed cover is opened to its maximum.

3.4.4 Ammunition Retrieval and Lifting of Belt

Once the feed cover is opened, the cartridge tray is exposed. At this stage, the system would have completed all its external operations and moved onto the internal operations culminating in successful retrieval of ammunition from the gun. To that end, the vicinal arm of the system would retract unto itself as the forearm rotates about the pitch axis to fold up alongside and parallel to the torso link.

Once the external operation arm is retracted, the system would rotate for 90 degrees about the yaw axis at the joint between the base and the carriage. This is done to provide better access to the cartridge tray of the gun for the internal operation arm.

The cartridge tray, which is the first possible point of recovery for jammed ammunition, is scanned thoroughly for the possibility of jammed ammunition located in the belt or in nearby regions that would be accessible from this point. To access imagery on the cartridge tray, the lower arm carrying the camera is rotated 90 degrees about the roll axis, and is positioned over the gun by the drive from the waist. If ammunition is found to be jammed anywhere on the cartridge tray, then the internal operation arm would position itself accordingly at a distance over the location of the ammunition. This clearance is provided so that the arm and its camera setup do not collide with the geometry of the gun.

From this point, the prismatic joint between the gripper and the lower arm drives the gripper inwards. The two-finger gripper would clutch the ammunition belt at a vantage point and pull it outward due to the action of the drive provided at its waist. Then the gripper would let go of the belt, and it would be dropped on the ground, essentially isolating it from the gun. If ammunition is found jammed in the belt, then it would have been retrieved as the belt was removed from the gun.

If ammunition is not found in the belt, but on the cartridge tray, then the gripper would be positioned towards it, and the bullet is gripped and extracted. If ammunition was not found anywhere on the cartridge tray, it would mean that the bullet has been stuck at a point further inside the gun. To access those regions, the system would then proceed to lift the cartridge tray, once the ammunition belt has been removed.

3.4.5 Cartridge Tray Opening

For lifting the cartridge tray, the system rotates 90 degrees in the opposite direction, about the yaw axis to revert to its original position. Then, the

external operation arm unfolds itself to revert to its original position. At this point, the three-fingered gripper positions itself towards the cartridge tray and clasps onto its outer edge. The edge is then lifted up (Figure 33) by the drive provided at the waist joint, and the grip is released once the tray has been lifted up from its closed position, thus providing access to the breech of the gun.
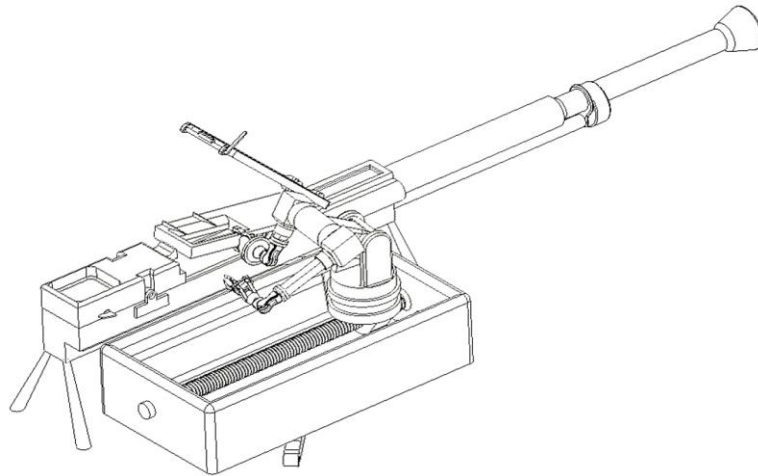
Figure 33: External Arm of the Robot lifts the cartridge tray up

3.4.6 Ammunition Extraction

If ammunition was not found anywhere directly under the cartridge tray, it would mean that it has been stuck at a point where it has either partially or completely entered the firing chamber.

To access a bullet that is seated inside the firing chamber, first the upper and lower arms carry the gripper into the space under the cartridge tray. Once it is there, the lower arm does a rotation about the roll axis to set itself horizontal to the axis of the gun, up to a maximum of 90 degrees.
From this orientation, the prismatic joint moves forward until contact is provided between the gripper and the circumference of the posterior region of the bullet, since it is seated in a horizontal position inside the chamber (Figure 34). From here, the gripper clasps and pulls at the ammunition, to try and extract it. The pull is provided by the prismatic joint retracing its path linearly. Once it is visually confirmed that the bullet has been extracted, the entire arm retraces its path of entry into the gun, and removes the bullet out.
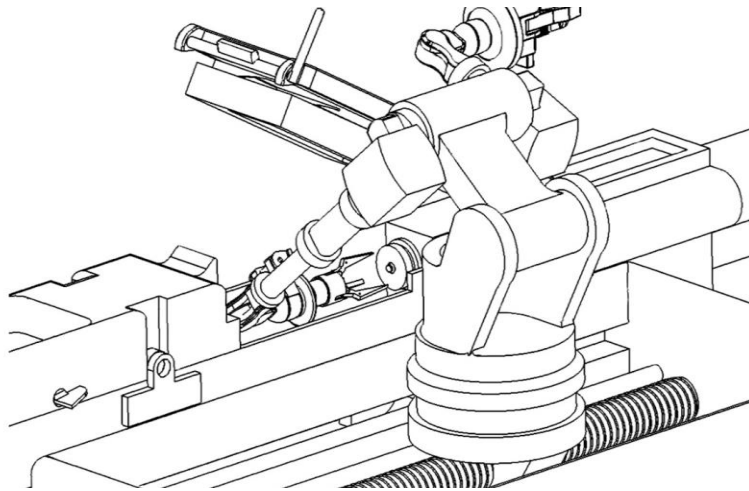
Figure 34: Internal Arm of the Robot progresses to extract the Ammunition jammed inside the firing chamber

The lower arm features a force sensor that gauges the amount of effort imparted by the arm towards retrieving the bullet. In case the maximum force exerted by the gripper is found to be insufficient, the sensor acts through a feedback mechanism to inform the processing unit of the situation. This may occur due to the gripper not being able to clasp well enough on the outer surface, or due to the bullet being wedged inside the chamber, contributing to the tightness of the fit in which it is held.

In such instances, the arm retraces its path to exit the inside of the machine gun, and equips itself with an alternative vacuum-based gripper (Figure 35). The vacuum gripper is designed with a diameter smaller than that of the bullet, and thus is able to latch onto its smooth posterior surface.

It applies suction on the bullet, and the prismatic joint retraces its path linearly outwards. Once it is visually confirmed that the bullet has been extracted, the entire arm retraces its path of entry into the gun, and removes the bullet out.
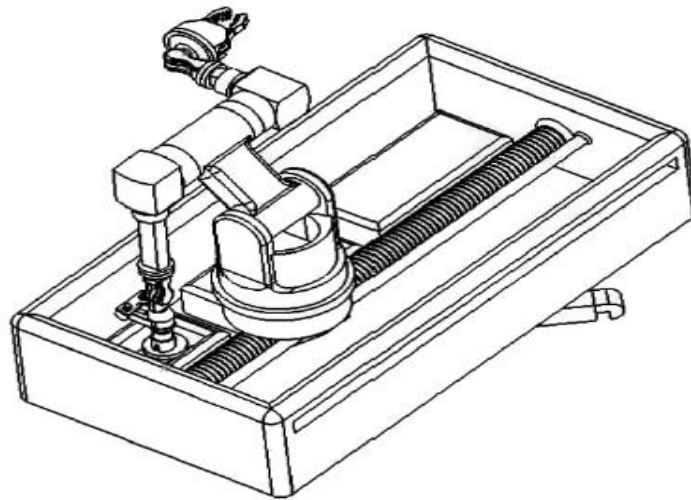
Figure 35: Internal Arm changing its end-effector at the Automatic Tool Changer

The task fulfilment is studied from the perspective of what kind of manoeuvres and to what extent should each component in the system perform, and the sequence of these should be determined. On this basis, the forces required to drive each component to be able to carry out the determined operations chronologically, are to be calculated, and on account of all these, the mechatronics systems design part of the project could be performed. Once all this is done, the 3D model of the system could be created, and design validation could be done by means of multi-body dynamics and finite element method analyses. The next stage would be to compute the work envelope of the system, which is essentially the entire field of influence of the system, or the entire volume in space where any part of the system could reach out to, calculated by means of its dimensions, degrees of freedom, and manipulating these to the maximum leverage in all possible directions. For these purposes, the Denavit-Hartenberg matrix is employed, which would yield the results on submission of appropriate input parameters in a MATLAB program. Following this, the virtual prototype of the mechanism would be fully prepared, and the final task would be the consolidation of the concept by means of a production drawing.

3.5 Design challenges

The following are the most critical challenges faced in the conceptualization and design of the robotic arm system as a solution to the given problem statement.

- A complicated work envelope with widespread critical points required that the degrees of freedom of the system be considerable in number, and pervading into all three planes, resulting in a spatial mechanism.
- Small margins of error dictated that the assembly components and the tool paths be precise because one inadvertent collision with a hazardous part of the gun could very easily trigger an explosion of the ammunition.
- The condition where strong force application was required for the external operations, and relatively weaker, but precise force application was required for the internal operations gave rise to two independent arms in the system.
- A strong base foundation structure was needed to withstand reaction forces arising due to the reaction, and it has been delivered in the form of the formidable substructure.
- Dearth of space made sure that link lengths and orientation needed to be precise and had to make the most out of the available degrees of freedom.
- The requirement for the system to be portable was fulfilled by making the entire system, and the tripod base foldable unto themselves, which would then be enclosed by the substructure unit which can be doubled over to act as a suitcase to transport the system safely into and out of the battlefield.

CHAPTER 4

KINEMATICS OF ROBOTIC ARM

Kinematics is the study of geometry in motion, restricted to pure geometrical description of motion, by defining position, orientation of any given body and their time-based derivatives. The first step towards simulating the realistic motion of the links comprising the robot arm system is to understand its capability to translate and rotate in three-dimensional space. The links of the robot are modelled as rigid bodies and its joints are assumed to provide pure rotation or translation.

For a robot to perform a specific task, the position and orientation of the end-effector, i.e., its pose or configuration, relative to the base should be in the ideal manner. Thus, the focal point of the end effector becomes the point of interest while computing the kinematics of a robotic manipulator. Robot kinematics studies the relationship between the dimensions and connectivity of kinematic chains and the position, velocity and acceleration of each of the links in the robotic system, in order to plan and control movement, and to compute actuator forces and torques.

4.1 Denavit-Hartenberg (DH) Methodology

To use a computer software to simulate the motion of a robot, we need to establish the corresponding robotic object first. In robotics, a manipulator is usually seen as a connecting rod consisting of a series of joints. In mechanical engineering, the Denavit–Hartenberg (DH) parameters are the four parameters associated with a particular convention for attaching reference frames to the links of a spatial kinematic chain, or robot manipulator. Jacques Denavit and Richard Hartenberg in 1995 proposed a method for establishing the satellite coordinate system for each member of the joint chain, and then using the 4*4 matrix to describe the spatial relations of the adjacent two connecting rods. In order to establish the kinematics equation of the robot, the position and posture of the terminal actuator relative to the base coordinate system can be deduced finally, usually called the DH parameter method.

4.1.1. DH Frame Assignment
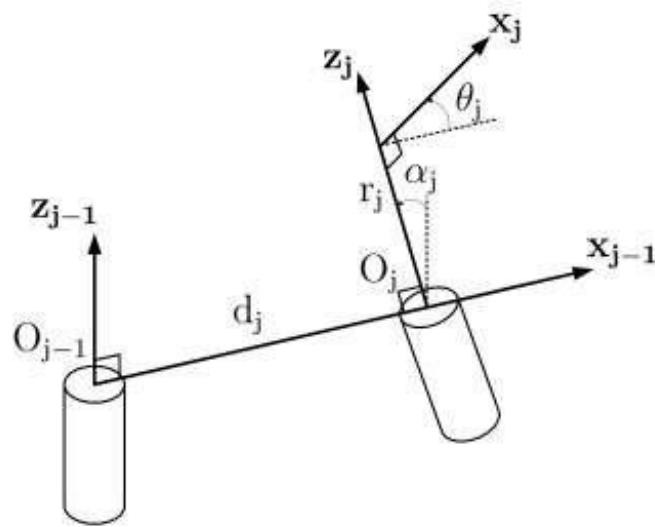
Figure 36: DH Parameters description

A frame of reference is a coordinate system which helps to enumerate distances between bodies and the nature of motion of a body using positional references. Frame of reference becomes critical in a multi-link mechanism because each link needs to be analysed with respect to its immediate neighbours, as well as with a global frame for perfect validation. Another point of note is that defining a frame for each joint would help to view the joint in terms of a standard point, say the point of contact with an adjoining member, or the centre of mass. This helps to simplify kinematic studies, akin to how a free body diagram is used to quantify a problem. The movement of an entire link can be studied as the translation and rotation of a point mass from one point to another in space.

There are certain rules to be followed while assigning frames of reference to each link as per the DH convention (Figure 36). The steps are to be followed in order for each joint, whence it is considered as the "ith" joint.
- o The joint axes, as in the axis of motion for each joint, are to be identified, and a line is to be drawn along said axis. This axis is essentially a line along the direction of translation for a prismatic joint, and a line along the axis of rotation for a revolute joint. This line is considered as the Z_i axis, and is the first step towards building up the Cartesian coordinate system for each joint.
- o The origin of the frame of the ith axis is to be set at the:

- Point of intersection of ith and $i+1^{th}$ joint axes, in the case of the Z axes of the current joint and the succeeding joint being mutually perpendicular (or)
- Start of the common perpendicular between the ith and $i+1^{th}$ joint axes, on the side of the current joint, in the case of the Z axes of the current joint and the succeeding joint being parallel to each other.

o The X_i axis is to be assigned depending on the orientation between the Z axes of the current and succeeding joints as:
- Normal to the plane containing the two axes, in the case of the Z axes of the current joint and the succeeding joint being mutually perpendicular (or)
- Along the common perpendicular, in the case of the Z axes of the current joint and the succeeding joint being parallel to each other.

o Once the aforementioned steps are done, the frames of reference are essentially defined, because the remaining Y_i axes are to be assigned by applying Right Hand Rule to each joint, and finding the direction.

o As all the previous steps are done, the only thing left would be an assignment for the universal frame. This is assigned outside of the body of the robot, to be set as a standard for defining all the movements independently. This is particularly useful when calculating relations between the end effector and the payload. The universal frame axes could be assigned to mimic the axis directions of the first link, for simplicity. Figure 37 contains an image of the frame assignments and the joint axes, calculated for our mechanism using the DH methodology.
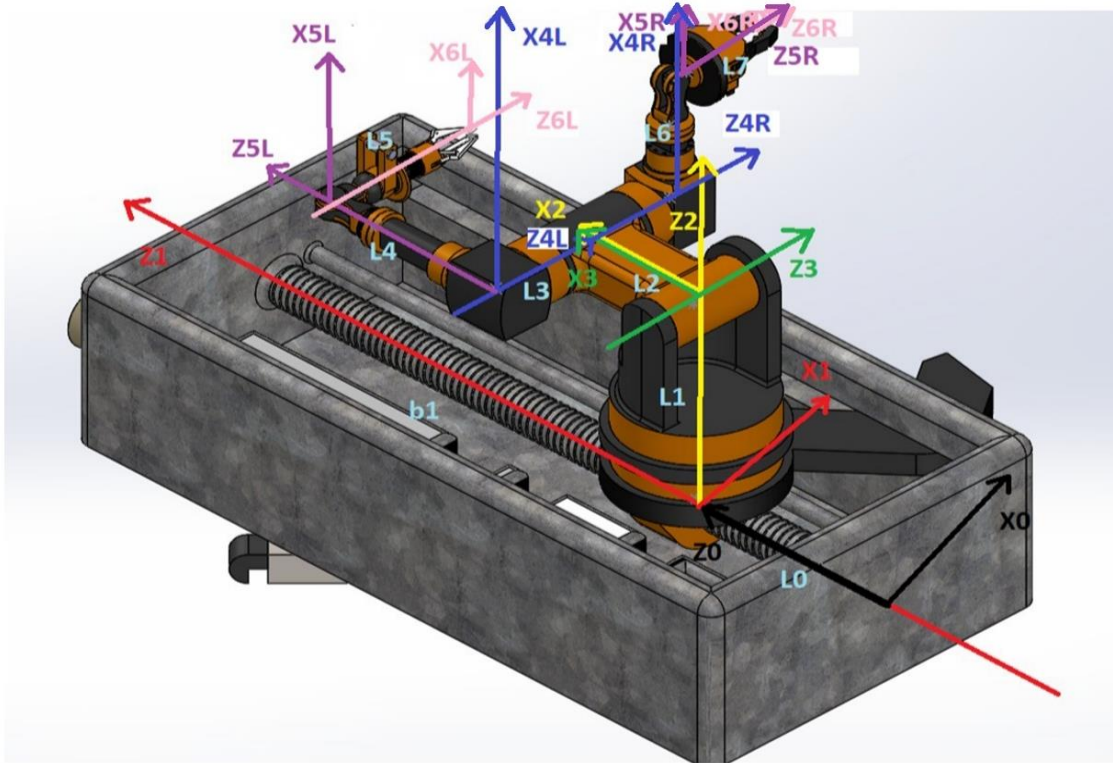
Figure 37: The model with the DH Frames of reference assigned

4.1.2 DH Parameters

On completion of the DH frame assignment, the next step is to evaluate the DH parameters, which are basically measures of a particular nature that help to describe the properties of a robotic arm manipulator that are critical towards understanding its physical features and motion capabilities, such as dimensions, joint positions, joint orientations, and degrees of freedom.

The following four transformation parameters are known as DH parameters:

- Twist Angle: α_{i-1}
- Link Length: a_{i-1}
- Joint Angle: θ_i
- Joint Offset: d_i
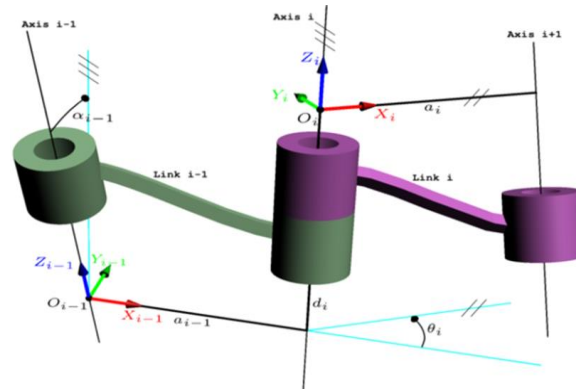
Figure 38: DH Modified convention

There are two conventions, either of which could be applied while calculating DH parameters. There is a standard or distal type of convention, which requires that all values be calculated for the current joint with respect to the previous joint. Another convention is the modified or proximal variant, which requires that all values be calculated for the current joint with respect to the next joint in sequence. The modified convention is being followed in this work, which is depicted in Figure 38.

Standard convention (Distal Variant)

α_i : the angle from z_{i-1} to z_i axis measured about x_i

a_i : the distance from z_{i-1} to z_i measured along x_i

θ_i : the angle from x_{i-1} to x_i measured about z_{i-1}

d_i : the distance from x_{i-1} to x_i measured along z_{i-1}

Modified convention (Proximal Variant)

α_{i-1} : the angle from z_{i-1} to z_i axis measured about x_{i-1}

a_{i-1} : the distance from z_{i-1} to z_i measured along x_{i-1}

θ_i : the angle from x_{i-1} to x_i measured about z_i

d_i : the distance from x_{i-1} to x_i measured along z_i

The calculated DH parameters can be represented in a DH Table. Our robotic arm system would require two separate DH tables, one for the operation of each arm, since both of them function independent of each other. Since the DH parameters are calculated with two successive links, the links on one arm would not come into the calculations of the links on the other. Each row of the table contains the four parameters pertaining to each joint, taken as sets of two links at a time. The "i" values

take up values from 1 to 6, since there are 6 joints in each kinematic chain, or arm. Of the 6, only 5 are motile joints, while the sixth joint is fixed, and is introduced as an element only to bring in the geometry of the end effector into the calculations, while computing forward and inverse kinematics. Since the terminal joint is basically considered the end effector, it would be erroneous to leave out the 6th joint from the table, because the end point of the fifth joint is not the part which needs to interact with the workspace. Firstly, the DH tables are filled up with values taken from the frame assignment stage, with "Theta" denoting the angle of rotation of each revolute joint, "B" denoting the distance of translation of the prismatic joint, adhering to sign conventions (Tables 2 and 3). The "L" values correspond to the dimensions of each link, in terms of distance between frames, which are measured from the CAD model, which has been designed based on the link length requirements entailed by the operation. Once this is done, the L values are substituted and the final DH tables are arrived at (Tables 4 and 5). Theta and B values remain as such because they are variables, and would be filled only as and when a transformation occurs between the links in question.

Table 2: DH Table for Left Arm

Joint	a(i-1)	alpha(i-1)	d(i)	theta(i)
0->1	0	0	L0 + B1	0
1->2	0	90	L1	90 + Theta2
2->3	0	90	0	Theta3
3->4	L2	0	-L3	90 + Theta4L
4->5	0	90	L4	Theta5L
5->6	0	-90	L5	0

Table 3: DH Table for Right Arm

Joint	a(i-1)	alpha(i-1)	d(i)	theta(i)
0->1	0	0	L0 + B1	0
1->2	0	90	L1	90 + Theta2
2->3	0	90	0	Theta3
3->4	L2	0	L3	90 + Theta4R
4->5	L6	0	0	Theta5R
5->6	0	0	L7	0

Table 4: Updated DH Table for Left Arm

Joint	a(i-1)	alpha(i-1)	d(i)	theta(i)
0->1	0	0	75 + B1	0
1->2	0	90	135	90 + Theta2
2->3	0	90	0	Theta3
3->4	82	0	-62	90 + Theta4L
4->5	0	90	120	Theta5L
5->6	0	-90	100	0

Table 5: Updated DH Table for Right Arm

Joint	a(i-1)	alpha(i-1)	d(i)	theta(i)
0->1	0	0	75 + B1	0
1->2	0	90	135	90 + Theta2
2->3	0	90	0	Theta3
3->4	82	0	62	90 + Theta4R
4->5	72	0	0	Theta5R
5->6	0	0	55	0

4.1.3 DH Transformation Matrix

Once the DH parameters have been calculated, we find the homogeneous transformation matrices (also known as the DH matrices) by plugging the parameter values into a 4x4 matrix, which is the homogeneous transformation matrix, T^{n-1}_n for joint n (i.e. the transformation from frame n-1 to frame n). This contains the information of the possible relative motion between one joint, or frame with respect to the preceding one. The transformation matrix comprises the rotation matrix and the position vector, which describe the differences in orientation and position respectively between the two states, thus giving a comprehensive description of the link's movement. The DH parameters are input into the transformation matrix as shown in Figure 39. R^{n-1}_n is the 3×3 sub-matrix in the upper left corner of T, that represents the rotation from frame n-1 (e.g. frame 0) to frame n (e.g. frame 1), and P^{n-1}_n is the 3×1 sub-matrix in the upper right that represents the translation (or displacement) from frame n-1 to frame n, in terms of unit vector components.

$$^A_B R = \begin{bmatrix} ^A\hat{X}_B & ^A\hat{Y}_B & ^A\hat{Z}_B \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

Figure 39(a): Rotation Matrix

$$^A P = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}$$

Figure 39(b): Position Matrix

A1	A2	A3	-(Px.A1 + Py.A2 + Pz.A3)
B1	B2	B3	-(Px.B1 + Py.B2 + Pz.B3)
C1	C2	C3	-(Px.C1 + Py.C2 + Pz.C3)
0	0	0	1

Figure 39(c): DH Transformation matrix

Figure 39: Elements of the DH Transformation matrix

The DH transformation matrix is obtained using the formula given, which makes use of the DH table values. It can be understood from this that the description of the link motions is used by the matrix to convey how any motion would affect the link's state. For the application, the state of the end effector is of more importance than those of other links. The end effector is what needs to reach the critical points to perform the actions required, and the positions of the remaining links can be wherever is best suited to facilitate that. Also, it is efficient for calculations and monitoring if transformations of all joints are taken with respect to a common frame of reference, also known as the base frame. Therefore, the final DH transformation matrix pertains to the end effector with respect to the base frame. To find out the transformation matrix $T^{n-2}{}_n$ of a joint n with respect to another joint n-2, the transformation matrix $T^{n-2}{}_{n-1}$ has to be multiplied with the transformation matrix, $T^{n-1}{}_n$. Similarly, to calculate the transformation matrix between the end effector and the zero frame, the transformation matrices of each link with respect to the previous, starting from the first link, right up to the end effector, has to be multiplied in order, as shown below.

$$
{}^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$T^0{}_6 = T^0{}_1 \ \text{x} \ T^1{}_2 \ \text{x} \ T^2{}_3 \ \text{x} \ T^3{}_4 \ \text{x} \ T^4{}_5 \ \text{x} \ T^5{}_6$$

4.2 Forward Kinematics

The kinematics has to be studied in terms of forward computation, and inverse computation, the relationship between which is illustrated in Figure 40.
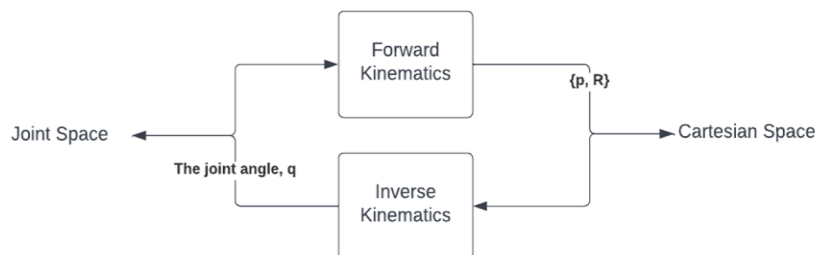

Figure 40: Forward and Inverse Kinematics relation

The information that the final DH transformation matrix contains about the motion of the end effector is essentially a by-product of the individual motions of all the joints. Even though the individual transformation matrices describe the motion of each joint in the cartesian coordinate system, practically, the joints can be controlled by means of their angle of rotation (for revolute joints) and distance of translation (for prismatic joints). The forward and inverse kinematics deal with these two outputs. Forward kinematics helps us to find the position and orientation of the manipulator tip relative to the base as a function of joint variables. Inverse kinematics finds the set of joint variables which give the configuration, given the desired position and rotation angles of the tip relative to the base. All the calculations relevant to these studies are done using MATLAB in this work. Since only one kinematic chain could be simulated in an instance by MATLAB, all plots shown further would portray only either one arm. The plot feature also does not support viewing from a particular desired angle; therefore, all images of arms would contain an inverted view.

Forward kinematics is used to establish the transformation matrix of the end effector with respect to the base, from the starting position of the robot arm, if the starting joint variables are fed into the program. In the obtained matrix, the position vector corresponds to the point in space that the terminal point of the end effector lies on. Thus, by changing these values alone, the matrix becomes the input to be used for inverse kinematics. The desired orientation for operation can be found using forward kinematics, and the same could be filled in as the rotation component of the input for inverse kinematics.

4.2.1 Operational Ranges

The operational range is the complete range a robotic arm needs to cover for its tasks. These values were established using forward kinematics by feeding in joint variables and picking out the minimum and maximum points that should lie within the reach of the end effector. Such joint variable extremities and robot arm positions in such extremities (Figures 41 and 42) are documented below in Tables 6 and 7.

Table 6: Operational Range of Left Arm

Joint Variable	Range of Values (mm/rad)	Home Position Value (mm/rad)
B 1	0 to 335	0
Theta 2	0 to pi/2	pi/2
Theta 3	-pi/2 to pi/2	0
Theta 4L	0 to pi	pi/2
Theta 5L	-pi/2 to pi/2	0

Table 7: Operational Range of Right Arm

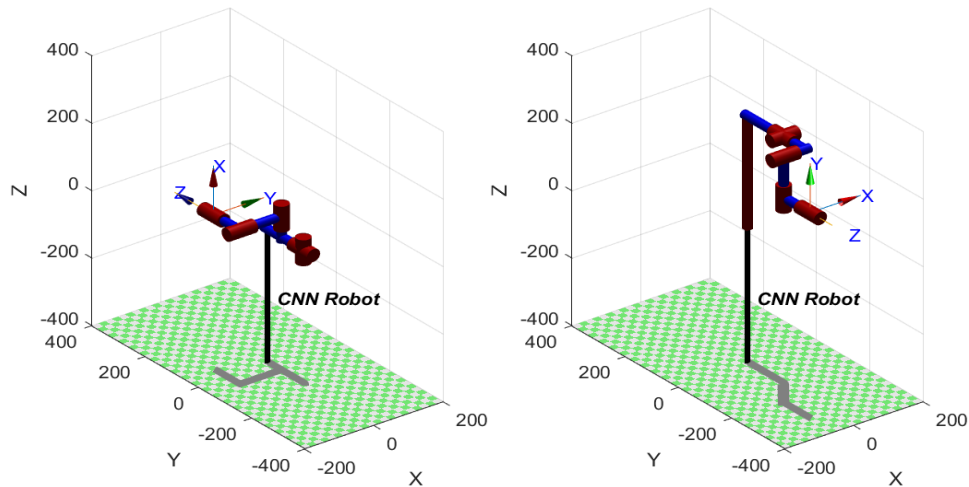| Joint Variable	Range of Values (mm/rad)	Home Position Value (mm/rad)
B 1	0 to 335	0
Theta 2	0 to pi/2	pi/2
Theta 3	-pi/2 to pi/2	0
Theta 4R	-pi/2 to pi	pi/2
Theta 5R	0 to pi	0

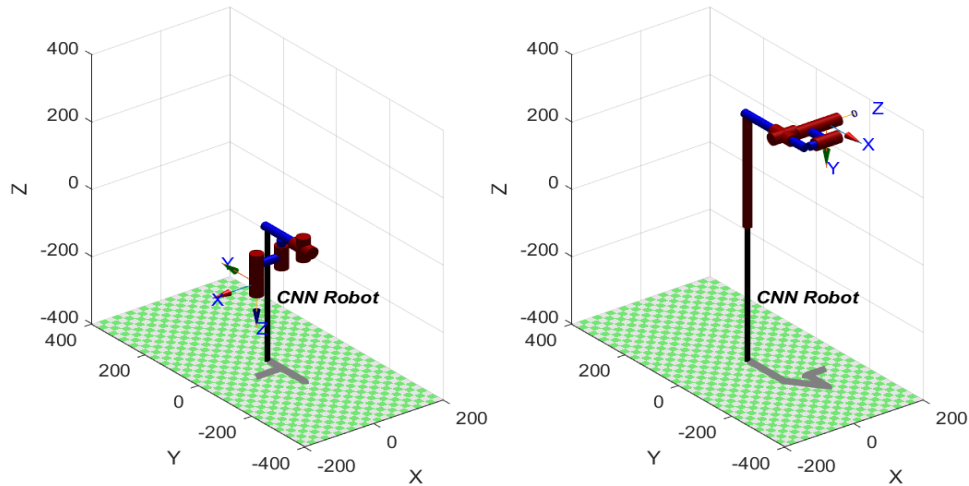Figure 41: Extreme Positions of the Left Arm

Figure 42: Extreme Positions of the Right Arm

4.2.2 Forward Kinematics Results

The forward kinematics computation was run to identify the transformation matrix of the end effector at the starting position and orientation of the operation cycle. The matrices and simulated images of the robot arm at the initial position are furnished below in Figures 43 and 44.

4.2.2.1 Forward Kinematics of the Internal Arm:

Transformation matrix:

$$\begin{matrix} 1 & 0 & 0 & -62 \\ 0 & 0 & 1 & -35 \\ 0 & -1 & 0 & 202 \\ 0 & 0 & 0 & 1 \end{matrix}$$
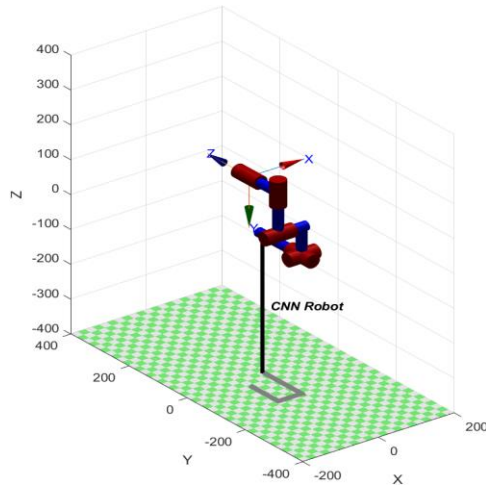
Figure 43: Initial Position of the Internal Arm

Refer Appendices A and C

4.2.2.2 Forward Kinematics of the External Arm:

Transformation matrix:

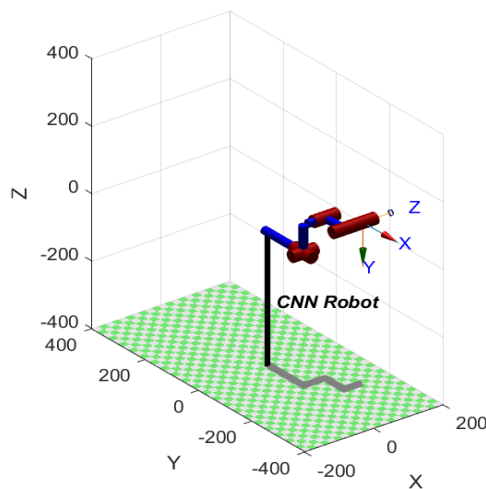| 0	0	1	117
-1	0	0	-207
0	-1	0	82
0	0	0	1

Figure 44: Initial Position of the External Arm

Refer Appendices B and D

4.3 Inverse Kinematics

Inverse kinematics is the mathematical process of calculating the variable joint parameters needed to place the end of a kinematic chain, such as a robot manipulator in a given position and orientation relative to the start of the chain. Given joint parameters, the position and orientation of the chain's end, can typically be calculated directly using multiple applications of trigonometric formulas, in forward kinematics. However, the inverse calculation is, in general, much more challenging. Especially since the robotic arm contains a high number of links, and is a spatial mechanism.

4.3.1 Critical Points

These are the points that the end-effector has to reach in the operational range. These points have been identified from the CAD model of the workspace involving the machine gun and ammunition. The critical points pertain to each operation that the arm is required to perform, and are chosen strategically to ensure that the end effector reaching the point is tantamount to it being capable of performing the relevant task. The critical points and the pertinent orientation for working at those points (as determined through forward kinematics), serve as the input for inverse kinematics. The joint variables required to bring about the motion towards reaching the critical points is calculated based on the input transformation matrix. Inverse kinematics results are more important than those of forward kinematics, because the joint displacements are then used to find out joint velocities, accelerations, forces, torques and powers. More pertinently, the fact that these parameters are obtained as functions of joint variables, is significant because each joint is driven by a motor each, and the motor could only apply the required torque for each joint as calculated for itself, and the Cartesian variables would not be of use for this purpose. The critical points chosen to simulate the operation of the arm system, are represented visually with the aid of an image of the machine gun in Figure 45, in addition to their coordinates with respect to the base frame in Table 8.

There is a total of seven critical points, but for calculations only six are chosen because the point denoting the cocking pin does not entail any kinematic movement to be reached. As mentioned previously, the base part of the arm accessing the

cocking pin is supposed to be fixed while butting up against the pin. Therefore, this operation comes of interest to us further only during the dynamic calculation involving payload. The six other critical points however, are divided three each among the external and internal operation arms (RE1, RE2, RE3, LE1, LE2, LE3).
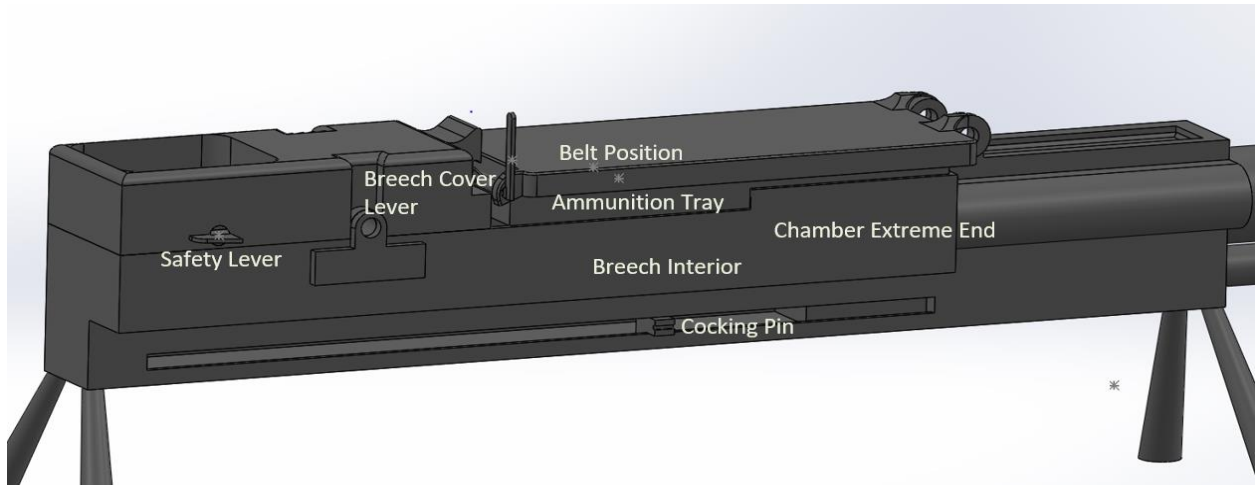

Figure 45: The Critical points on the Machine Gun

Table 8: Coordinate Values of all the Critical Points

Point Significance	Notation	X – Coordinate	Y – Coordinate	Z – Coordinate
Cocking Pin	-	160	7	155
Safety Lever	RE1	132	76	395
Breech Cover Lever	RE2	125	105	250
Belt Position	LE1	166	89	190
Ammunition Tray	RE3	138	89	190
Breech Interior	LE2	40	89	190
Chamber Extreme End	LE3	191	60	81

4.3.2 Inverse Kinematics Results

Narrowing down our motion to the specific points we need to reach with our robot, i.e. the critical points, and having found those points in the base frame of reference we had to input them into a MATLAB program to compute the joint variables required to reach those points by using inverse kinematics. The critical points are

coupled with the rotation matrix describing the desired final orientation, as calculated from forward kinematics, and input in the form of the DH transformation matrices shown below, alongside the plot depicting the robot arm in the required state at each critical point, in Figures 46, 47, 48, 49, 50 and 51. The calculated joint variables for each critical point are also displayed in Table 9.
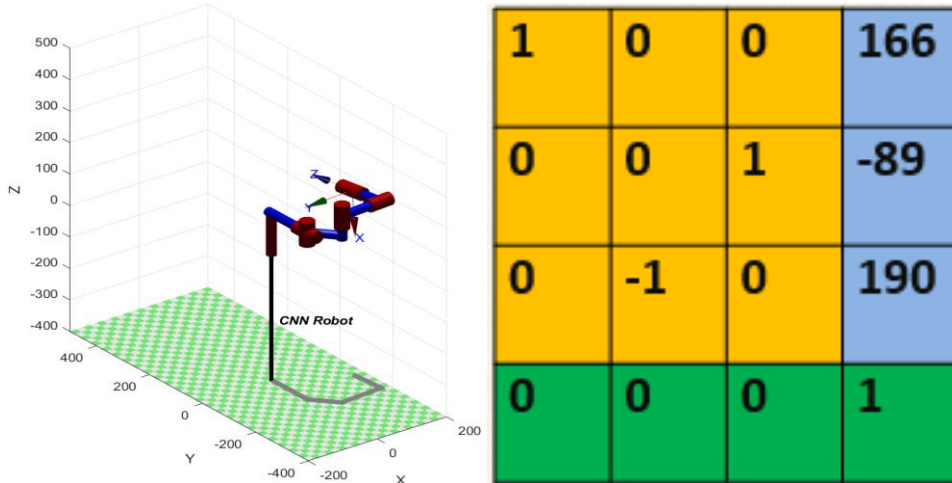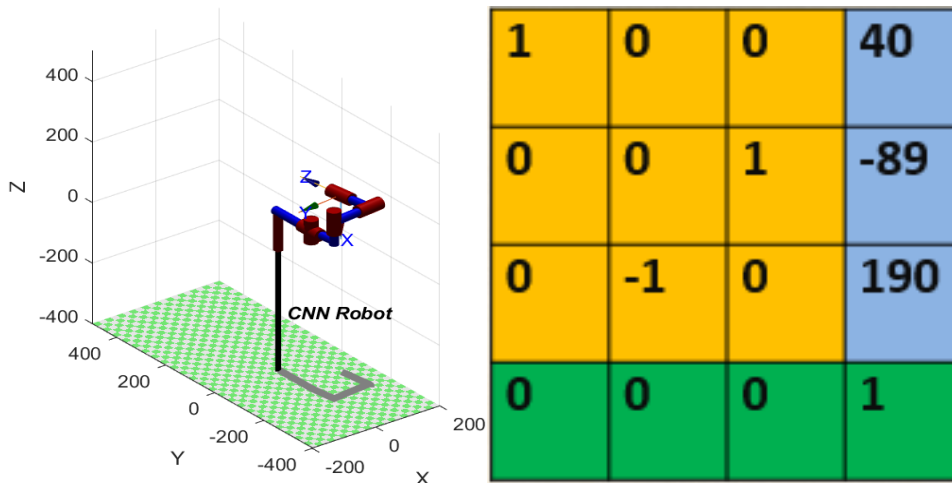

1	0	0	166
0	0	1	-89
0	-1	0	190
0	0	0	1

Figure 46: LE1

Refer Appendix E

1	0	0	40
0	0	1	-89
0	-1	0	190
0	0	0	1

Figure 47: LE2

Refer Appendix F

1	0	0	191
0	0	1	-60
0	-1	0	81
0	0	0	1
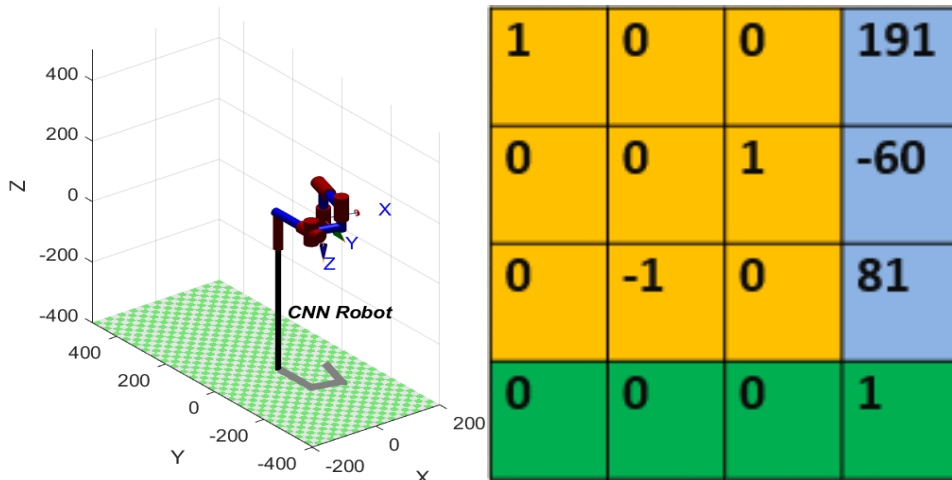
Figure 48: LE3

Refer Appendix G

0	0	1	117
-1	0	0	-76
0	-1	0	395
0	0	0	1

Figure 49: RE1

Refer Appendix H

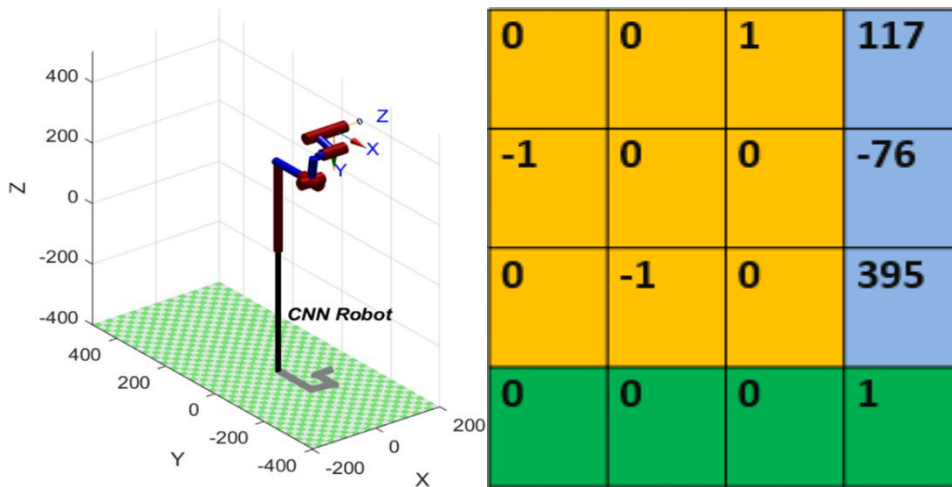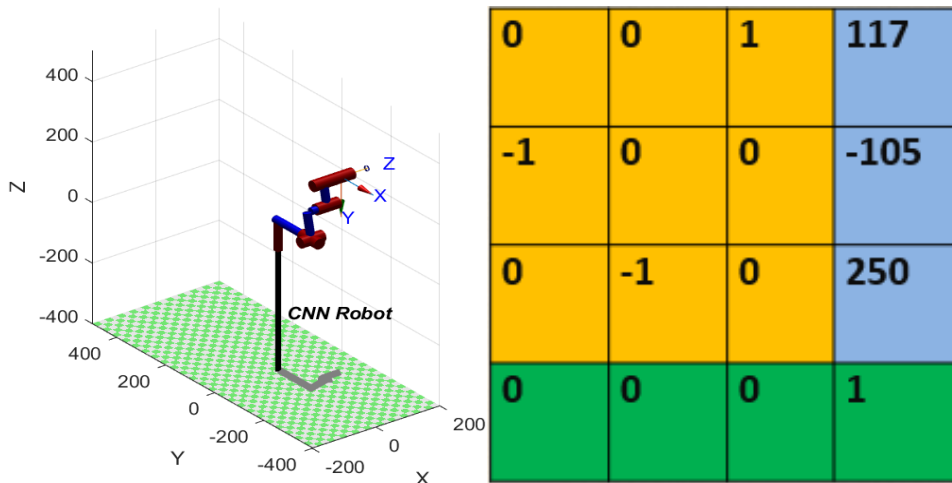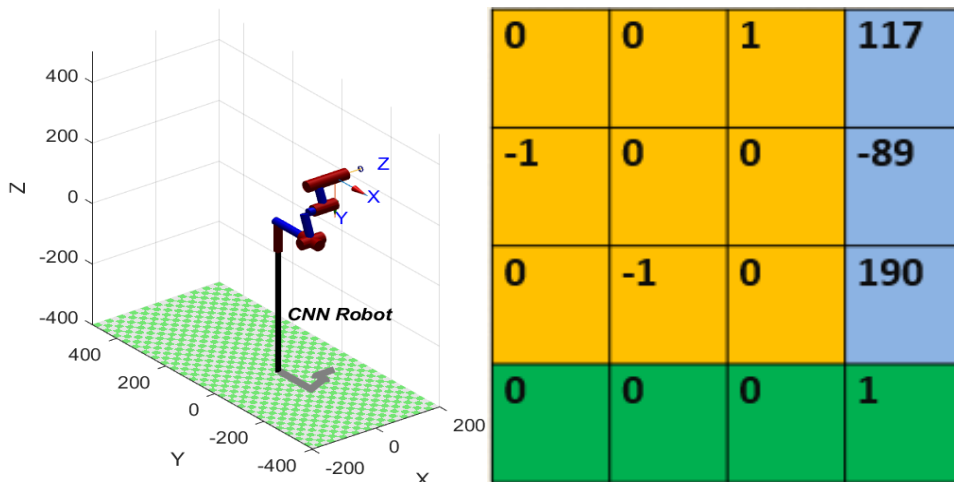|			
0	0	1	117
-1	0	0	-105
0	-1	0	250
0	0	0	1

Figure 50: RE2

Refer Appendix I

Figure 51: RE3

Refer Appendix J

Table 9: Joint Variables of the Critical Points

Point Significance	B 1	Theta 2	Theta 3	Theta 4	Theta 5
Safety Lever	135.2541	0	0.8595	0.6391	1.4985
Breech Cover Lever	98.9632	1.5708	-0.2055	0.0202	1.7561
Belt Position	135.2541	0	0.8595	0.6391	1.4985
Ammunition Tray	75.0125	1.5708	-0.9704	1.2760	1.2652
Breech Interior	128.0080	0	1.4577	0	1.5708
Chamber Extreme End	119.0151	0	0.2504	0.0214	0

4.4 Trajectory Plots

Trajectory is the path followed by the end effector of the robotic arm to reach the critical points from its home position. Trajectory planning directly affects the quality of the robot's work. The trajectory forms as the basis to arrive at the displacement, velocity and acceleration of the manipulator in motion. Calculating the expected trajectory is trajectory planning, which is based on the requirements of the task. The requirement of the trajectory planning is to control the movement speed of the manipulator during the movement and the movement space is always kept within the allowable range of each joint movement. The path traced by the robot arm in

reaching the critical points are represented with a colour coding for each operation path.

In the case of the right arm (Figure 52), the green path corresponds to LE1, prussian blue to LE2, and yellow to LE3. Coming to the left arm (Figure 53), the light blue line corresponds to LE1, dark blue to LE2 and red to LE3.
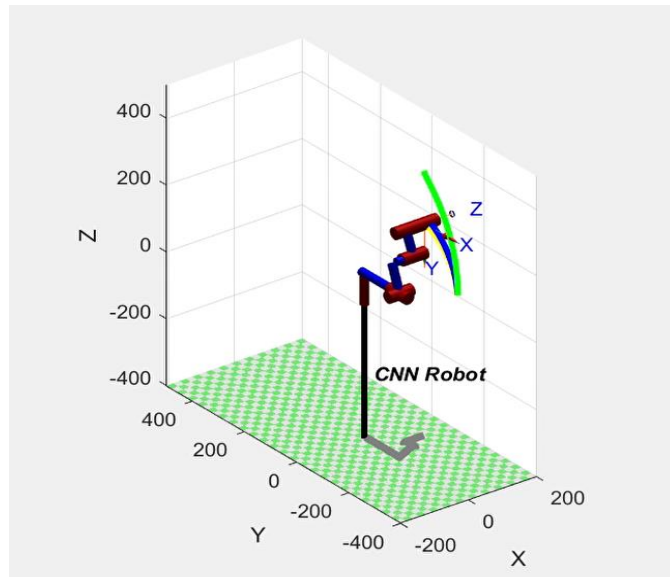

Figure 52: Trajectory of External Arm's End Effector
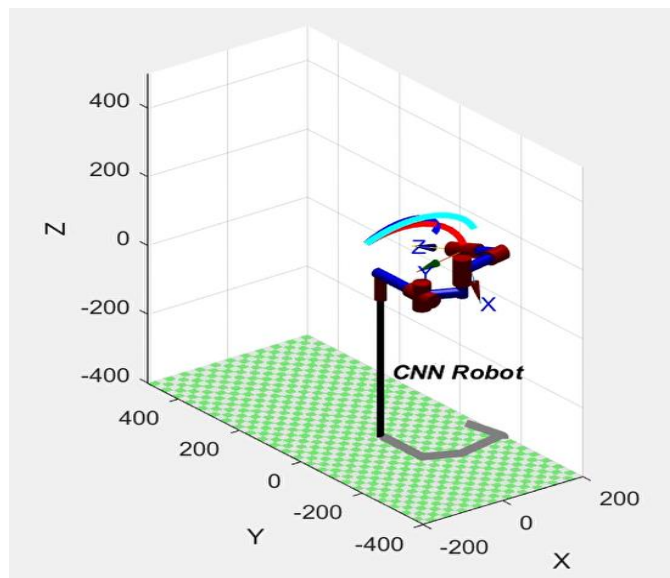
Refer Appendix L

Figure 53: Trajectory of Internal Arm's End Effector

Refer Appendix K

4.5 Joint Variables

4.5.1 Jacobian

In this section, the relationships between the joint rates and the corresponding end-effector's angular and linear velocities are presented. This mapping is described by a matrix called Jacobian, which depends on the robot's configuration. Jacobian is a multi-dimensional matrix made up of velocity vectors. The Jacobian constitutes one of the most important tools for robot characterization. In fact, it is useful for:

- finding singular configurations,
- analysing redundancy,
- determining inverse kinematics algorithms for velocity analysis,
- describing the relationship between the forces applied at the end-effector and the resulting torques at the joints, and
- deriving dynamics algorithms

The following equation in Figure 54 is the Cartesian-Joint Relation, which relates cartesian variables to joint space variables:

$$
\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}_{6 \times 1} = J_{6 \times n} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dots \\ \dot{q}_n \end{bmatrix}_{n \times 1}
$$

Figure 54: Jacobian helps relate the joint velocities to the cartesian velocities in a robotic manipulator.

4.5.2 Cartesian Displacement

Cartesian displacements are the X, Y, and Z displacement components of the end effector with respect to the base frame.

Standard application ranges were selected for fitting these equations, based on the trajectory points calculated from inverse kinematics, to smoothen the paths.
- Left x = -3.74 t3 + 29.27 t2 – 4.13 t - 66.49
- Left y = -0.81 t3 - 6.99 t2 – 3.92 t - 35.02
- Left z = 0.66 t3 – 11.01 t2 – 40.84 t + 191.31
- Right x = 117
- Right y = -2.86 t3 + 26.64 t2 – 36.46 t + 201.15
- Right z = -4.67 t3 + 35.59t2 – 5.46 t + 73.51

4.5.3 Joint Displacement

Joint displacements are the change in individual joint variables (B/Theta). They are derived from the cartesian displacements using the cartesian-joint relation by using the jacobian. The joint space displacements of the linkages in this mechanism can be analysed by the following equations.

Left1: -1.96 t^3 + 17.61 t^2 – 13.09 t + 1.23
Left2: 0.02 t^3 – 0.21 t^2 + 0.15 t + 1.56
Left3: -0.01 t^3 + 0.11 t^2 – 0.08 t + 0.01
Left4: 0.01 t^3 – 0.12 t^2 + 0.09 t + 1.56
Left5: 0.01 t^3 – 0.01 t^2 + 0.01 t + 1.57

Right1: -4.25 t^3 + 38.25 t^2 – 28.43 t + 2.67
Right2: 1.57
Right3: -0.01 t^3 + 0.02 t^2 – 0.01 t + 0.01
Right4: 0.04 t^3 – 0.39 t^2 + 0.29 t + 1.54
Right5: -0.04 t^3 + 0.37 t^2 – 0.28 t + 0.03

4.5.4 Joint Velocity

The joint velocities are the first derivatives of the individual joint displacement equations found above.

Left1: -5.88 t² + 35.22 t - 13.09
Left2: 0.06 t² – 0.42 t + 0.15
Left3: -0.03 t² + 0.22 t – 0.08
Left4: 0.03 t² – 0.24 t + 0.09
Left5: 0.03 t² – 0.02 t + 0.01

Right1: -12.75 t² + 76.5 t – 28.43
Right2: 0
Right3: -0.03 t² + 0.04 t – 0.01
Right4: 0.12 t² – 0.78 t + 0.29
Right5: -0.12 t² + 0.74 t – 0.28

4.5.5 Joint Acceleration

The joint accelerations are the first derivatives of the individual joint velocity equations and second derivatives of the individual joint displacement equations found above.

Left1: 11.76 t + 35.22
Left2: 0.12 t – 0.42
Left3: -0.06 t + 0.22
Left4: 0.06 t – 0.24
Left5: 0.06 t – 0.02

Right1: -25.5 t + 76.5
Right2: 0
Right3: -0.06 t + 0.04
Right4: 0.24 t – 0.78
Right5: -0.24 t + 0.74

All the calculated joint variables are plotted against the operation cycle time, with pertinence to each individual joint in Figures 55 and 56. The velocity has been given a smooth profile, with slow variation in it towards the beginning and the end, to maintain operational efficiency. The plots of "Right-2" joint variables remain a flat line since the second joint, in the context of the right arm does not entail any change in its position, but it cannot be removed from the structure due to its relevance when it comes to the left arm movement.
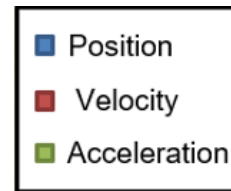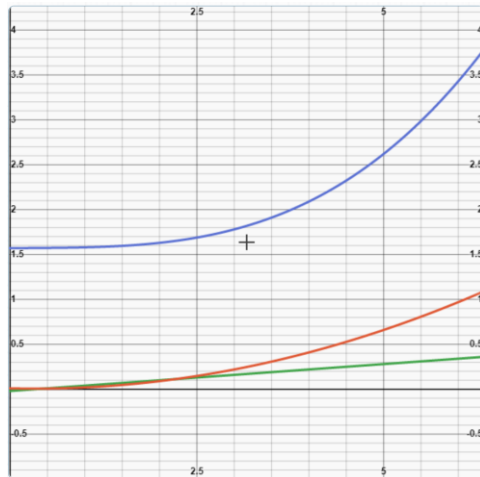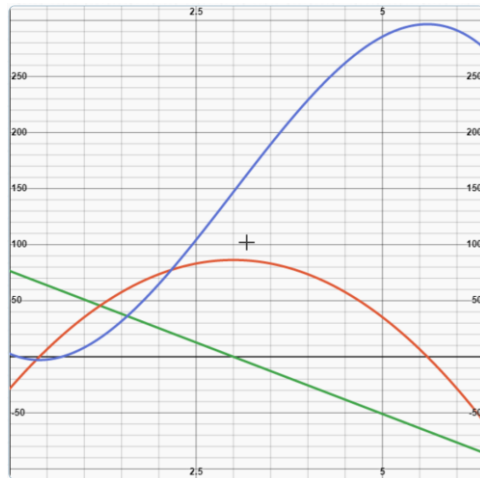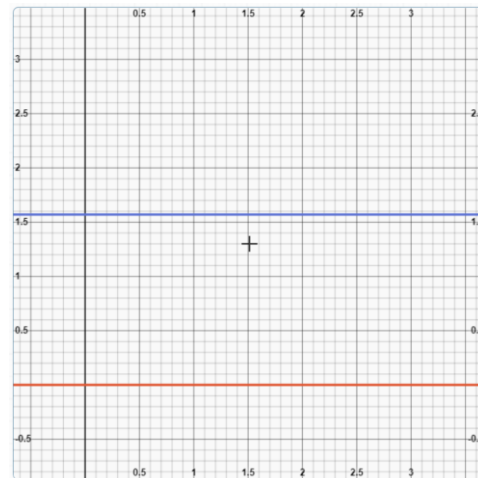

(a) LEFT - 1
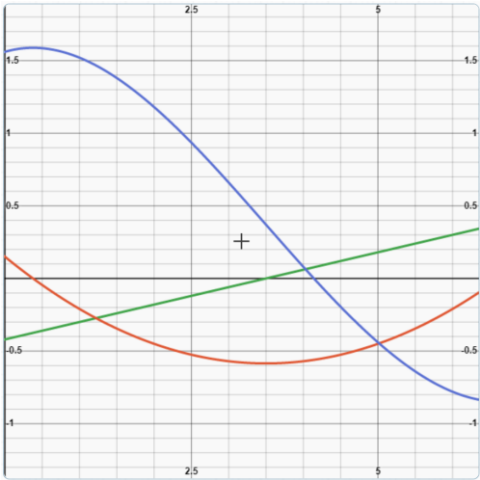
(b) LEFT - 2
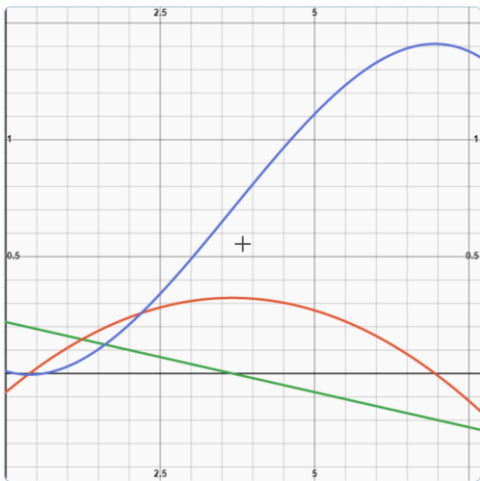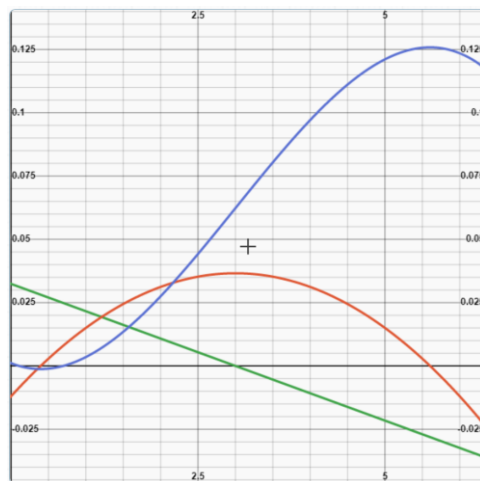
(c) LEFT - 3

(d) LEFT - 4
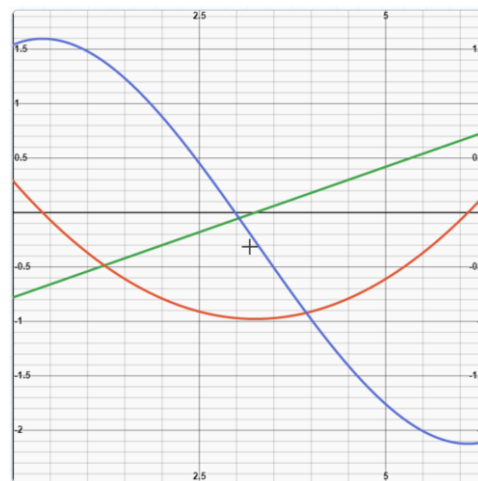
(e) LEFT - 5

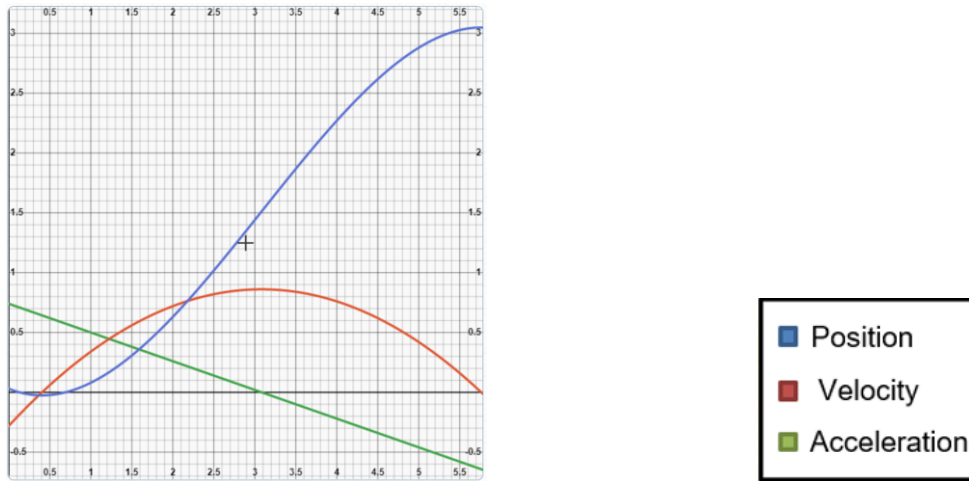Fig 55: Joint Variables of Internal (Left) Arm

(a) RIGHT - 1

(b) RIGHT – 2

(c) RIGHT - 3

(d) RIGHT - 4

(e) RIGHT - 5

Figure 56: Joint Variables of External (Right) Arm

4.6 Workspace

In robotics, the Workspace is defined as the set of points that can be reached by its end-effector. Put in other words, the workspace of a robot is the space in which the mechanism is working. Several authors also refer to workspace as work volume or work envelope. In the computation of a robot workspace, what is most important is its shape and volume (dimensions and structure). Both aspects have a significant importance due to their impact on the design and manipulability of the robot.

For using a robot, the exact knowledge about the shape, dimensions and structure of its workspace is important since:

- The shape is important for the definition of the environment where the robot will work.
- The dimensions are important for determining the reach of the end-effector.
- The structure of workspace is important for assuring kinematic characteristics of the robot which are in relation with the interactions of the robot to its environment.

The workspace of this system has been computed and represented through plots as follows. To arrive at these, the forward kinematics was utilized, being run in a loop

carrying joint variables from the minimum and maximum extremities discussed earlier, in small increments for better visualization. They are shown as two-dimensional plots in Figures 57 and 58, to better understand the dominant planes of motion of each arm, which would be difficult to discern from a three-dimensional plot.
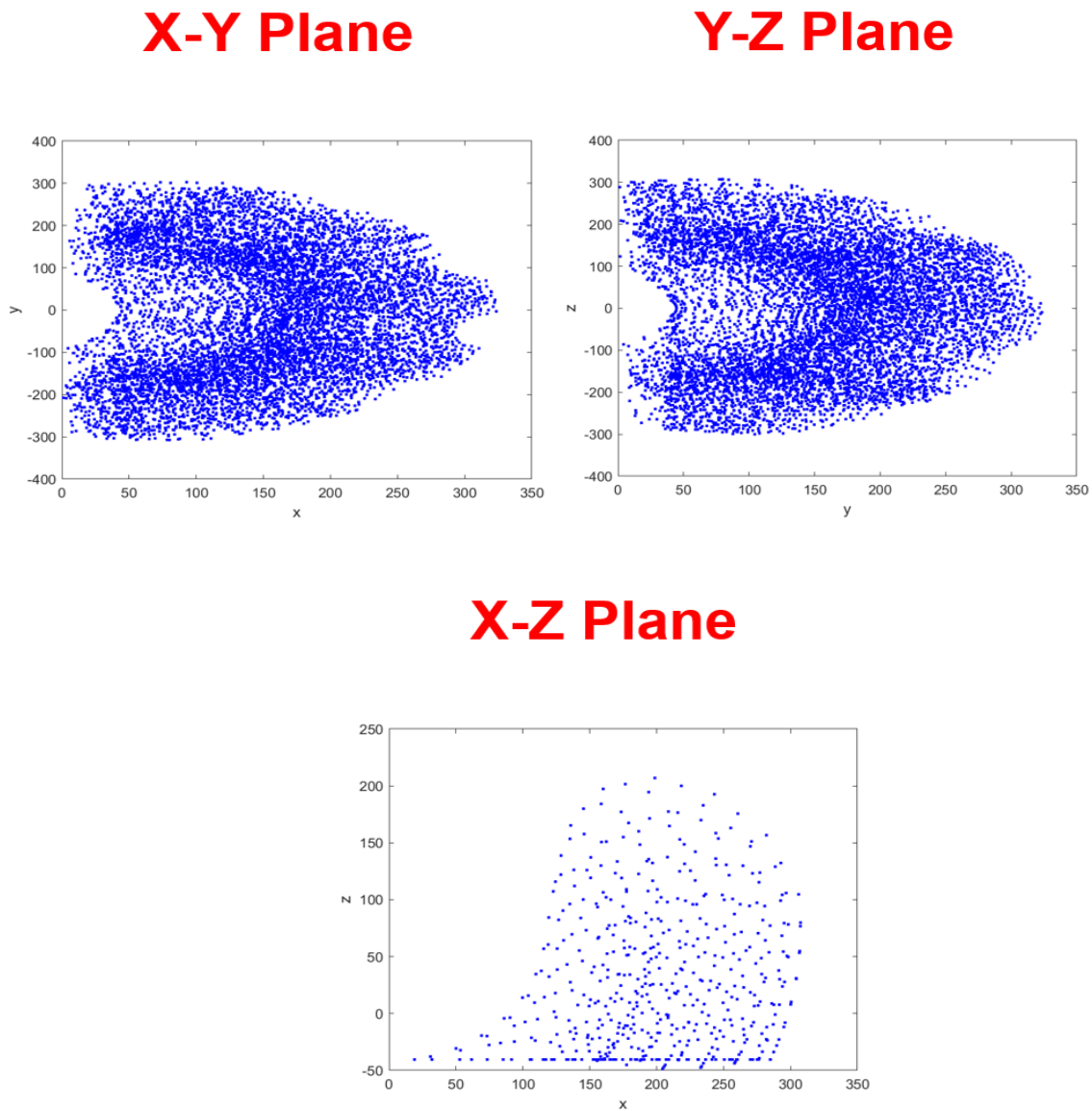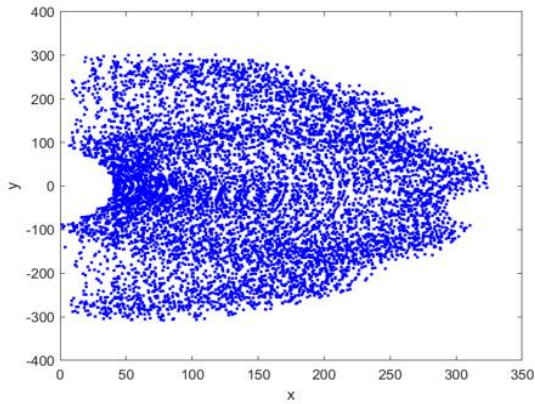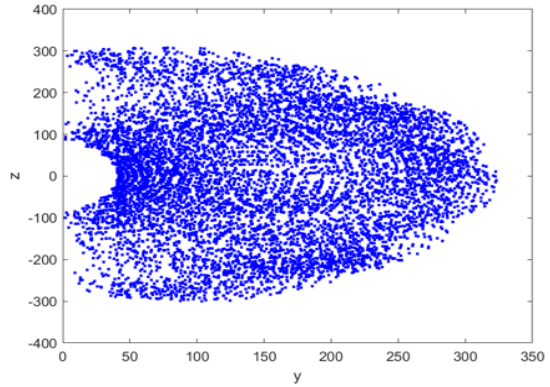

Figure 57: Workspace of the Internal Arm

Refer Appendix N

X-Y Plane

Y-Z Plane

X-Z Plane
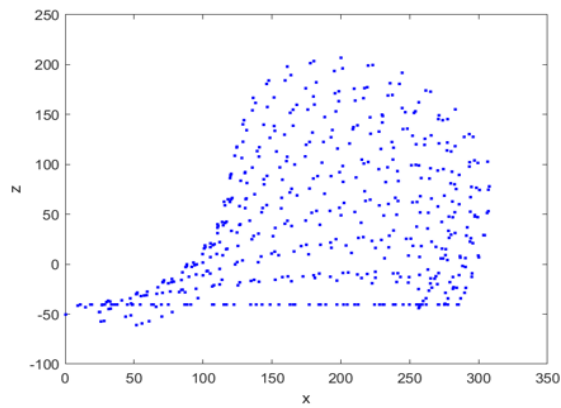
Figure 58: Workspace of the External Arm

Refer Appendix O

4.7 Singularity

Singularity is a point in space where the robot loses one or more degrees of freedom (DOF). Here, it is impossible to move the end-effector in a particular direction.

There are two types of singularities:
- Workspace-Boundary singularities, where the mechanism is either fully stretched out or folded back on itself.
- Workspace-Interior singularities, where two or more joint axes line up with each other.

For a robotic manipulator at a particular point, the determinant of the Jacobian of the instantaneous transformation matrix with respect to the ground shows the presence of singularities.

If det [J(θ)] = 0, singularity exists
If det [J(ϑ)] ≠ 0, singularity does not exist

Our robotic system has 5508 number of singularity points in its workspace and its trajectories have been made sure to not encounter any of them.

Refer Appendices M and P for the MATLAB programs for Jacobian and Singularities.

CHAPTER 5

DYNAMICS OF ROBOTIC ARM

Dynamics is the branch of classical mechanics that is concerned with the study of forces and their effects on motion. Dynamics relates how actuating forces and torques affect the motion. Dynamic modelling is vital for simulation and implementation of control algorithms. MSC ADAMS software is used for the dynamic analysis of the robot model. The basic prerequisites include geometric accuracy and the validity of constraints which were done in SolidWorks. The material of the robot parts (as this determines the mass and other basic dynamic parameters) is also equally significant. The Newton-Lagrange methodology used here relies on the fundamental lemma of calculus of variations, in order to determine inverse dynamics.

5.1 Multi-Body Dynamics

Multibody system is a system consisting of multiple parts and components attached together with joints and connectors. The motion of the subsystem is kinematically constrained due to different types of joints. Multibody simulation is a field of applied mechanics researching multibody systems and their dynamics. It allows a dynamic analysis of interconnected rigid and deformable components. Several mechanical and structural systems like vehicles, space structures, robotics mechanisms and aircraft, consist of interconnected bodies that experience large translational and rotational displacements.

The software MSC ADAMS, which stands for Automatic Dynamic Analysis of Mechanical Systems, provides a robust solution to solve mechanical systems models. The software checks the model, formulates and automatically solves the equations of motion for kinematic, static, quasi-static and dynamic simulations. It is also possible to optimize the model defining the variables, constraints and design objectives.

For the dynamic analysis of any multibody system, the inertial, the constraining and any externally applied forces must be kept in equilibrium. This property is the basis for the formulation of the equations of motion, which are formulated in terms of expressions for the kinetic and potential energy of the mechanical system. So, the Euler-Lagrange equation is fundamental for the formulation of equations of motion in ADAMS/Solver, the computational engine of the software.

5.2 Euler-Lagrange Equation

In the calculus of variations and classical mechanics, the Euler–Lagrange equations constitute a system of second-order ordinary differential equations whose solutions are stationary points of the given action functional. The dynamic model of a robot can be derived in a systematic way using the concept of generalized coordinates and a scalar function called Lagrangian. The Lagrangian (L) of the dynamic system is defined as the difference between the kinetic (T) and potential energies (V) of the mechanical system under study, i.e.,

$$L = T - V$$

Note that the kinetic energy depends on both the configurations, i.e., position and orientation, and the velocity of the links of a robotic system, whereas the potential energy depends only on the configuration of the links.

Euler–Lagrange equations of motion are then given by:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \left(\frac{\partial L}{\partial q_i}\right) - Q + \sum_{k=1}^{m}\left(\frac{\partial \Phi_k}{\partial q_i}\right)\lambda_k = 0$$

where,

L = Lagrangian
q = generalised coordinates of the system
Φ = constraint function
Λ = Lagrange-multiplicators
Q = generalised external loads
m = number of constraint equations
l = 1,2,5,6 (translational and rotational components)

For multibody dynamics, the Lagrangian becomes,

$$L = \sum_{j=1}^{N} T_j - V_j$$

where T_j and V_j are the kinetic and potential energy for each part of the N parts which compose the system.

The motion of a multibody is given by,

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} + \Phi_q^T \lambda = Q$$

where q is the column matrix of generalised coordinates, Φ_q is the n×m array which couples the constraints conditions into the equation, and λ is the column matrix of m < n Lagrange multipliers. The above equation represents the implementation of the laws of Lagrangian in Adams/Solver. To solve these equations the Newton method is used. This method leads to robust and fast simulations.

5.3 Building up the problem in MSC ADAMS

The mathematical model of moving solids can be created entirely by MSC ADAMS. The main advantage here is obviating the compilation of difficult equations of motion, as geometric models of solids and descriptions of their dynamics are sufficient. The program will automatically calculate the movement of the whole system in pursuance of individual members, which are prescribed by movement or by force application.

ADAMS allows comprehensive analysis of systems by using a simple and flexible modelling of solid objects. Adams offers multiple types of simulations and many tools to help build models as accurately as possible. All objects in the assembly need to be interconnected by means of kinematic links into a single unit, which corresponds to the real characteristics of the manipulator.

In MSC ADAMS, there are three different simulation types from which the user can choose the most appropriate one. Those are: kinematic, static and dynamic.

The kinematic mode is applicable when there are zero degrees of freedom. Any movement in the system is done by forced motions at joints and there are no freely moving parts. In this mode, ADAMS uncouples the motion and force equations and rst solves positions, then velocities, accelerations and forces algebraically. In static or quasi-static mode reaction forces are determined so that they balance out the external forces and loads and the whole system is in equilibrium independently at each time step.

When the system has at least one degree of freedom, a dynamic analysis is required. In dynamic simulation, the differential equations are automatically formulated and numerically solved to determine the system's components' positions, velocities, accelerations and forces.

Care must be taken while importing the CAD model of the robot into ADAMS, as shown in Figure 59. There are 8 links and 7 joints in the system. Initially, these joints were defined in the software after importing the model. The position of joints was made to correspond with the stipulated assignment as per the default 'Home' position, as per Figure 60.
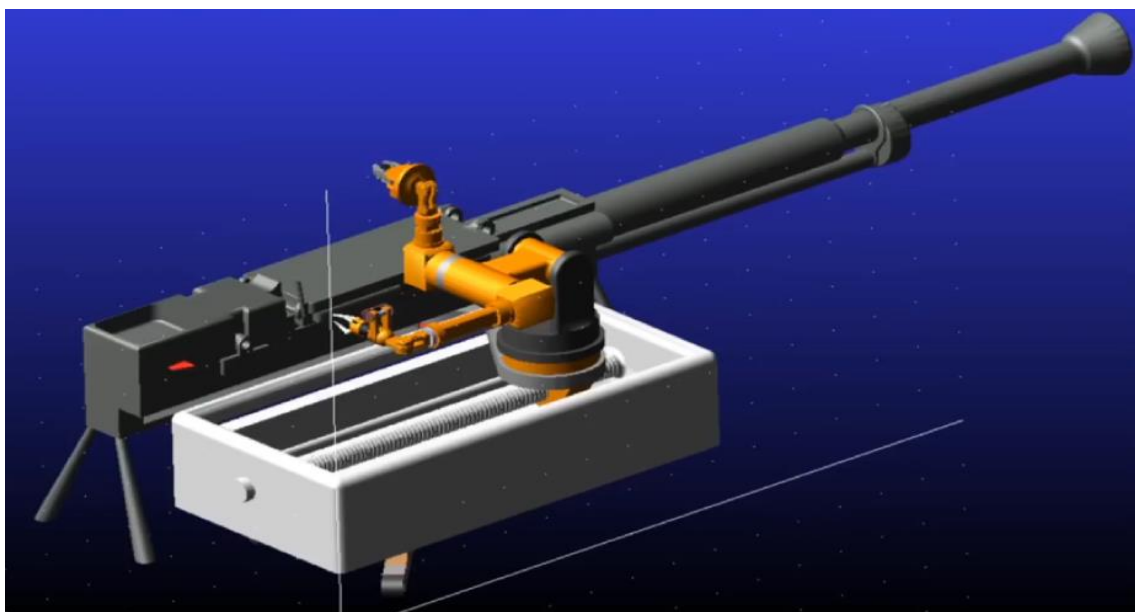

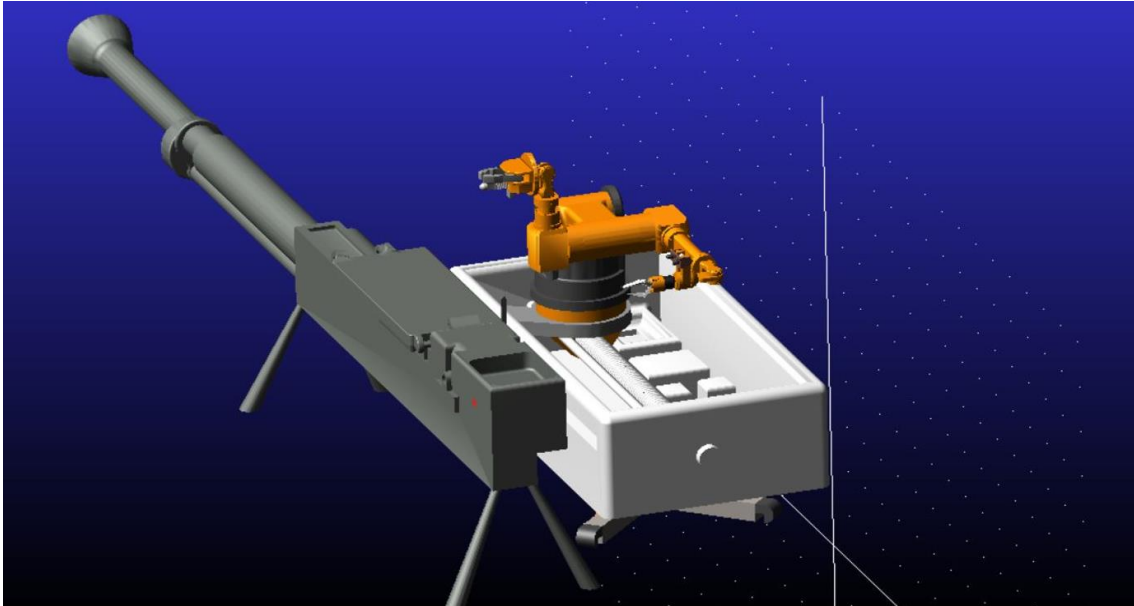Figure 59: The system as imported into ADAMS

Figure 60: "Home" position of the robotic arm links, before the start of an operation

The simulations are all done at predetermined time steps, known as integration time steps. Step sizes need to be selected, though the program computes additional time points between the steps. The simulation stop time, that is, the time limit for how long the simulation is run, is also set.

Upon import of the CAD model, it has to be noted that the mates and references created in SolidWorks would not be transferred. This would mean that the bodies are in the positions in space as created, but do not have any joints or supports attributed to them. In other words, the assembly is not a kinematic chain yet. This is rectified when joints are created between elements. The most fundamental joint is the fixture between a part and the ground link, for which a joint can be made with the link and any part of the empty workspace, which is considered as the ground. Further, the other joints between all the remaining parts need to be defined. The successive parts in the assembly which are created as separate entities but do not form part of the rotary and prismatic joints of essence to the joint motion, can be given fixed joints to ensure that they do not cause unwarranted motion and to maintain simplicity. An alternative way to achieve this is to perform- the Boolean operation of addition on those bodies. Then, the actual 7 joints of interest which grant the degrees of freedom of the assembly are created, along with the joints

needed to perform gripper finger movements. Figure 61 shows a total view of all the steps involved in creating a joint, which will be explained in detail, step by step.
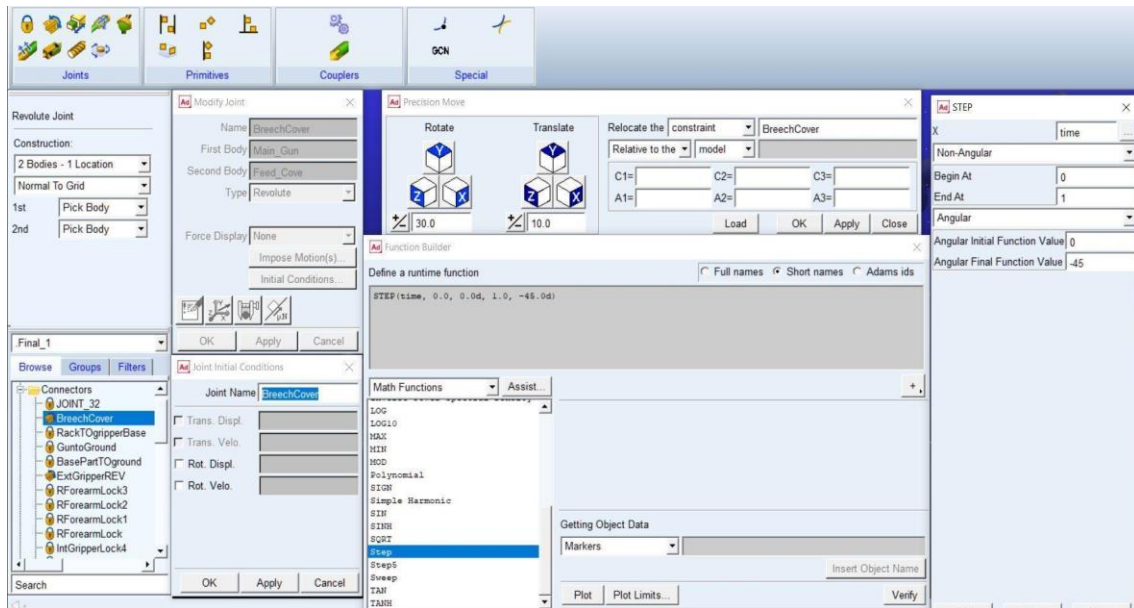

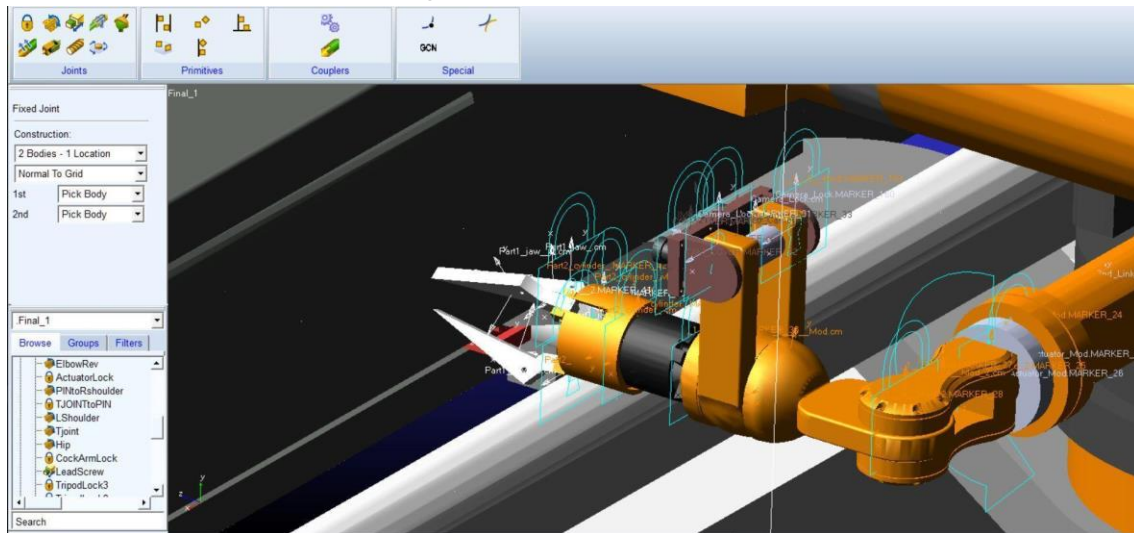Figure 61: Joint creation

Figure 62: Joint assignment

Firstly, the bodies involved in a joint should be selected. Then, a location for the joint needs to be specified. Figure 62 shows the various "Fixed" joints created on the left arm end effector subassembly to convey their absence of relative motion.
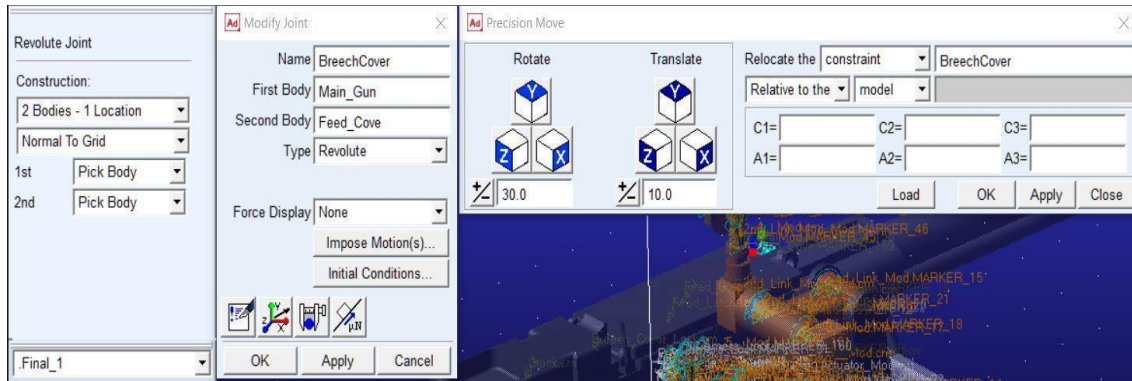
Figure 63: Joint Axis Orientation

Following joint location, the joint orientation has to be assigned using the tools shown in Figure 63. This is critical because the axis of rotation in case of revolute joints and translation in case of prismatic joints would result in significant changes in kinematic and dynamic parameters, if the orientation is not accurate.
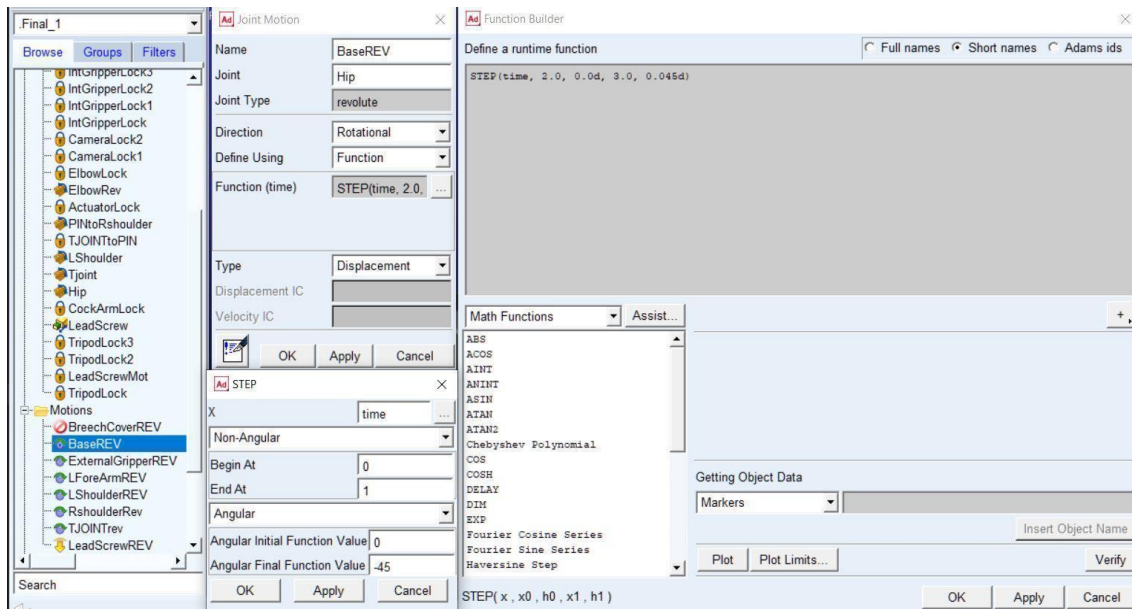
Figure 64: Joint motion definition

The next step is to define the joint motion. The motion is defined as a function of the displacement because in the simulations carried out, the joints would need to perform the initial motion within the basic cycle time of 6 seconds, as determined by inverse kinematics, to reach the critical points. From then onwards, the joints involved would need to perform another motion to bring about the operations to be

done. Thus, the commands were written accordingly, as seen in Figure 64, and are explained further in detail for each operation.

5.4 Analyses and Results

The joint variables for each joint, which are rendered from the inverse kinematics calculations done in MATLAB are given as input for the simulation in ADAMS.

5.4.1 Cocking

The analysis is begun by determining the motion to completely cock the gun using the pin, up until it is locked in position at the maximum compression state of the spring that it is coupled with, as depicted in Figure 65. For this analysis, the carriage was made to translate till the end of its lead screw, and the parameters of this simulation are documented based on the plots obtained at the end of post-processing (Figure 66), in carriage translation distance, carriage translation velocity, force required to drive the carriage against the cocking pin, and the power required for the operation.
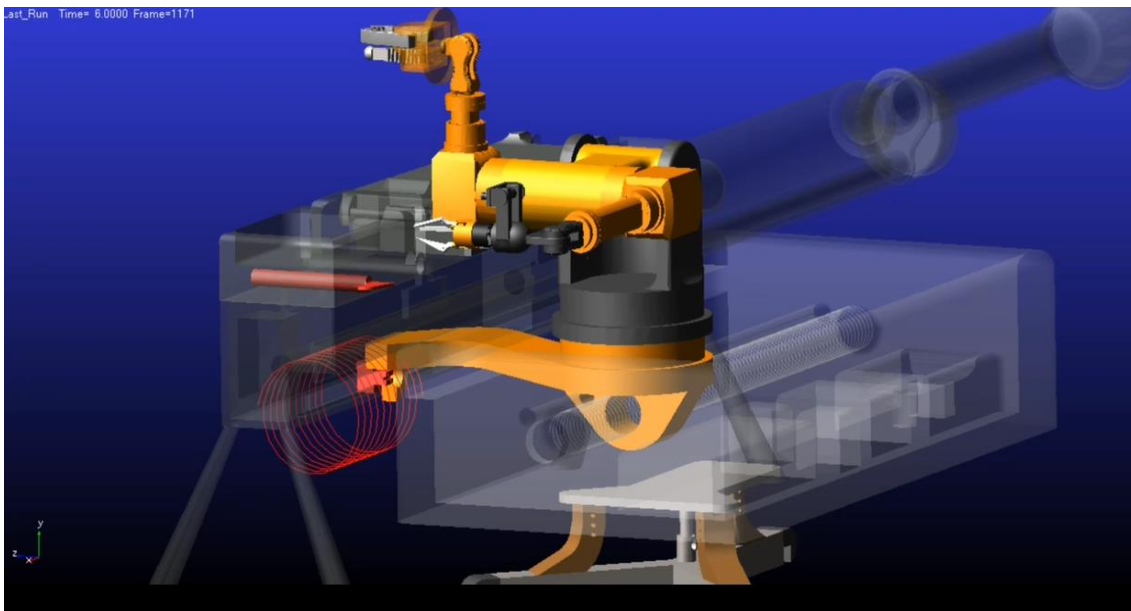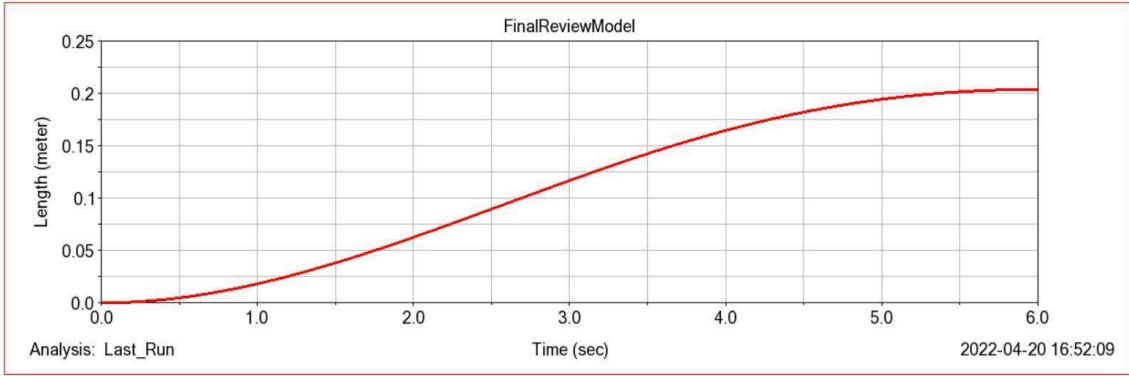

Figure 65: Cocking process

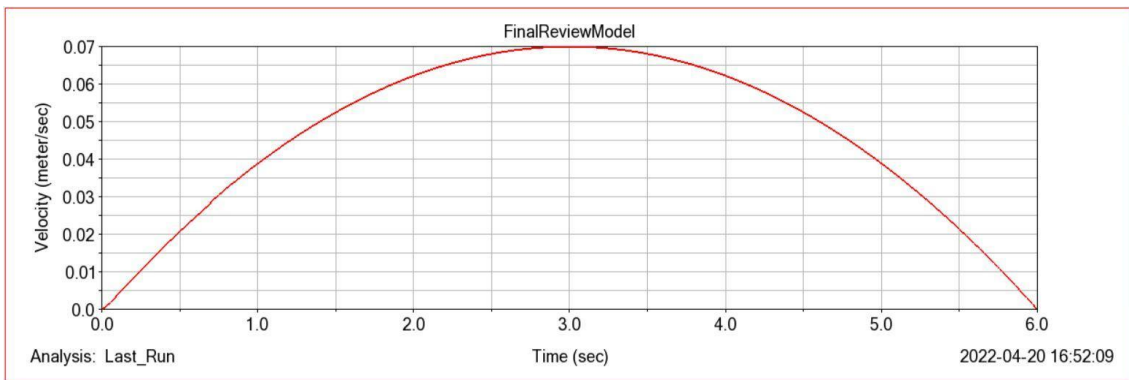Figure 66(a): Carriage Translation Distance
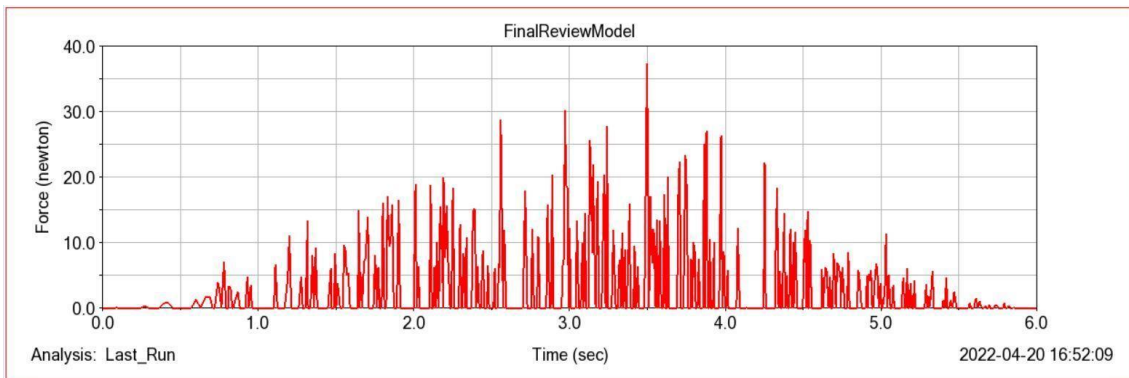

Figure 66(b): Carriage Translation Velocity

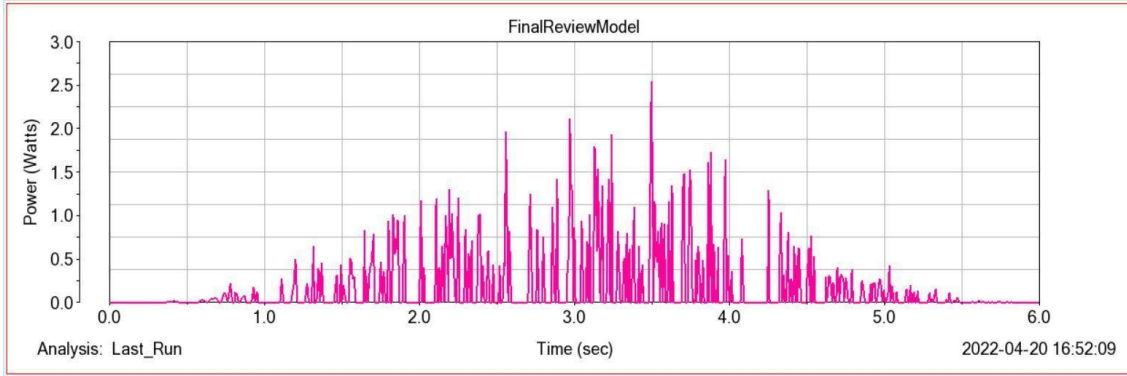Figure 66(c): Force required to cock the gun

Figure 66(d): Power required to cock the gun

Figure 66: Results obtained from MBD simulation of cocking process

5.4.2 Safety Lever Operation

The safety lever, as discussed earlier, can be rotated only when it is pushed inwards while operating. The motion to reach and rotate the safety lever was communicated to the joints, with the operation being performed through contributions from the carriage, T-Joint, Right Upper Arm and the rotary motion of the gripper. The safety lever was operated to safety in 2 seconds. The whole movement from the home position took 10 seconds, which was also the total simulation time used. By the end of 6th second, the end effector positions itself in front of the safety lever, ready to operate it (Figure 67). From the 6th to 8th second, the claws converge and grip the safety lever, at the same time pushing the safety lever 3 mm inwards by means of the rack (Figure 68). From the 8th to the 10th second the end effector rotates about 180 deg as intended. The results of this analysis are documented in Figure 69.
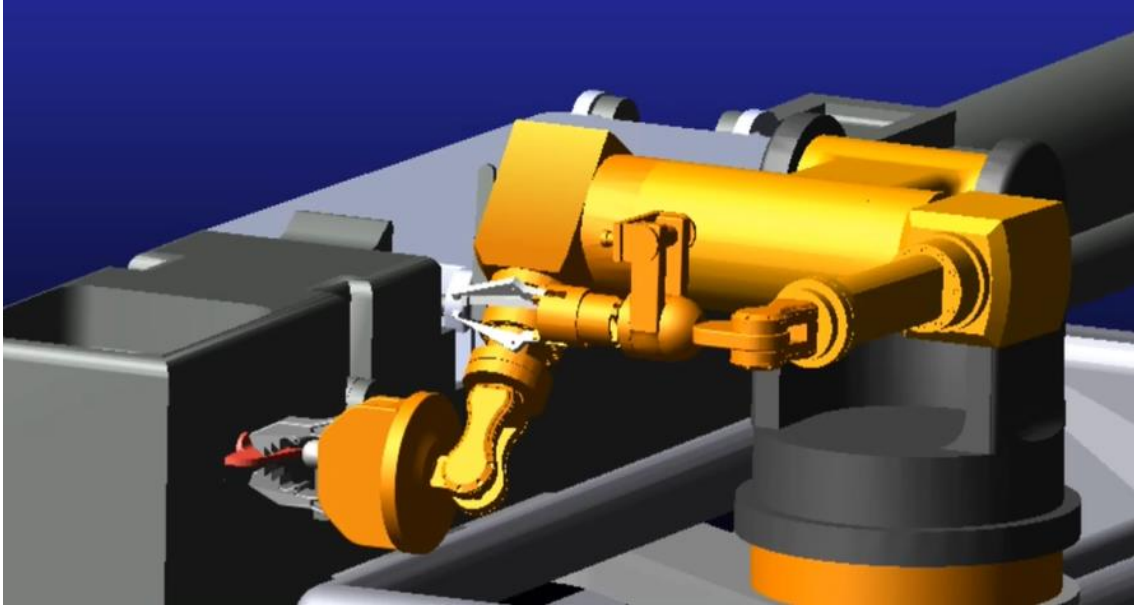
Figure 67: Safety lever operating position
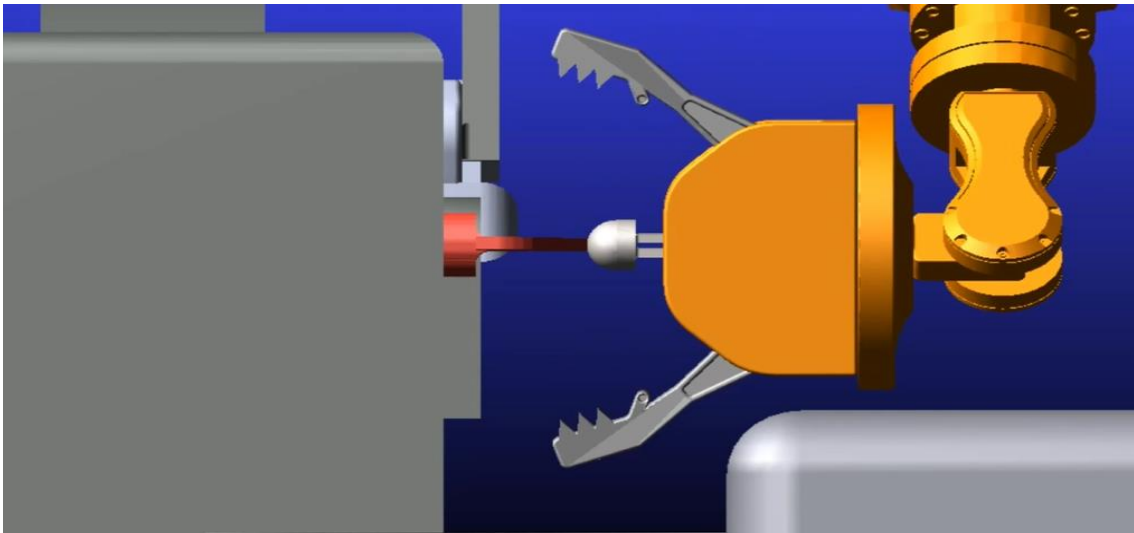
Figure 68(a): Claws on the verge of grasping safety lever
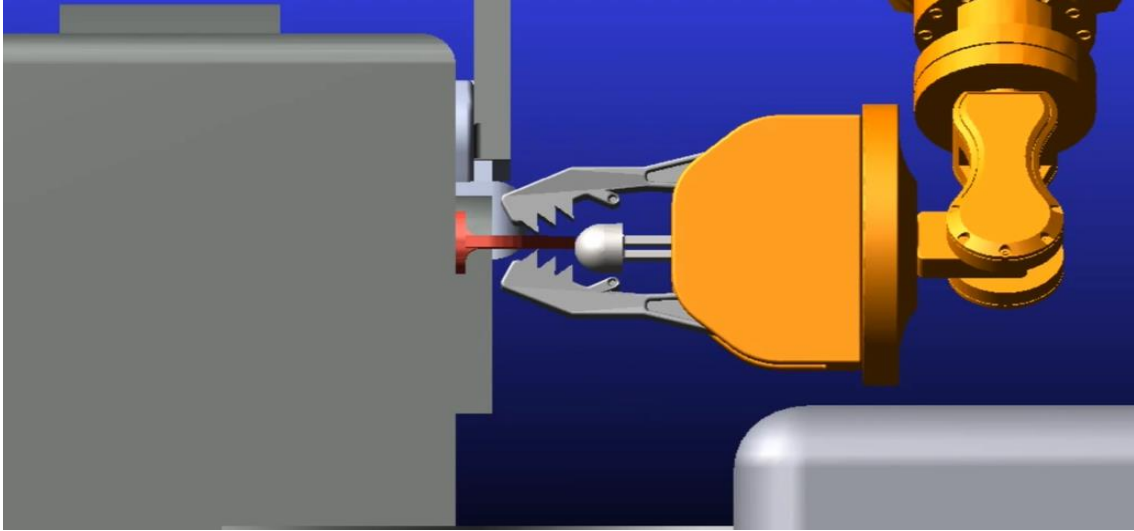
Figure 68(b): Convergence of claws to push and hold the safety lever
Figure 68: Gripper Claws movement to operate safety lever

Figure 69(a): End Effector distance of travel from Home position
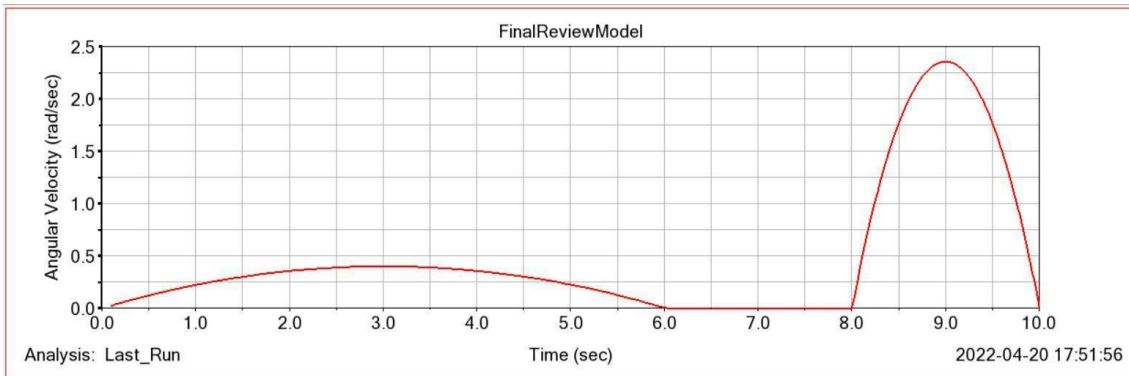
Figure 69(b): End Effector Angular Velocity over time
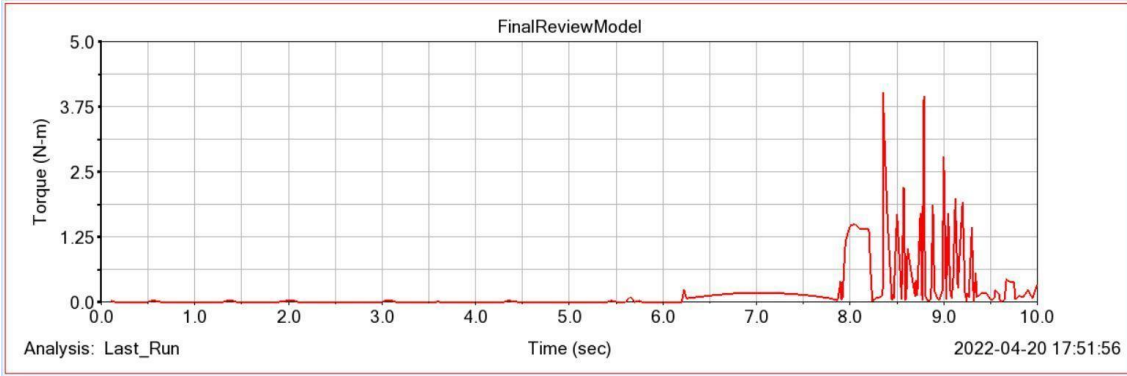
Figure 69(c): Torque experienced by the safety lever
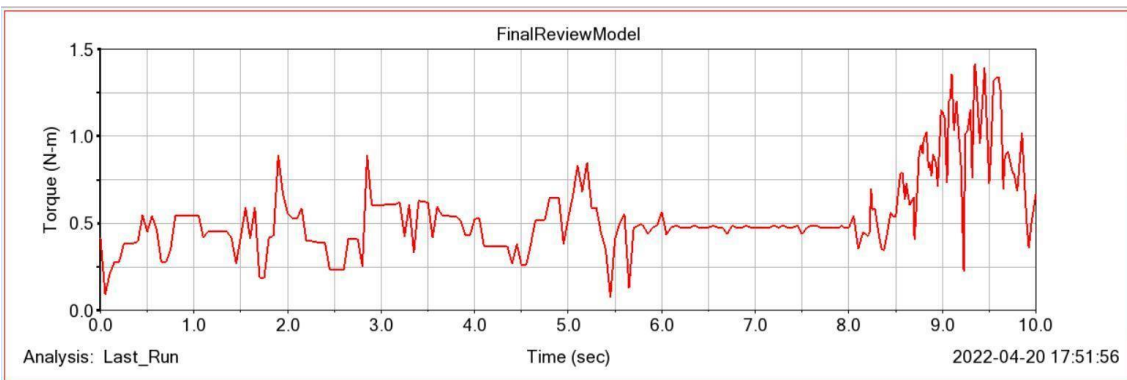
Figure 69(d): Torque due to the rotation of the gripper base
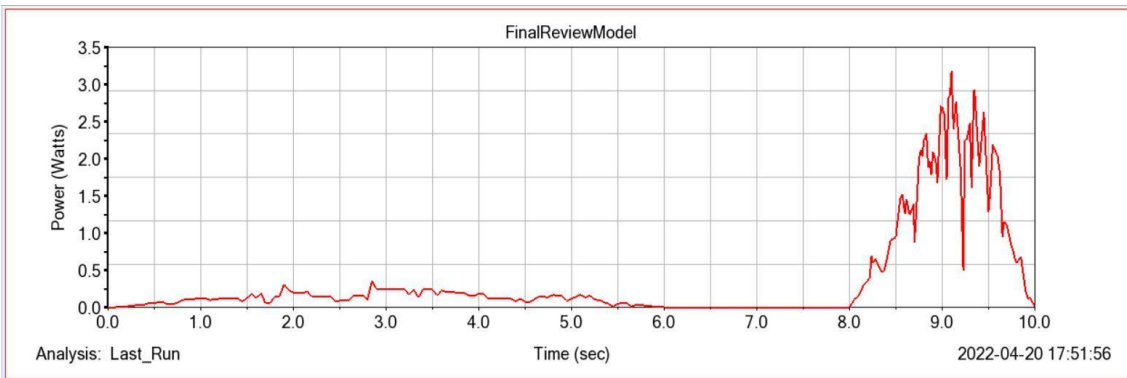
Figure 69(e): Power requirement for rotation of the gripper base

Figure 69: Results obtained from MBD simulation of safety lever operation

5.4.3   Breech Cover Opening

The motion to hold and open the breech cover involved the carriage translation, and rotation of T-Joint, Right Upper Arm, and Right Gripper. Once the initial movements to reach the critical point were done, the breech cover needed to be rotated from zero to 45 degrees in 2 seconds. The claw fingers clasp onto the breech cover lever and bring about this activity.

In the first 3 seconds, the links all reach the respective positions so as to bring the end effector at the critical point for this operation. From the 3rd to 4th seconds, the claws converge and grip the breech cover's lever, and the carriage moves linearly backwards of about 10mm and from the 4th to 5th seconds, the T-joint rotates 45 degrees and hence successfully opens the breech cover by the 5th second (Figure 70). The results of this analysis are documented in Figure 71.
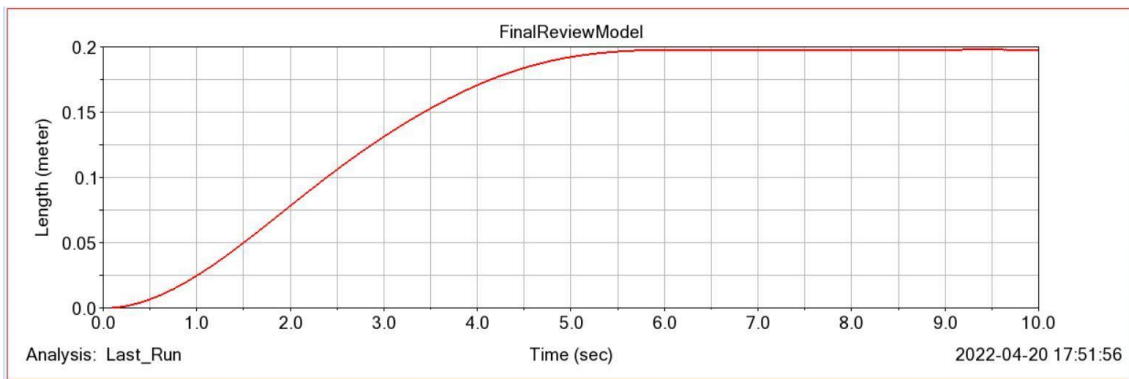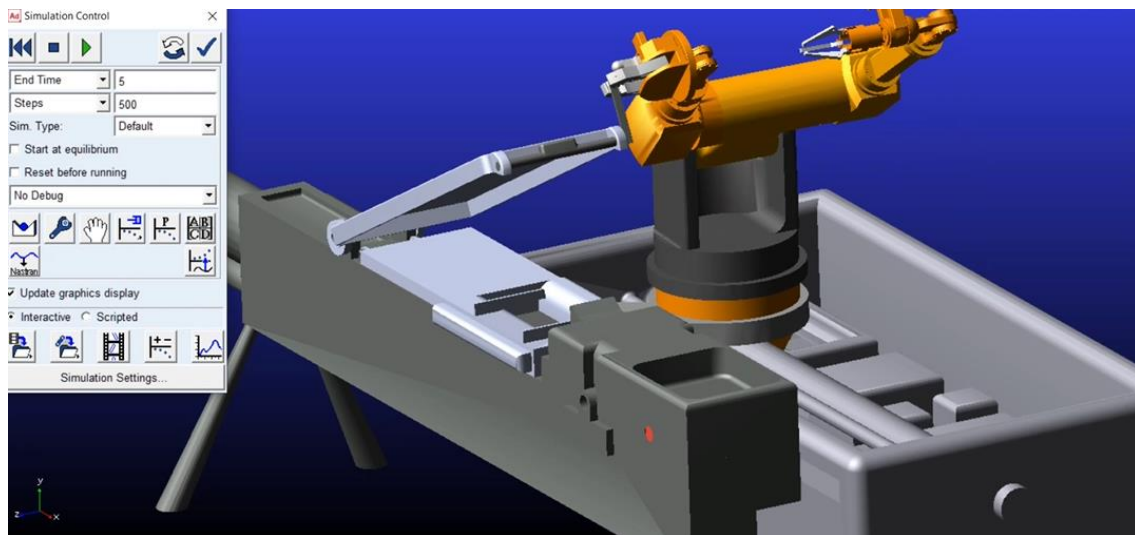

Figure 70: External Arm opening the breech cover

Figure 71(a): End Effector distance of travel from Home position
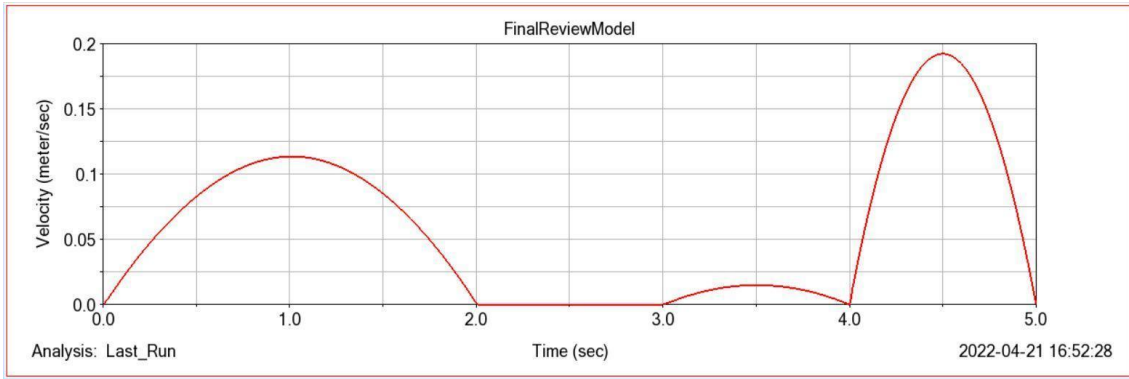
Figure 71(b): End Effector Angular Velocity over time
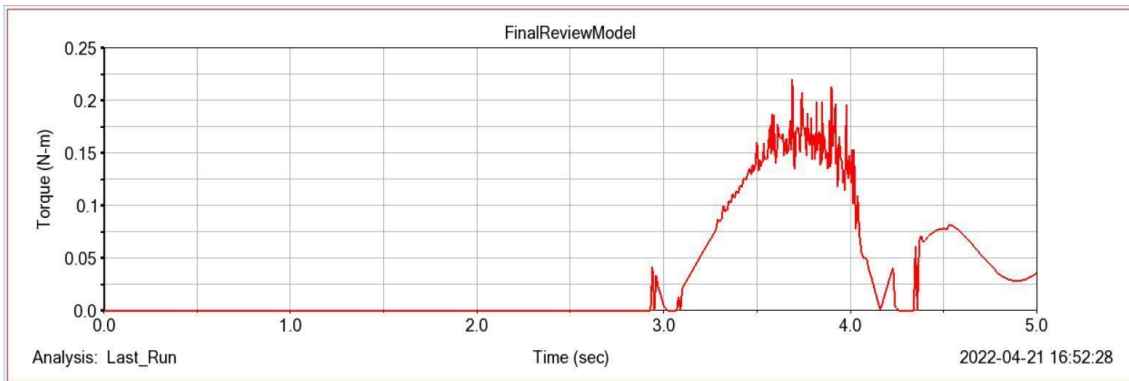
Figure 71(c): Torque for driving the T-Joint
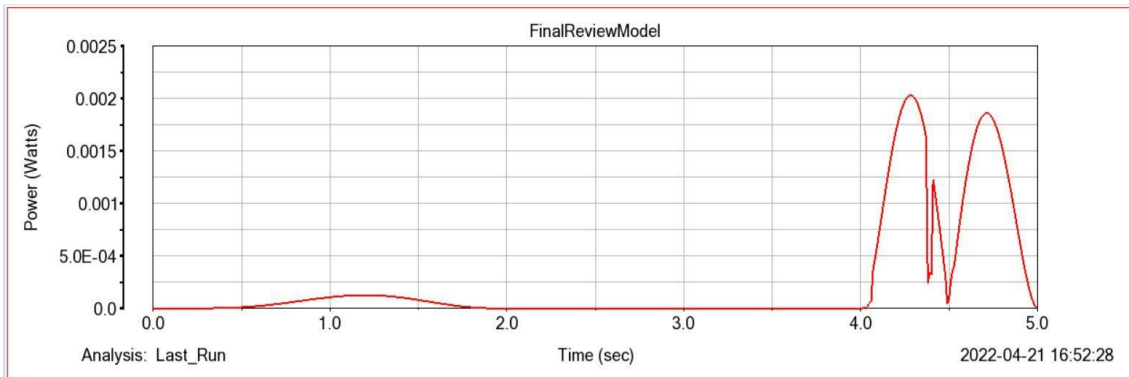
Figure 71(d): Power for driving the T-Joint
Figure 71: Results obtained from MBD simulation of breech cover opening

5.4.4   Cartridge Tray Opening

The motion to hold and open the cartridge tray involved the carriage translation, and rotation of T-Joint, Right Upper Arm, and Right Gripper. Once the initial movements to reach the critical point were done, the cartridge tray needed to be rotated from zero to 45 degrees in 2 seconds. The claw fingers clasp onto the tray and bring about this activity.

In the first 6 seconds, the links all reach the respective positions, so as to bring the end effector at the critical point for this operation. From the 6th to 8th seconds, the claws converge and grip the edge of the tray, and the carriage moves linearly backwards of about 8mm and from the 8th to 10th seconds, the T-joint rotates 45 degrees, while the gripper rotates 30 degrees in the opposite direction simultaneously, and hence successfully opens the cartridge tray by the 10th second (Figure 72). The results of this analysis are documented in Figure 73.

Figure 72: External arm opening cartridge tray

Figure 73(a): End Effector distance of travel from Home position
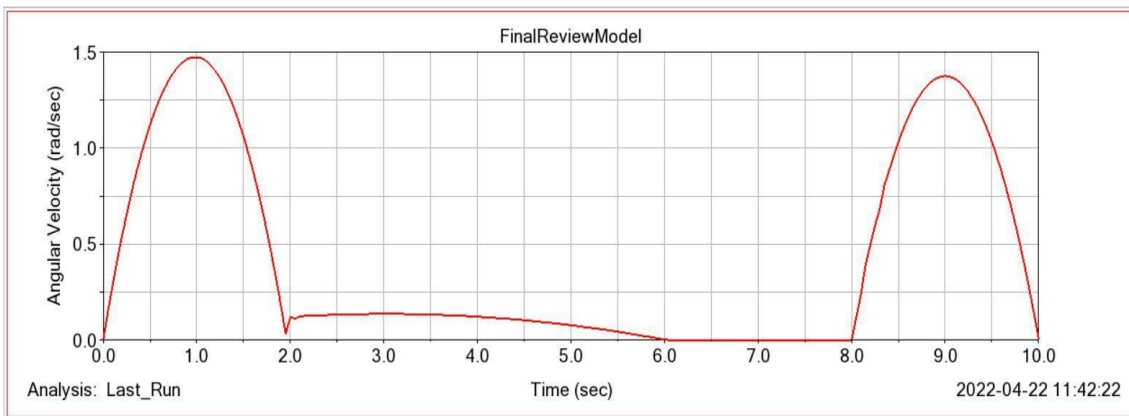
Figure 73(b): End Effector Angular Velocity over time

Figure 73(c): Torque for driving the Right Upper Arm
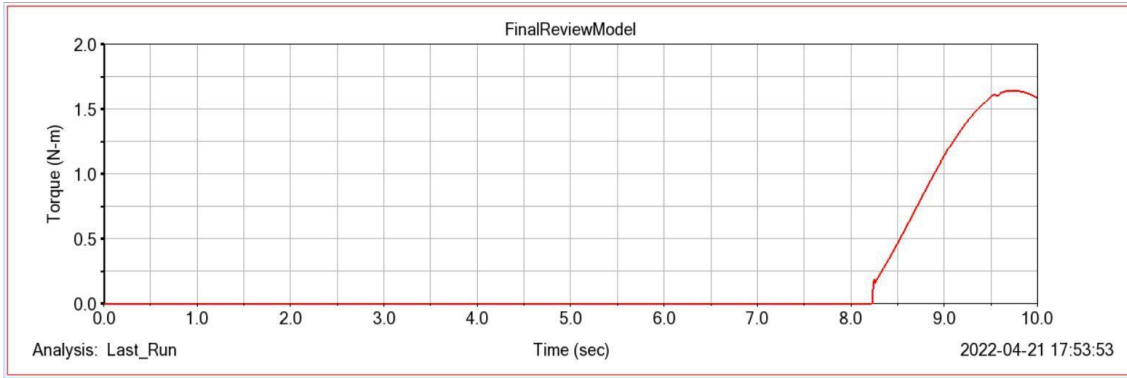
Figure 73(d): Torque experienced by the Cartridge Tray
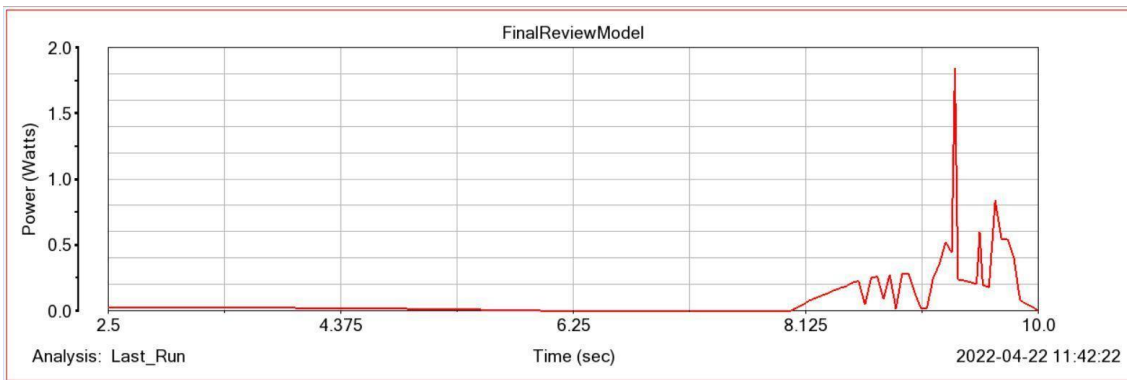
Figure 73(e): Power required to drive the Right Upper Arm

Figure 73: Results obtained from MBD simulation of cartridge tray opening

5.4.5 Belt Retrieval

The motion to hold and lift the belt by the Internal Operation Arm involved the carriage translation, rotation of T-Joint, Left Upper Arm, and Left Lower Arm. Once the initial movements to reach the critical point were done, the ammunition belt needed to be pulled upwards to a distance of 40 mm from its original position in 2 seconds. The gripper jaws clasp onto the belt and bring about this activity.

In the first 6 seconds, the links all reach the respective positions, so as to bring the end effector at the critical point for this operation (Figure 74). From the 6th to 8th seconds, the claws converge and grip a link of the belt, and then the T-Joint rotates in the clockwise direction from the 8th to 12th seconds, so as to successfully retrieve the ammunition belt, and recover the payload by the 12th second (Figure 75). The results of this analysis are documented in Figure 76.
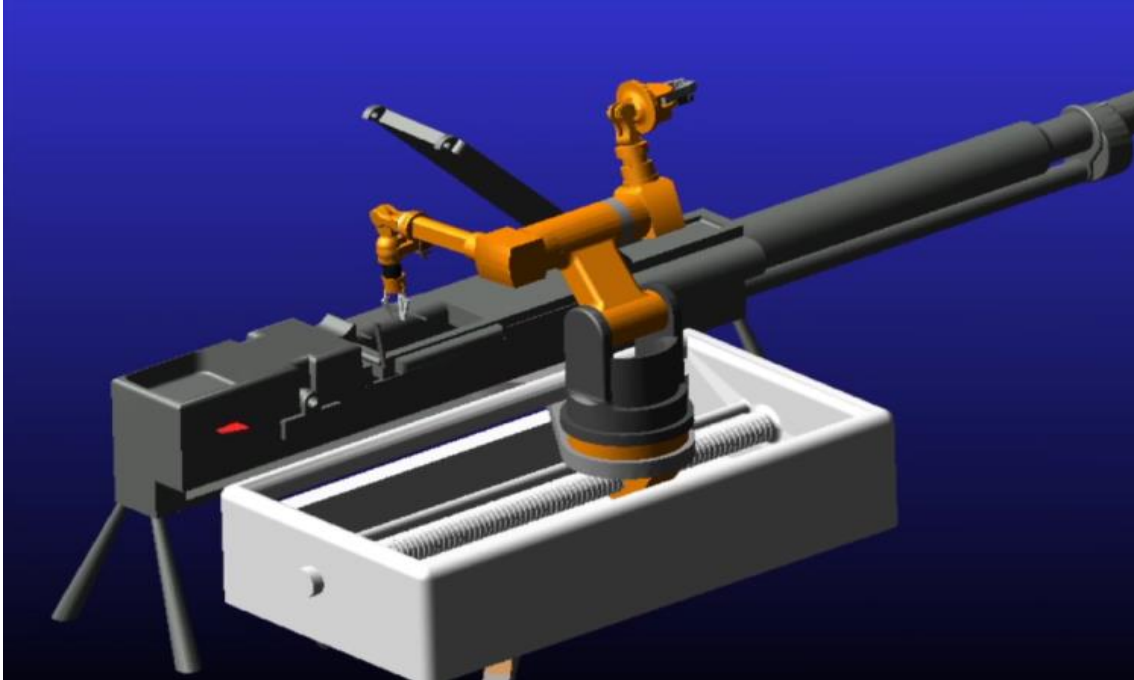
Figure 74: Internal Operation Arm positioned to lift the belt

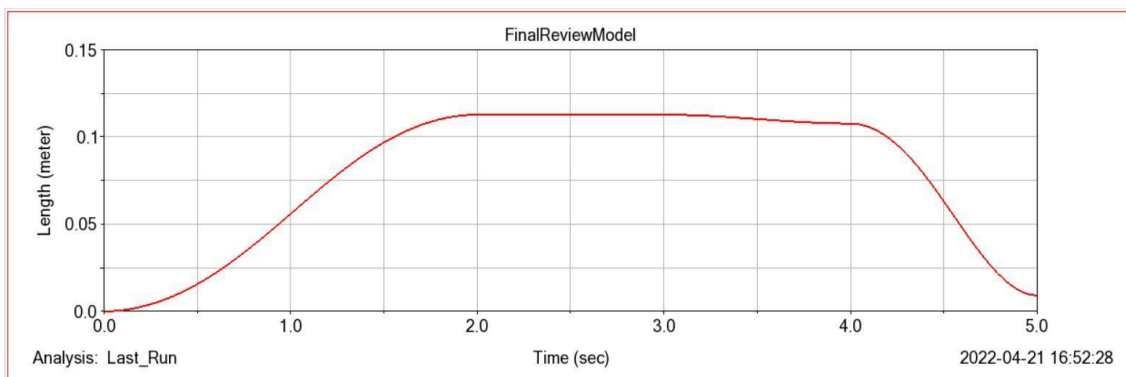Figure 75: Internal Operation Arm lifting the belt
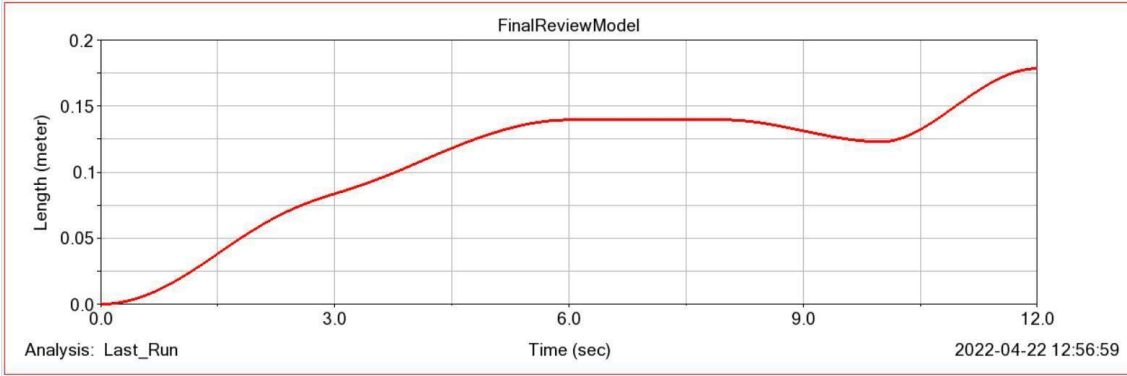
Figure 76(a): End Effector distance of travel from Home position
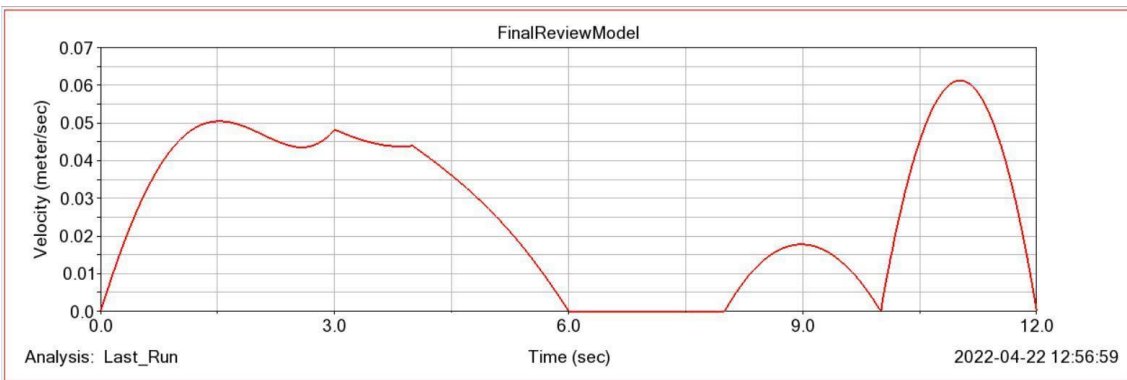

Figure 76(b): End Effector Angular Velocity over time
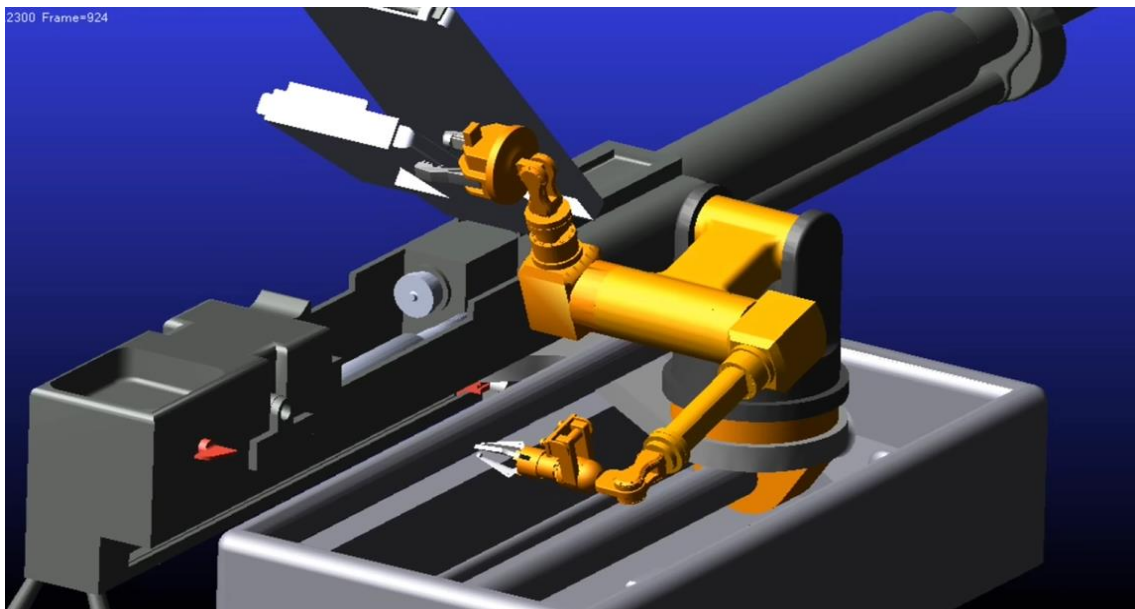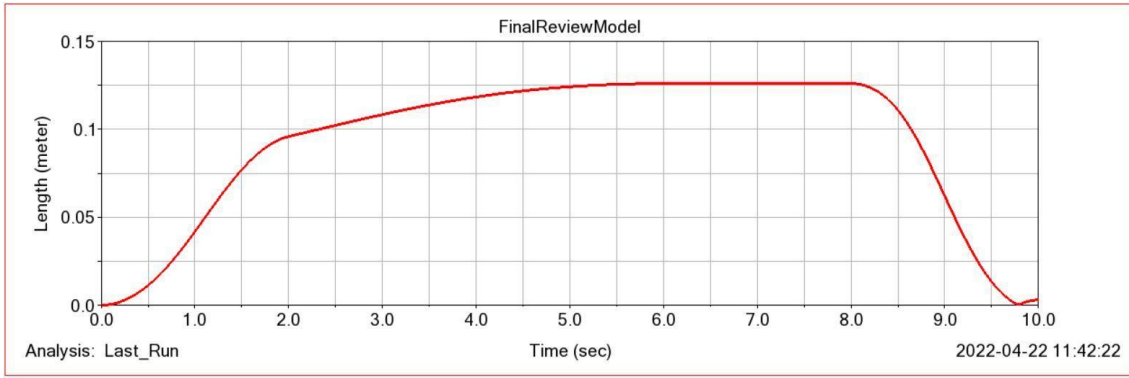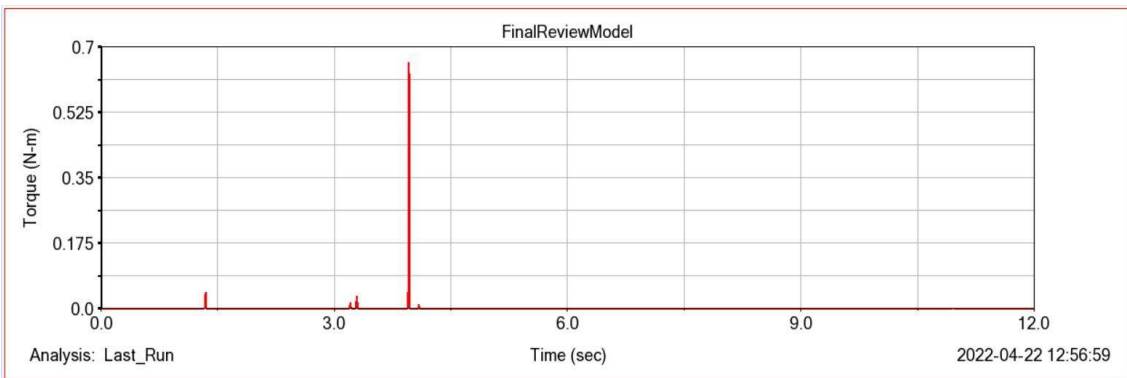

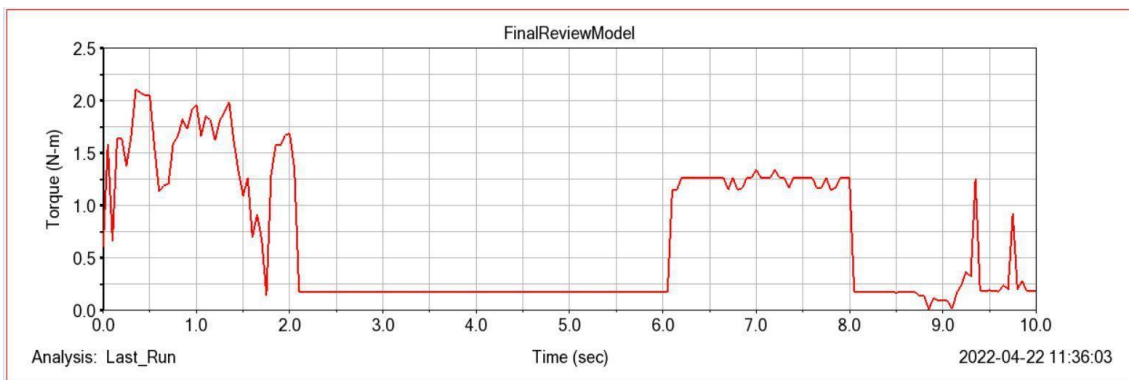Figure 76(c): Torque for driving the Left Lower Arm
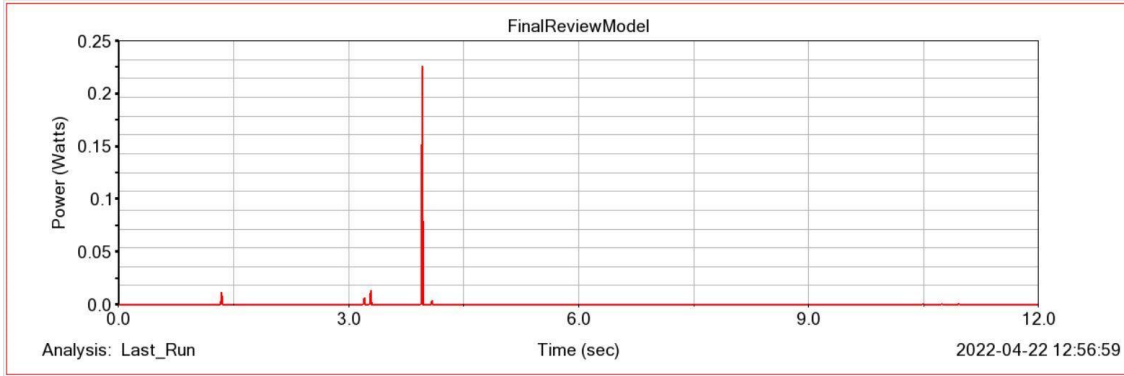
Figure 76(d): Power for driving the Left Lower Arm
Figure 76: Results obtained from MBD simulation of belt retrieval

5.4.6 Ammunition Retrieval

The motion to retrieve the ammunition from any position inside the firing chamber involves the carriage translation, rotation of T-Joint, Left Upper Arm, and Left Lower Arm. Once the initial movements to reach the critical point were done, the ammunition needed to be pulled outwards to a distance of 120 mm from its original position in 2 seconds, to completely remove it from the chamber. The gripper jaws clasp onto the dorsal end of the bullet to bring about this activity. This movement can be considered as simulation of bullet retrievals from both partial (Figure 77) and whole jamming (Figure 78) in the firing chamber. This is because, the difference between the former and the latter is only a change in the tool used, in mechanical gripper and vacuum-based gripper, respectively. Otherwise, the movement involved is the same.

In the first 6 seconds, the links all reach the respective positions, so as to bring the end effector at the critical point for this operation. From the 6th to 8th seconds, the claws converge and grip the backside of the bullet, and then the carriage translates for about 120 mm away from the chamber, from the 8th to 15th seconds, so as to successfully retrieve the ammunition, and recover the payload by the 15th second (Figure 79). The results of this analysis are documented in Figure 80.
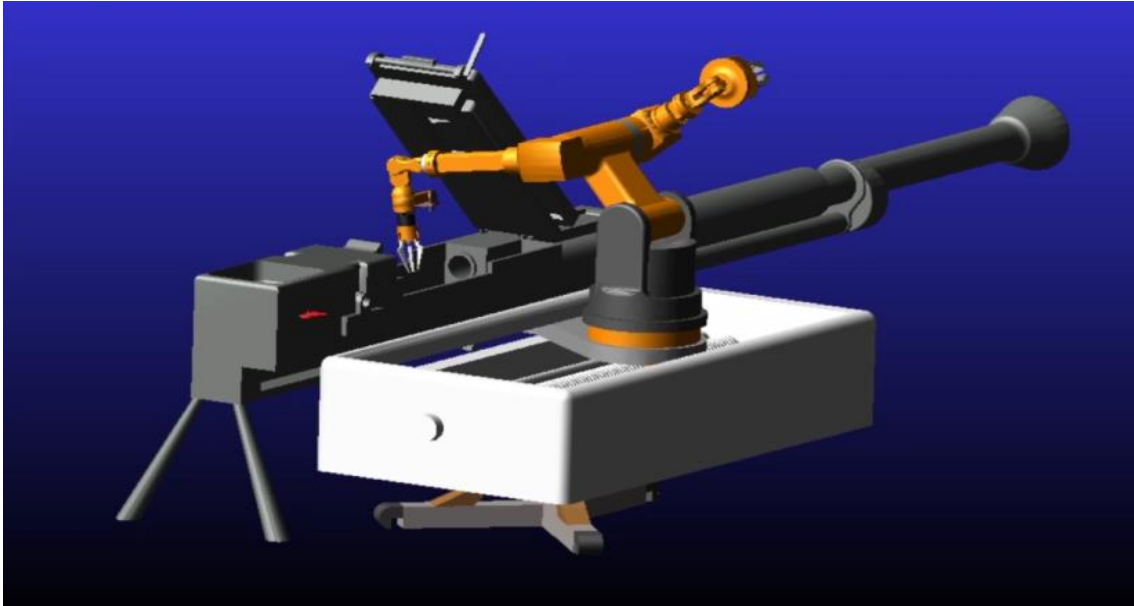
Figure 77: Internal operation arm positioned at one of the ammunition retrieval positions. This is the case where the ammunition is jammed in a part of the breech, on its way to the firing chamber
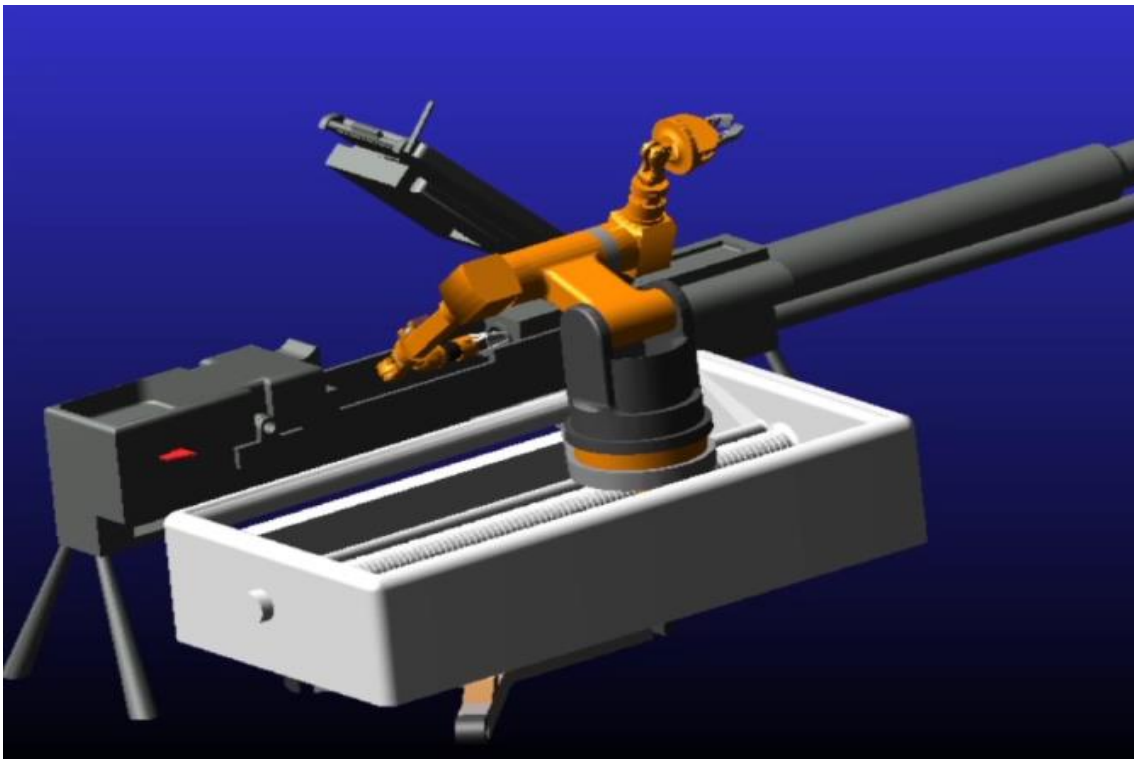
Figure 78: Internal Operation Arm positioned at the final ammunition retrieval position, which is when the ammunition gets jammed inside the firing chamber
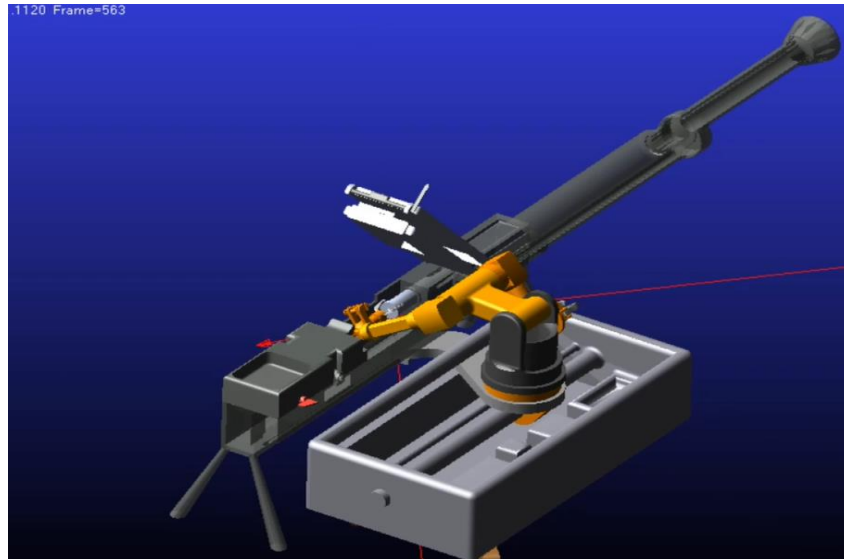
Figure 79(a): Left Gripper retrieving ammunition
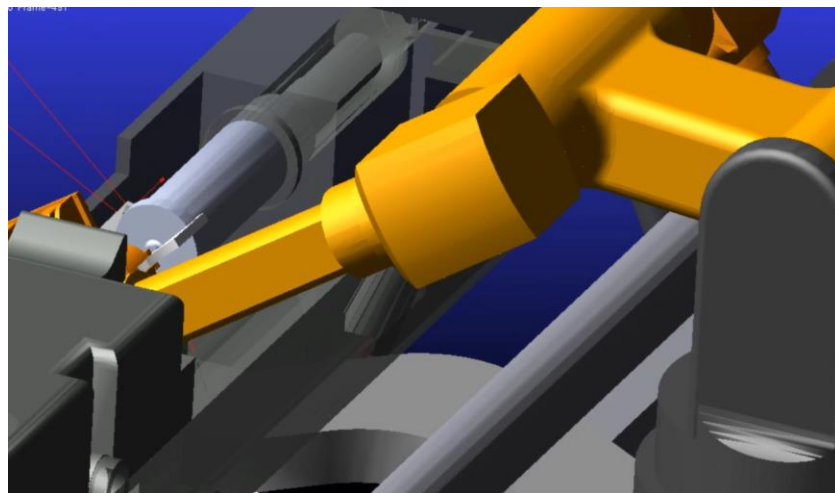
Figure 79(b): Close-up View of Left Gripper retrieving ammunition
Figure 79: Retrieval of Ammunition by the Left Gripper
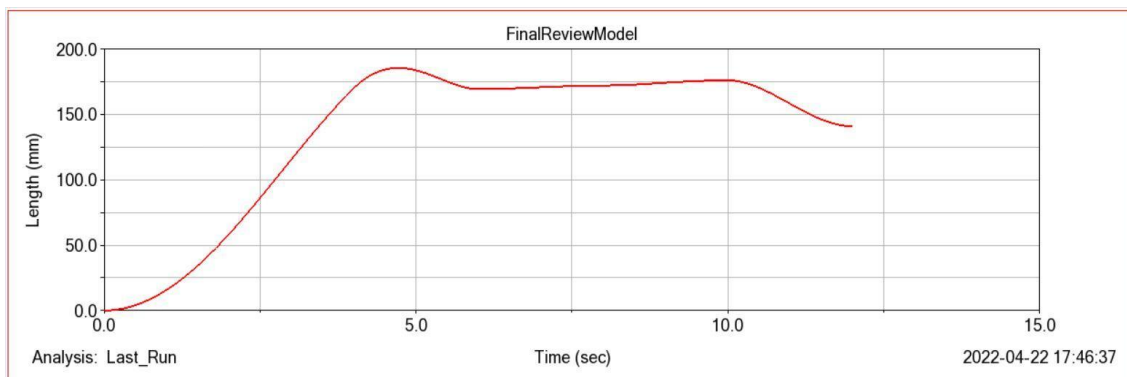
Figure 80(a): End Effector distance of travel from Home position
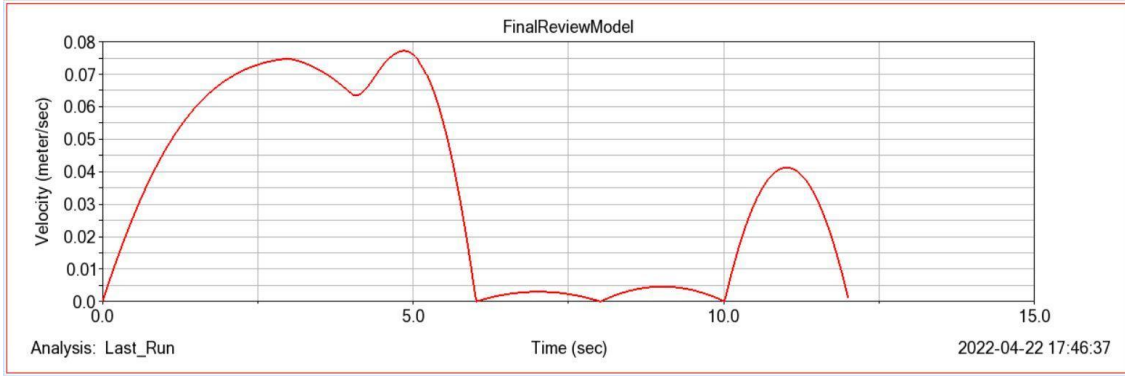
Figure 80(b): End Effector Angular Velocity over time
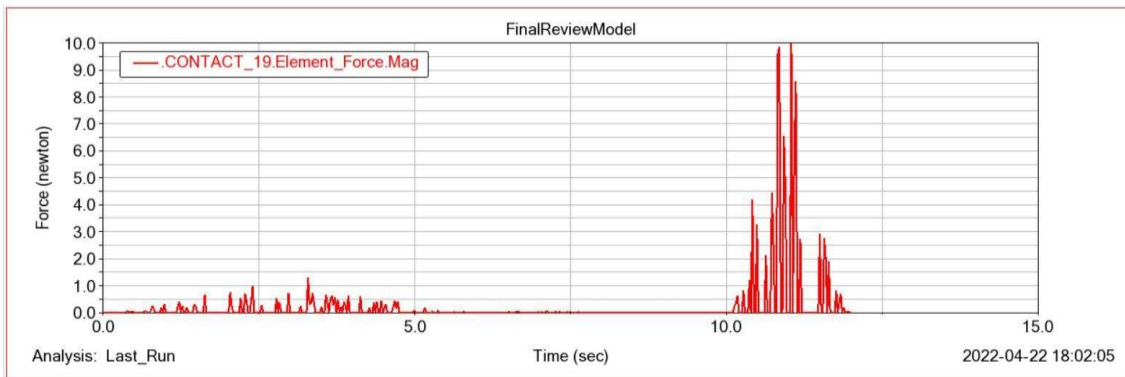

Figure 80(c): Force for driving the carriage
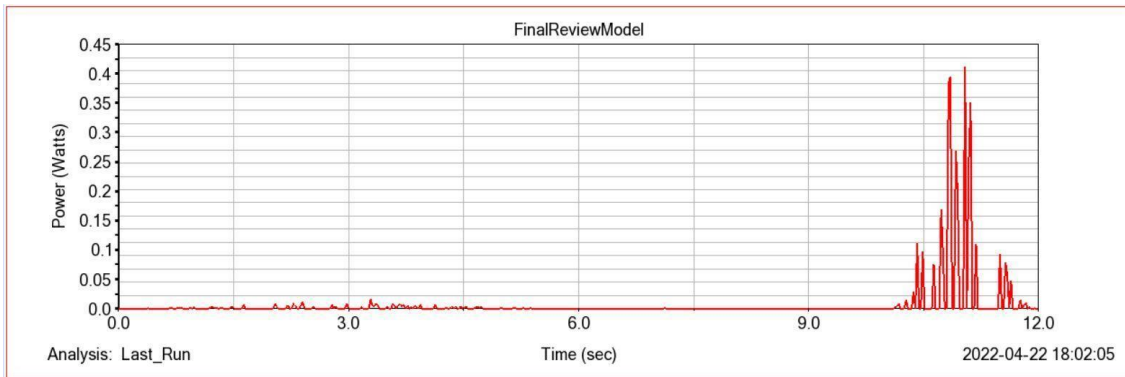
Figure 80(d): Power for driving the carriage
Figure 80: Results obtained from MBD simulation of ammunition retrieval

5.5 Motor Selections

The results of MBD in each individual operation has yielded the following results, which contain the highest force, torque and power requirements necessitated by each individual link, considering all the operations that it is supposed to perform. Based

on such requirements, brushed DC motors have been selected from the catalogue provided by the motor manufacturers, Maxon. The dimensions of such motors that fit the applications have been attached alongside in Table 10, in order to account for their assembly within the system.

Table 10: Motor Dimensions for each Joint

Driven Link	Max Torque (Nm)/ Force (N)	Max Power (W)	Motor Diameter (mm)
Carriage	37.5 N	2.5	22
Base	0.8 Nm	1.2	16
T-Joint	2.15 Nm	1.8	16
Right Upper Arm	1.85 Nm	0.6	10
Right Lower Arm	1.37 Nm	3.2	19
Left Upper Arm	0.7 Nm	1.5	19
Left Lower Arm	0.5 Nm	2.5	16

CHAPTER 6

STRUCTURAL ANALYSIS

Structural analysis has been used to determine the strength and rigidity of the proposed system. Finite Element Analysis (FEA) tools have been employed to get the intended results. FEA employs the fields of applied mechanics, materials science and applied mathematics to compute a structure's deformations and stresses. These results are usually depicted via a color scale that shows, for example, the stress distribution over the object.

The process of building up an FEA model consists of many steps, which are listed below.

- Defining the geometric domain of the problem
- Defining the element types to be used
- Define the material properties of element
- Defining the geometric properties of elements
- Defining element connectivity i.e. meshing the model
- Defining the physical constraints (boundary condition)
- Defining the loadings.

All the processes involved in building up the problem can be collectively termed as "pre-processing". All the elements are three-dimensional homogeneous in nature, and the material is defined to be Al 7075 t6 alloy, as mentioned earlier. For our model, we made use of two Finite Element solvers, in ADAMS Flex, and the ANSYS software, to carry out FEA. The entity that needs to be solved is then meshed into simpler elements for easier calculation. Meshing or discretization is important because, even though all the physical phenomena around us are continuous, solving a problem using a computer with that approach is very difficult, if not impossible. So, the basis of the numerical methods is to discretize the problem in order to make it understandable to the computer. The result of this discretization is a mesh composed of nodes located in space, connected with entities called elements. Basically, the calculations are done at the nodes and the results are interpolated to the elements to acquire a general solution. So, the results accuracy depends on the number of nodes used to discretize the system, and thus increasing the number of

calculation points increases the accuracy. Following generation of the results, post-processing takes place, during which, the visualization of the results could be defined, by means of a variety of output requests as per the requirement.

6.1 ADAMS Flex Simulation

Traditionally, the multi-body dynamics simulation done using ADAMS yields the loads experienced by the various components of the assembly, which would then be fed as loading conditions to a simulation model in ANSYS. The Flex module however, directly transfers the calculated forces and torques to carry out FEA in ADAMS itself. It has all the required geometric data and material assignments, which are incorporated into the Flex FEA model by selecting the parts required to be analyzed, and converting them from rigid bodies, into flexible bodies, so that the stresses impinged on them could be considered. Flex meshes the bodies according to specifications such as element type, shape, size, orientation, and order of computation.

6.1.1 Cocking

We performed a Flex FEA during the cocking operation, on the carriage which translates so that its cocking arm would push the pin of the gun into the "cocked" position. At the end of MBD, this operation was identified as the one consuming the most exertion from the system. Since the carriage is the only link required for this operation, it was the most critical member of the mechanism, as it is also required to bear the weight of the entire system. Thus, this part of the cycle was considered for Flex FEA (Figure 81).

The stress contours observed on the carriage during cocking, is depicted below (Figure 82), along with data of the maximum Von Mises stress values found, their location, along with the time during the loading cycle when the particular stress was felt in that location (Table 11).
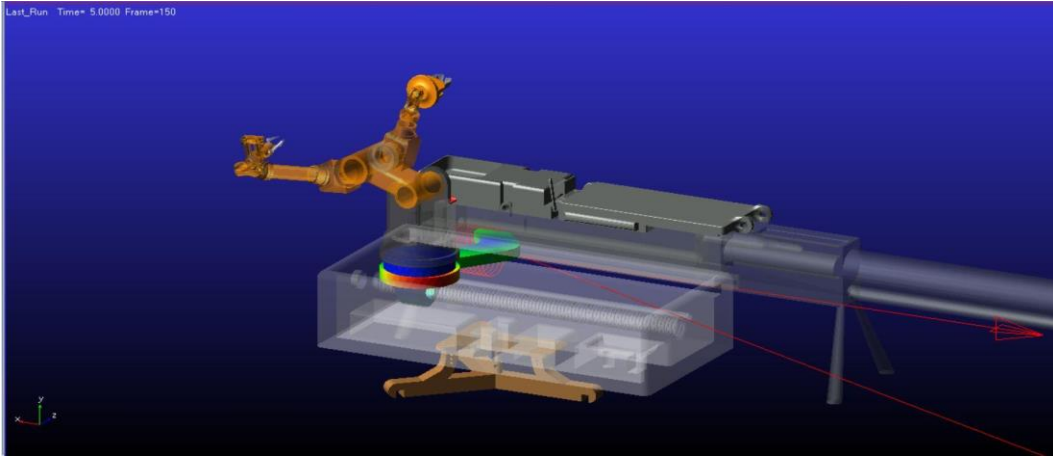
Figure 81: ADAMS Flex view of analysis results
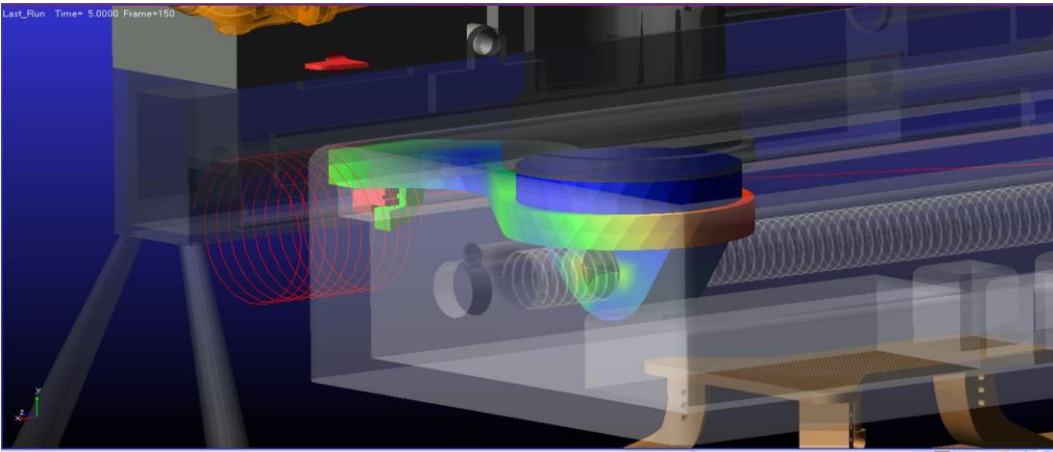
Figure 82: Stress contours observed on the carriage at the end of cocking

Table 11: Von Mises Stress Results for Cocking

VON MISES Hot Spots for CockArm_CV_Mod_flex Date= 2022-04-23 17:48:39						
Model= .Flex		Analysis= Last_Run		Time = 0 to 5 sec		
Top 25 Hot Spots			Abs	Radius= 5 mm		
Hot Spot	Stress	Node	Time	Location wrt LPRF (mm)		
#	(newton/mm**2)	id	(sec)	X	Y	Z
1	83.672	103	4.2	-389.53	135.703	170.541
2	69.0952	104	4.2	-380.017	135.703	173.871
3	64.2721	140	4.08294	-389.573	135.703	176.01
4	63.4014	164	4.16323	-398.163	135.703	171.314
5	59.9749	133	4.08294	-395.218	135.703	184.244
6	48.4231	416	4.2	-393.936	143.232	168.106
7	44.7682	388	4.16323	-389.53	150.703	170.541
8	41.2313	232	4.08294	-389.573	143.182	176.01
9	35.0704	102	4.2	-398.063	135.703	165.177
10	34.6566	208	4.2	-399.814	142.095	170.139
11	34.3301	389	4.16323	-380.017	150.703	173.871
12	32.193	105	4.2	-370.002	135.703	175
13	32.1526	66	4.16323	-389.549	135.703	187.952
14	31.2339	65	4.08294	-378.574	135.703	182.545
15	30.8249	418	4.2	-375.03	143.232	174.718
16	23.4344	257	4.2	-398.163	150.703	171.314
17	21.9777	163	4.2	-405.686	135.703	165.021
18	21.2432	264	4.08294	-395.221	143.59	184.247
19	20.9464	64	4.08294	-382.053	135.703	192.109
20	20.3155	387	4.59737	-398.063	150.703	165.177
21	19.6451	417	4.2	-384.856	143.226	172.477
22	17.3924	267	4.2	-395.218	150.703	184.244
23	17.2575	134	4.16323	-400.871	135.703	192.489
24	15.6592	415	4.2	-401.816	143.226	161.824
25	15.3452	390	4.16323	-370.002	150.703	175

6.2 ANSYS Simulation

Following this, the model was exported to ANSYS, where a Static Structural analysis could be done on the whole system. To set up the FE model in ANSYS, the geometry was assigned with a material created based on the properties of Al 7075 t6 alloy. Then, the meshing process was done, where a hexahedral mesh was created on all the bodies. There were separate analyses done for the safety lever manipulation, breech cover opening, and the belt retrieval processes, based on the

loads pertaining to each process, as calculated by ADAMS. They were done at the moment when the mechanism just engages with the payloads, facing the full brunt of their load, and the whole system is considered as a structure for that specific instance, in order to understand how the stress values owing to payload vary, from the end effector to the base. The particulars of those analyses are documented in the following sections.

6.2.1 Safety Lever Manipulation

This analysis considers the right arm end effector as the point of force application. In this operation, the load was experienced by the claws holding the safety lever and rotating it to 180 degrees. A 1.37 Nm torque is needed to rotate the lever, and this load was applied on the system. As for constraints, the part of the carriage where it is coupled with the lead screw was assigned. The view of the assembly after meshing is shown below. The simulation results shown denote that the working stress and corresponding deformation are well within the yield limits of the considered material, thus substantiating the validity of the design parameters. The maximum value of 85.81 MPa Von Mises stress (Figure 83), and 0.00103 mm deformation (Figure 84) are both found occurring at the claw of the end effector.
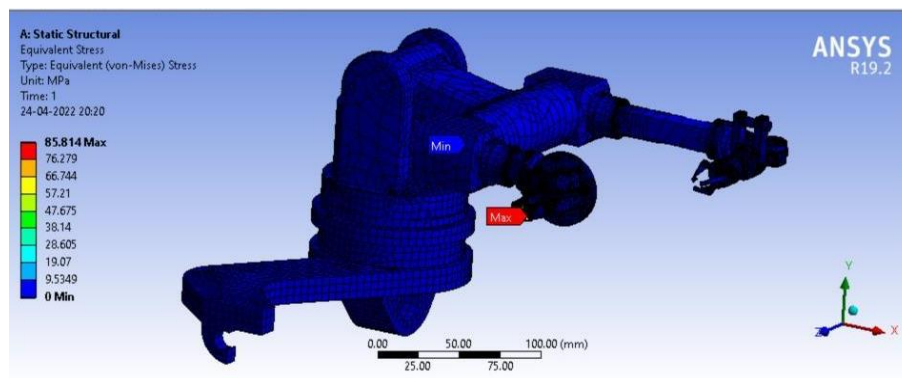
Figure 83: Stress Plot for Safety Lever Manipulation
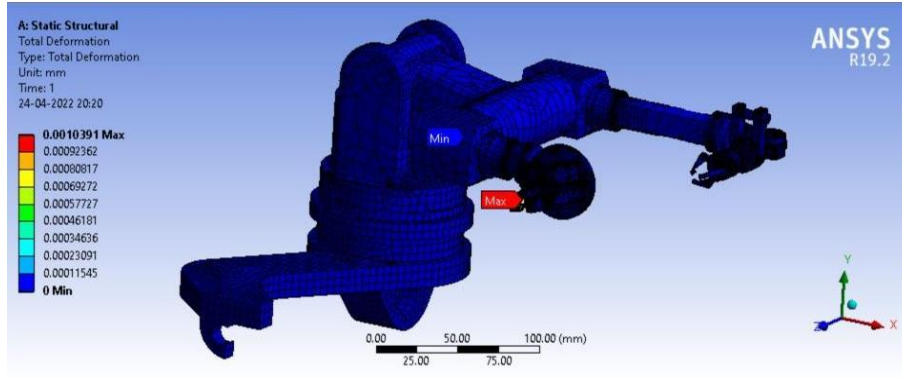
Figure 84: Deformation Plot for Safety Lever Manipulation

6.2.2 Breech Cover Opening

In this operation, the load was experienced by the claws holding the breech cover lever and opening it. A 0.225 Nm torque is involved in this task, and this load was applied on the system. As for constraints, the part of the carriage where it is coupled with the lead screw was assigned. The view of the assembly after meshing is shown below. The simulation results shown denote that the working stress and corresponding deformation are well within the yield limits of the considered material, thus substantiating the validity of the design parameters. The maximum value of 5.49 MPa Von Mises stress (Figure 85), and 9.49E-5 mm deformation (Figure 86) are both found occurring at the claw of the end effector.
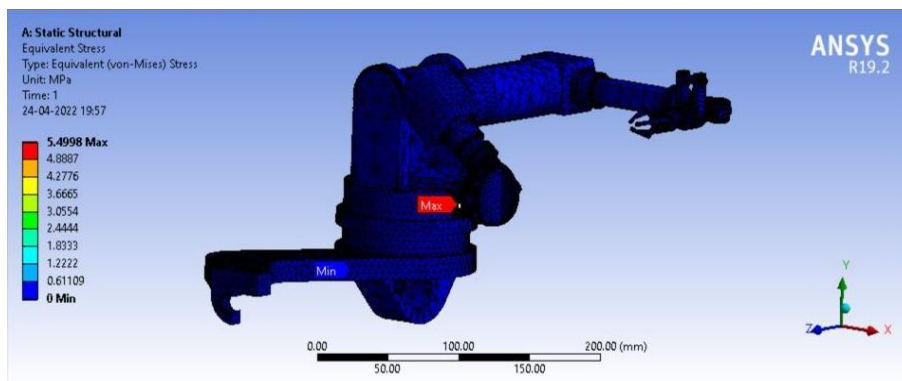
Figure 85: Stress Plot for Breech Cover Opening
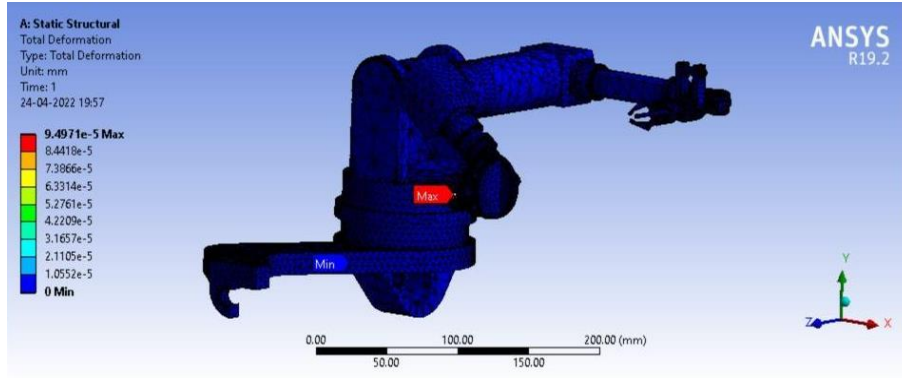
Figure 86: Deformation Plot for Breech Cover Opening

6.2.3 Belt Retrieval

This analysis considers the right arm end effector as the point of force application. In this operation, the load was experienced by the jaws of the gripper holding onto the ammunition belt and lifting it. The weight of the belt in 5 N along the negative Y-axis direction, was given as the force applied on the system. As for constraints, the part of the carriage where it is coupled with the lead screw was assigned. The view of the assembly after meshing is shown below. The simulation results shown denote that the working stress and corresponding deformation are well within the yield limits of the considered material, thus substantiating the validity of the design parameters. The maximum value of 30.97 Von Mises stress was found occurring at the joint between the jaw and the gripper drive shaft (Figure 87), while the maximum deformation of 0.082 mm (Figure 88) was found at the tip of the gripper in contact with the belt.
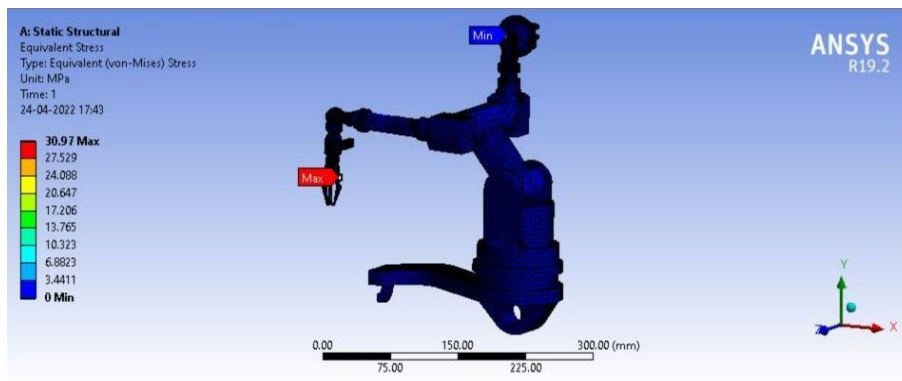
Figure 87: Stress Plot for Belt Retrieval
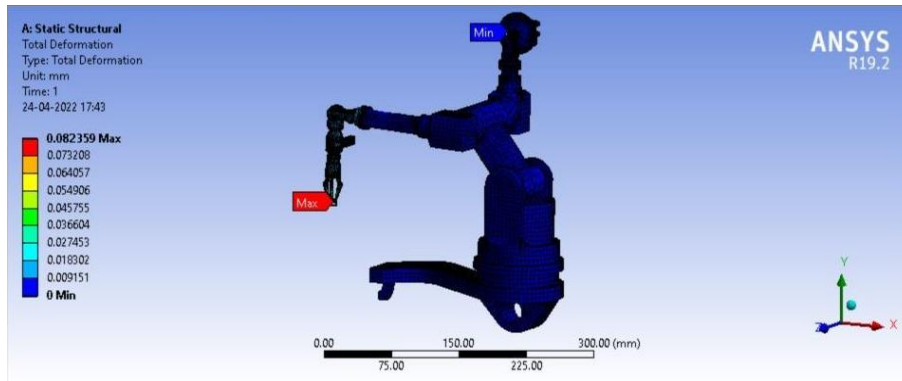
Figure 88: Deformation Plot for Belt Retrieval

CHAPTER 7

SUMMARY

This project has provided a comprehensive documentation of a specialised 7-DOF robotic arm system, right from the conceptualization and design up to simulation and analysis of its motion in the scope of kinematics and dynamics. The system has been completely modelled towards performing the multitude of operations in sequence, as entailed by the safe retrieval process of ammunition from a heavy machine gun. The form, function, and degrees of freedom of the mechanism are validated by means of the results obtained from the simulations using MATLAB and ADAMS software. The study progressed from initially identifying each individual arm's range of motion, and calculating the critical points with reference to the gun, strategically chosen to verify the capability of the arm to reach the positions that it needs to, in order to serve its purposes. Further, the forward and inverse kinematics calculations provided the basis for developing trajectories, cartesian velocities, and subsequently, joint velocities and joint accelerations. Using those, the inverse dynamics was undertaken to gauge the torque, force and power requirements of the system, which enabled the identification of motor dimensions for the same. Finally, based on the loads experienced in operation, finite element analyses have been carried out in critical areas, to serve as validation for the structural integrity of the bodies.

In conclusion, it is postulated that this project presents a viable solution towards the problem statement posed by the Ministry of Defence, India. A patent for this robotic arm system has been drafted under the guidance of CVRDE scientists, and is currently put up for approval at CVRDE. It is intended that this dissertation may provide the impetus for further prototyping and manufacturing of this system, and that it may prove to augment the safety of the soldiers of Defence Forces worldwide.

REFERENCES

➢ Pawat Chusilp, Weerawut Charubhun, and Artit Ridluan (2011), Developing Firing Table Software for Artillery Projectiles using Iterative Search and 6-DOF Trajectory Model, *The Second TSME International Conference on Mechanical Engineering*, AME04.

➢ Qiang Liu, Kedong Zhou, Chao Shen, Lei He (2021), Effect of an Internal Impact Balance Mechanism on the Perceptible Recoil of Machine Gun, *E3S Web of Conferences*, 231(1):03001.

➢ Yifang Sun, Li Li, Feng Xie (2020), Research on recoil dynamic characteristics of a gatling gun automatic machine, *Journal of Physics Conference Series*, 1345(3):032042.

➢ Mingbo Piao, Genglin Dai, Guangshuo Ding (2020), Gait planning and simulation of quadruped robot climbing one-step stair, *IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, pp. 797-802.

➢ Sharath Surati, Shaunak Hedaoo, Tushar Rotti, Vaibhav Ahuja, Nishigandha Patel (2021), Pick and Place Robotic Arm: A Review Paper, *International Research Journal of Engineering and Technology*, 08(02).

➢ Ingrid van Haaren, PWM van Zutven (2012), Design of a robot arm, *Eindhoven University of Technology*, Eindhoven, Netherlands.

➢ Jie Cai, Jinlian Deng, Wei Zhang, Weisheng Zhao (2021), Modeling Method of Autonomous Robot Manipulator Based on D-H Algorithm, *Artificial Intelligence and Edge Computing in Mobile Information Systems*, Article ID 4448648.

➢ Siddarth Oli, Mark, Lown, Frederic H. Moll, Enrique Romo, David S. Mintz, and Allen Jiang (2016), Configurable robotic surgical system with virtual rail and flexible endoscope, WO2016054256A1.

➢ Midriem Mirdanies, Ary Setijadi Prihatmanto, Estiko Rijanto (2013), Object Recognition System in Remote Controlled Weapon Station using SIFT and SURF Methods, *Mechatronics, Electrical Power, and Vehicular Technology,* 04, pp. 99-108.

➢ M. Lesser (2014), 3 - Charge coupled device (CCD) image sensors, *High Performance Silicon Imaging*, pp. 78-97.

➢ Tom Harris and Chris Pollette (2022), How Robots Work.

➢ Daniel Huczala, Tomáš Kot, Martin Pfurner, Dominik Heczko, Petr Oščádal, and Vladimír Mostýn (2021), Initial Estimation of Kinematic Structure of a Robotic Manipulator as an Input for Its Synthesis, *Innovative Robot Designs and Approaches*, 11(8), 3548.

➢ Hang, Han Yin and Wang Ying, Lian Shiqi (2017), Inverse kinematics solution system for use with robot, WO2018205707A1

➢ Natesan Babu (2018), A Vehicle and Method for Detecting and Neutralizing an Incendiary Object, *United States Patent Application Publication,* US 2018 / 0252503 A1.

➢ Microbot Inc (1989), Robotic Arm, United States Patent, 4,806,066.

➢ Peter Movsesian (1995), Mobile Robotic Arm, United States Patent, 5,413,454.

➢ Karen Shakespear Koenig, Pablo Eduardo, Garcia Kilroy, Sina Nia Kosari, Thomas D. Egan,(2020), Robotic Arm And Robotic Surgical System, US 2021/0045817 A1.

➢ Hai-Wu Lee (2020), Study of a Mechanical Arm and Intelligent Robot, IEEE Access pp. (99):1-1.

➢ Paul Bacchi and Paul S Filipski (1998), Specimen holding robotic arm end effector, WO2000033359B1

➢ Multibody Dynamics Simulation of an Electric Bus, Ricardo R. Teixeira et al, December 2015, Procedia Engineering 114:470-477, 10.1016/j.proeng.2015.08.094

➢ ADAMS model validation for an all-terrain vehicle using test track data, Husain Kanchwala and Anindya Chatterjee, Advances in Mechanical Engineering, 2019, Vol. 11(7) 1–18, DOI: 10.1177/1687814019859784

APPENDIX A

The following program is run to determine the Forward Kinematics of the Internal Arm:

```
clear all

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, -62, 82, 0, 0],'modified');
L(5) = Link([0, 120, 0, pi/2, 0],'modified');
L(6) = Link([0, 100, 0, -pi/2, 0],'modified');

L(1).qlim = [75 410];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

qf = [0 pi/2 0 pi/2 pi/2 0]
Tf = CNN.fkine(qf)
CNN.plot(qf,'workspace',[-200 200 -400 400 -400 400],'scale',0.4)
```

APPENDIX B

The following program is run to determine the Forward Kinematics of the External Arm:

```
clc;
clear all

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, 62, 82, 0, 0],'modified');
L(5) = Link([0, 0, 72, 0, 0],'modified');
L(6) = Link([0, 55, 0, 0, 0],'modified');

L(1).qlim = [75 410];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

qf = [0 pi/2 0 pi/2 0 0]
Tf = CNN.fkine(qf)
CNN.plot(qf,'workspace',[-200 200 -400 400 -400 400],'scale',0.4)
```

APPENDIX C

The following program is run to determine the Forward Kinematics motion of the Internal Arm:

```
clear all

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, -62, 82, 0, 0],'modified');
L(5) = Link([0, 120, 0, pi/2, 0],'modified');
L(6) = Link([0, 100, 0, -pi/2, 0],'modified');

L(1).qlim = [75 410];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

for l1 = 0:20:335
    for th2 = 0:0.2:pi/2
        for th3 = -pi/2:0.4:pi/2
            for th4 = 0:0.4:pi
                for th5 = -pi/2:0.4:pi/2
                    for th6 = 0:0
                        CNN.plot([l1 th2 th3 th4 th5 th6], 'workspace',[-200
200 -400 400 -400 400],'scale',0.4)

                    end
                end
            end
        end
    end
end
```

APPENDIX D

The following program is run to determine the Forward Kinematics motion of the External Arm:

```
clear all

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, 62, 82, 0, 0],'modified');
L(5) = Link([0, 0, 72, 0, 0],'modified');
L(6) = Link([0, 55, 0, 0, 0],'modified');

L(1).qlim = [75 410];

CNN = SerialLink(L)
```

```
CNN.name = 'CNN Robot';

for l1 = 0:20:335
    for th2 = 0:0.2:pi/2
        for th3 = -pi/2:0.4:pi/2
            for th4 = -pi/2:0.4:pi/2
                for th5 = 0:0.4:pi
                    for th6 = 0:0
                        CNN.plot([l1 th2 th3 th4 th5 th6], 'workspace',[-200
200 -400 400 -400 400],'scale',0.4)

                    end
                end
            end
        end
    end
end
```

APPENDIX E

The following program is run to determine the Inverse Kinematics of the Internal Arm (LE1):

```
clear all

syms px py pz

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, -62, 82, 0, 0],'modified');
L(5) = Link([0, 120, 0, pi/2, 0],'modified');
L(6) = Link([0, 100, 0, -pi/2, 0],'modified');

L(1).qlim = [75 410];
L(2).qlim = [0 0];
L(3).qlim = [-pi/2 pi/2];
L(4).qlim = [0 pi];
L(5).qlim = [-pi/2 pi/2];
L(6).qlim = [0 0];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

px= 166
py= -89
pz= 190

T = [1 0 0 px;0 0 1 py; 0 -1 0 pz; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 pi/2 0])
CNN.plot(Ti,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
```

APPENDIX F

The following program is run to determine the Inverse Kinematics of the Internal Arm (LE2):

```
clear all

syms px py pz

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, -62, 82, 0, 0],'modified');
L(5) = Link([0, 120, 0, pi/2, 0],'modified');
L(6) = Link([0, 100, 0, -pi/2, 0],'modified');

L(1).qlim = [75 410];
L(2).qlim = [0 0];
L(3).qlim = [-pi/2 pi/2];
L(4).qlim = [0 pi];
L(5).qlim = [pi/2 pi/2];
L(6).qlim = [0 0];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

px= 40
py= -89
pz= 190

T = [1 0 0 px;0 0 1 py; 0 -1 0 pz; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 pi/2 0])
CNN.plot(Ti,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
```

APPENDIX G

The following program is run to determine the Inverse Kinematics of the Internal Arm (LE3):

```
clear all

syms px py pz
```

```
L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, -62, 82, 0, 0],'modified');
L(5) = Link([0, 120, 0, pi/2, 0],'modified');
L(6) = Link([0, 100, 0, -pi/2, 0],'modified');


L(1).qlim = [75 410];
L(2).qlim = [0 0];
L(3).qlim = [-pi/2 pi/2];
L(4).qlim = [0 pi];
L(5).qlim = [0 0];
L(6).qlim = [0 0];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';


px= 191
py= -60
pz= 81


T = [1 0 0 px;0 0 1 py; 0 -1 0 pz; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 0 0])
CNN.plot(Ti,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
```

APPENDIX H

The following program is run to determine the Inverse Kinematics of the External Arm (RE1):


```
clear all

syms px py pz

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, 62, 82, 0, 0],'modified');
L(5) = Link([0, 0, 72, 0, 0],'modified');
L(6) = Link([0, 55, 0, 0, 0],'modified');


L(1).qlim = [75 410];
L(2).qlim = [0 pi/2];
```

```
L(3).qlim = [-pi/2 pi/2];
L(4).qlim = [-pi/2 pi/2];
L(5).qlim = [0 pi];
L(6).qlim = [0 0];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

px= 117
py= -76
pz= 395

T = [0 0 1 px;-1 0 0 py; 0 -1 0 pz; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 0 0])
CNN.plot(Ti,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
```

APPENDIX I

The following program is run to determine the Inverse Kinematics of the External Arm (RE2):

```
clear all

syms px py pz

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, 62, 82, 0, 0],'modified');
L(5) = Link([0, 0, 72, 0, 0],'modified');
L(6) = Link([0, 55, 0, 0, 0],'modified');

L(1).qlim = [75 410];
L(2).qlim = [0 pi/2];
L(3).qlim = [-pi/2 pi/2];
L(4).qlim = [-pi/2 pi/2];
L(5).qlim = [0 pi];
L(6).qlim = [0 0];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

px= 117
py= -105
pz= 250

T = [0 0 1 px;-1 0 0 py; 0 -1 0 pz; 0 0 0 1]
```

```
Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 0 0])
CNN.plot(Ti,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
```

APPENDIX J

The following program is run to determine the Inverse Kinematics of the External Arm (RE3):

```
clear all

syms px py pz

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, 62, 82, 0, 0],'modified');
L(5) = Link([0, 0, 72, 0, 0],'modified');
L(6) = Link([0, 55, 0, 0, 0],'modified');

L(1).qlim = [75 410];
L(2).qlim = [0 pi/2];
L(3).qlim = [-pi/2 pi/2];
L(4).qlim = [-pi/2 pi/2];
L(5).qlim = [0 pi];
L(6).qlim = [0 0];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

px= 117
py= -80
pz= 240

T = [0 0 1 px;-1 0 0 py; 0 -1 0 pz; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 0 0])
CNN.plot(Ti,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
```

APPENDIX K

The following program is run to determine the full Trajectory of the Internal Arm:

```
clear all

syms px1 py1 pz1 px2 py2 pz2 px3 py3 pz3

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, -62, 82, 0, 0],'modified');
L(5) = Link([0, 120, 0, pi/2, 0],'modified');
L(6) = Link([0, 100, 0, -pi/2, 0],'modified');

L(1).qlim = [75 410];
L(2).qlim = [0 0];
L(3).qlim = [-pi/2 pi/2];
L(4).qlim = [0 pi];
L(5).qlim = [-pi/2 pi/2];
L(6).qlim = [0 0];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

px= 166
py= -89
pz= 190

T = [1 0 0 px;0 0 1 py; 0 -1 0 pz; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 pi/2 0])

qf = [0 pi/2 0 pi/2 pi/2 0]
qi = [Ti]
t = 0:0.15:3
Q = jtraj(qf,qi,t)
Ttraj = fkine(CNN,Q)

for i = 1:1:length(t)
    A = Ttraj(i);
    B = transl(A);
    xx(i)=B(1);
    yy(i)=B(2);
    zz(i)=B(3);
end

plot(CNN,Q,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
hold on
plot3(xx,yy,zz,'Color',[0 1 1],'LineWidth',3)

hold on
```

```
px= 40
py= -89
pz= 190

T = [1 0 0 px;0 0 1 py; 0 -1 0 pz; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 pi/2 0])

qf = [0 pi/2 0 pi/2 pi/2 0]
qi = [Ti]
t = 0:0.15:3
Q = jtraj(qf,qi,t)
Ttraj = fkine(CNN,Q)

for i = 1:1:length(t)
    A = Ttraj(i);
    B = transl(A);
    xx(i)=B(1);
    yy(i)=B(2);
    zz(i)=B(3);
end

plot(CNN,Q,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
hold on
plot3(xx,yy,zz,'Color',[0 0 1],'LineWidth',3)

hold on

px= 191
py= -60
pz= 81

T = [1 0 0 px;0 0 1 py; 0 -1 0 pz; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 0 0])

qf = [0 pi/2 0 pi/2 pi/2 0]
qi = [Ti]
t = 0:0.15:3
Q = jtraj(qf,qi,t)
Ttraj = fkine(CNN,Q)

for i = 1:1:length(t)
    A = Ttraj(i);
    B = transl(A);
    xx(i)=B(1);
    yy(i)=B(2);
    zz(i)=B(3);
end

plot(CNN,Q,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
hold on
plot3(xx,yy,zz,'Color',[1 0 0],'LineWidth',3)
```

APPENDIX L

The following program is run to determine the full Trajectory of the External Arm:

```
clear all

syms px1 py1 pz1 px2 py2 pz2 px3 py3 pz3

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, 62, 82, 0, 0],'modified');
L(5) = Link([0, 0, 72, 0, 0],'modified');
L(6) = Link([0, 55, 0, 0, 0],'modified');

L(1).qlim = [75 410];
L(2).qlim = [0 pi/2];
L(3).qlim = [-pi/2 pi/2];
L(4).qlim = [-pi/2 pi/2];
L(5).qlim = [0 pi];
L(6).qlim = [0 0];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

px1= 117
py1= -76
pz1= 395
240

T = [0 0 1 px1;-1 0 0 py1; 0 -1 0 pz1; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 0 0])

qf = [0 pi/2 0 pi/2 0 0]
qi = [Ti]
t = 0:0.15:3
Q = jtraj(qf,qi,t)
Ttraj = fkine(CNN,Q)

for i = 1:1:length(t)
    A = Ttraj(i);
    B = transl(A);
    xx(i)=B(1);
    yy(i)=B(2);
    zz(i)=B(3);
end

plot(CNN,Q,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
hold on
```

```matlab
plot3(xx,yy,zz,'Color',[0 1 0],'LineWidth',3)

hold on

px2= 117
py2= -105
pz2= 250

T = [0 0 1 px2;-1 0 0 py2; 0 -1 0 pz2; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 0 0])
CNN.plot(Ti,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)

qf = [0 pi/2 0 pi/2 0 0]
qi = [Ti]
t = 0:0.15:3
Q = jtraj(qf,qi,t)
Ttraj = fkine(CNN,Q)

for i = 1:1:length(t)
    A = Ttraj(i);
    B = transl(A);
    xx(i)=B(1);
    yy(i)=B(2);
    zz(i)=B(3);
end

plot(CNN,Q,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
hold on
plot3(xx,yy,zz,'Color',[0 0 1],'LineWidth',3)

hold on

px3= 117
py3= -80
pz3= 240

T = [0 0 1 px3;-1 0 0 py3; 0 -1 0 pz3; 0 0 0 1]

Ti = CNN.ikcon(T,[0 pi/2 0 pi/2 0 0])
CNN.plot(Ti,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)

qf = [0 pi/2 0 pi/2 0 0]
qi = [Ti]
t = 0:0.15:3
Q = jtraj(qf,qi,t)
Ttraj = fkine(CNN,Q)

for i = 1:1:length(t)
    A = Ttraj(i);
    B = transl(A);
    xx(i)=B(1);
    yy(i)=B(2);
    zz(i)=B(3);
end

plot(CNN,Q,'workspace',[-200 200 -400 500 -400 500],'scale',0.4)
```

```
hold on
plot3(xx,yy,zz,'Color',[1 1 0],'LineWidth',3)
```

APPENDIX M

The following program is run to determine the Jacobian of the Robotic Arm:

```
clear all
syms px py pz

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, 62, 82, 0, 0],'modified');
L(5) = Link([0, 0, 72, 0, 0],'modified');
L(6) = Link([0, 55, 0, 0, 0],'modified');

L(1).qlim = [75 410];
L(2).qlim = [0 pi/2];
L(3).qlim = [-pi/2 pi/2];
L(4).qlim = [-pi/2 pi/2];
L(5).qlim = [0 pi];
L(6).qlim = [0 0];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

qf = [0 pi/2 0 pi/2 0 0]

J=jacob0(CNN,qf)
rank(J)
det(J)
```

APPENDIX N

The following program is run to determine the Workspace of the Internal Arm:

```
clear all

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, -62, 82, 0, 0],'modified');
L(5) = Link([0, 120, 0, pi/2, 0],'modified');
L(6) = Link([0, 100, 0, -pi/2, 0],'modified');
```

```
L(1).qlim = [75 410];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

i=0;
for l1 = 0:20:335
    for th2 = 0:0.2:pi/2
        for th3 = -pi/2:0.4:pi/2
            for th4 = 0:0.4:pi
                for th5 = -pi/2:0.4:pi/2
                    for th6 = 0:0
                        %CNN.plot([l1 th2 th3 th4 th5 th6], 'workspace',[-200
200 -400 400 -200 200])
                        T01= trotz (l1)*transl(0,0,75)*trotx (pi/2);
                        T12= trotz (th2)*transl(0,pi/2,135)*trotx (pi/2);
                        T23= trotz (th3)*transl(82,0,0);
                        T34= trotz (th4)*transl(72,pi/2,62);
                        T45= trotz (th5)*transl(0,0,0);
                        T56= trotz (th6)*transl(0,0,55);
                        T06 = T01*T12*T23*T34*T45*T56;
                        i=i+1;
                        p=T06(1:3,4);
                        p1(i)=p(1);
                        p2(i)=p(2);
                        p3(i)=p(3);
                    end
                end
            end
        end
    end
end

figure (1)
plot ((p3.^2 + p2.^2).^0.5, p1, 'b.');
xlabel('y')
ylabel('z')

figure (2)
plot ((p2.^2 + p1.^2).^0.5, p3, 'b.');
xlabel('x')
ylabel('z')

figure (3)
plot ((p1.^2 + p3.^2).^0.5, p2, 'b.');
xlabel('x')
ylabel('y')
```

APPENDIX O

The following program is run to determine the Workspace of the External Arm:

```
clear all

L(1) = Link([0, 75, 0, 0, 1],'modified');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'modified');
L(3) = Link([0, 0, 0, pi/2, 0],'modified');
L(4) = Link([pi/2, 62, 82, 0, 0],'modified');
L(5) = Link([0, 0, 72, 0, 0],'modified');
L(6) = Link([0, 55, 0, 0, 0],'modified');

L(1).qlim = [75 410];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

i=0;
for l1 = 0:20:335
    for th2 = 0:0.2:pi/2
        for th3 = -pi/2:0.4:pi/2
            for th4 = -pi/2:0.4:pi/2
                for th5 = 0:0.4:pi
                    for th6 = 0:0
                        T01= trotz (l1)*transl(0,0,75)*trotx (pi/2);
                        T12= trotz (th2)*transl(0,pi/2,135)*trotx (pi/2);
                        T23= trotz (th3)*transl(82,0,0);
                        T34= trotz (th4)*transl(72,pi/2,62);
                        T45= trotz (th5)*transl(0,0,0);
                        T56= trotz (th6)*transl(0,0,55);
                        T06 = T01*T12*T23*T34*T45*T56;
                        i=i+1;
                        p=T06(1:3,4);
                        p1(i)=p(1);
                        p2(i)=p(2);
                        p3(i)=p(3);
                    end
                end
            end
        end
    end
end

figure (1)
plot ((p3.^2 + p2.^2).^0.5, p1, 'b.');
xlabel('y')
ylabel('z')

figure (2)
plot ((p2.^2 + p1.^2).^0.5, p3, 'b.');
xlabel('x')
ylabel('z')

figure (3)
plot ((p1.^2 + p3.^2).^0.5, p2, 'b.');
xlabel('x')
ylabel('y')
```

APPENDIX P

The following program is run to determine the Singularity of the Robotic Arm:

```
clear all

syms sing

L(1) = Link([0, 75, 0, pi/2, 1],'standard');
L(2) = Link([pi/2, 135, 0, pi/2, 0],'standard');
L(3) = Link([0, 0, 82, 0, 0],'standard');
L(4) = Link([pi/2, 62, 72, 0, 0],'standard');
L(5) = Link([0, 0, 0, 0, 0],'standard');
L(6) = Link([0, 55, 0, 0, 0],'standard');

L(1).qlim = [75 410];

CNN = SerialLink(L)
CNN.name = 'CNN Robot';

sing = 0;
i=0;
for l1 = 0:20:335
    for th2 = 0:0.2:pi/2
        for th3 = -pi/2:0.4:pi/2
            for th4 = -pi/2:0.4:pi/2
                for th5 = 0:0.4:pi
                    for th6 = 0:0
                        qf = [l1 th2 th3 th4 th5 th6];
                        J=jacob0(CNN,qf);
                        rank(J);
                        det(J);
                        if det(J)==0
                            sing = sing + 1;
                        end
                    end
                end
            end
        end
    end
end

disp(sing)
```