ENPM662 Project 2 Final Report

<u>Laparoscopic Surgical Robot having 6 dof Manipulator and 2 dof Surgical Instrument</u>

Member 1: Aditya Varadaraj (117054859)

Member 2: Saurabh Prakash Palande (118133959)

Table of Contents

Table of Figures

1. Introduction

Nowadays, Industrial Manipulators are being used for a wide range of applications from factories and warehouses to surgery. In the past few years, a very significant and constant development has been observed in the flexibility, versatility, dexterity, precision and accuracy of industrial manipulators, thanks to Industry 4.0. Due to this increase in performance of industrial manipulators, they are being widely used to perform intricate medical procedures. We propose to apply our knowledge of robot modelling to one such application of industrial manipulator to laparoscopic surgery. Laparoscopic Surgery is a surgical technique in which small, narrow tube-like instruments (trochars) and probes with cameras are inserted into the body (especially abdominal and prostate area) via narrow cuts that allows them to manipulate, cut and sew tissue [3]. Without these tools, it would have taken wider and more no. of cuts for the surgeon to operate on the patient.

We will first explain the significance of our chosen application in section 2. Then, in section 3, we will discuss the robot type. Then, in section 4, we will further discuss the degrees of freedom (dofs) of the manipulator and the instrument and their dimensions. In section 5, we discuss the CAD model. In section 6, we will highlight the DH parameters and frames chosen for the same. In section 7, the final transformation matrix at the start position will be shown. In section 8, the Jacobian matrix at the start configuration will be shown. In section 9, forward and inverse kinematics validation will be discussed. In section 10, we show the workspace for the system. In section 11, we state the assumptions made in our model. In section 12, we discuss the control method chosen. In section 13, we will provide link to the working video of the simulation of robot following the desired trajectory for the task. In section 14, we will list the problems faced. In section 15, we will highlight the lessons learned. In section 16, we validate the results and draw conclusions. In Section 17, we discuss possible future work.

2. Application

Nowadays, Minimally Invasive Surgery is gaining a lot of popularity. It helps surgeons perform surgery with better accuracy and precision which cannot be achieved by human surgery. Laparoscopic Surgery is widely used in Prostatectomy, GallBladder surgery, surgery related to liver, pancreas, small intestine, appendix and sometimes also in repairing bleeding valves in heart [4]. Laparoscopic Surgery has following advantages compared to conventional surgery [5]:

- Smaller Scars are required.

- Less recovery/healing time.
- The patient feels less pain while the scars heal.
- Less internal scarring

Robots are more flexible compared to the human arm thus allowing more degrees of freedom which helps to reach conventionally inaccessible areas with high dexterity and precision. They still allow some level of control to the surgeon with haptic feedback devices to perform the procedures.

In our application, we are using a 6 dof manipulator alongwith a 2 dof surgical instrument which is to be used for laparoscopic surgery. The proposed robot has a total 8 degrees of freedom (6 of the manipulator and 2 of the surgical instrument) which enables the robot to reach narrow and unreachable areas. The extra 2 dofs of the instrument allow the surgeon to manipulate tissues in constrained small spaces. Thus, the surgery requires only a small incision on the body.

3. Robot Type

We are using a manipulator.

4. DOFs and Dimensions

A 6 d.o.f. manipulator (KUKA KR120 2700) was selected for the translational motion of the 2 d.o.f. surgical instrument. So in all, the setup has 8 d.o.f.

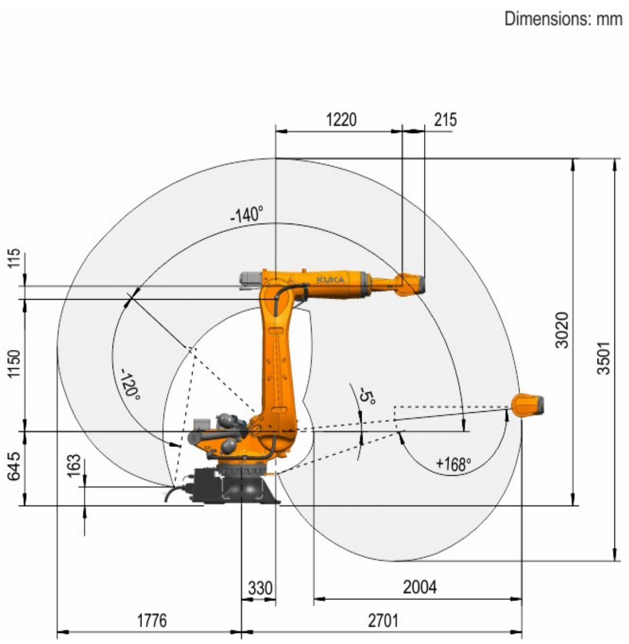


Figure 1: Dimensions of KUKA KR-120 2700[2]

In figure 1, the dimensions we have referred from KUKA KR120 2700 datasheet are shown. Joint 1 - Joint 5 of the external arm contributes to the translational motion of the surgical instrument while Joint 6 contributes to the rolling of the surgical instrument. The surgical instrument itself has 2 degrees of freedom about J7 and J8 (in Figure 2) which contribute to the yawing motion and pitching motion of the surgical instrument respectively. The surgical instrument has a pneumatic gripper which uses inflatable balloons to grip objects.

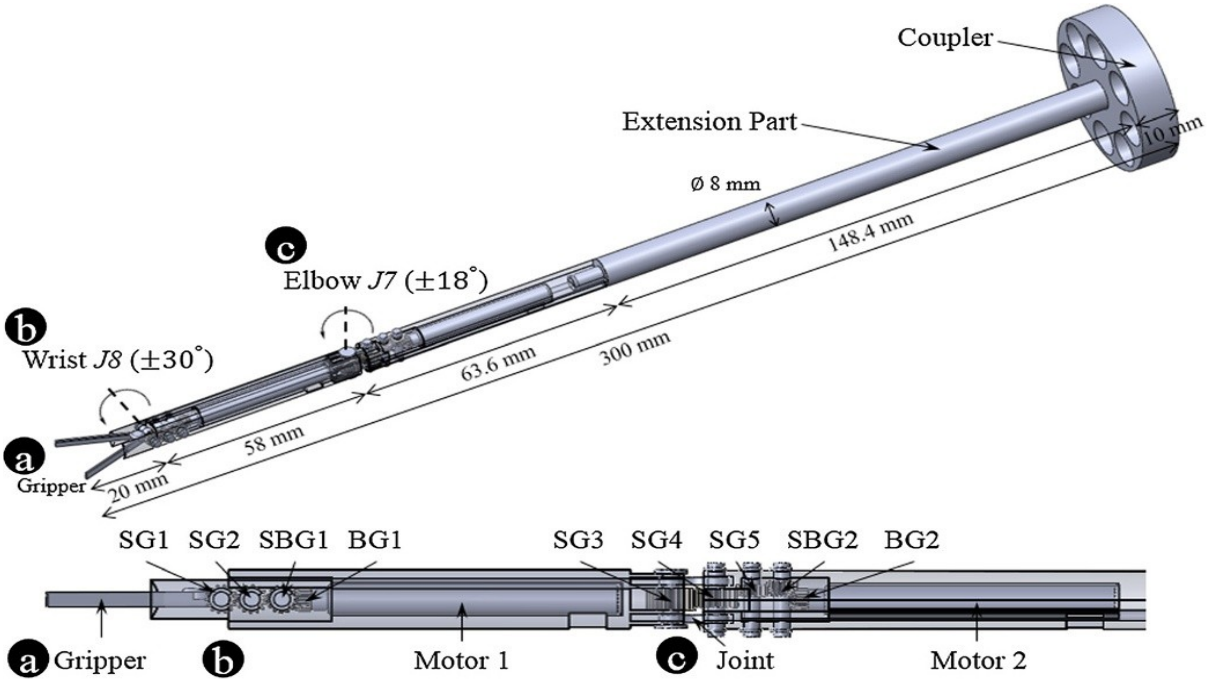


Figure 2: Dimensions of Surgical Instrument[1]

5. CAD Model

The entire CAD parts and assembly have been designed by us in SOLIDWORKS. We have not used any external resources for the same. The part files and assembly file have been attached in the submission.

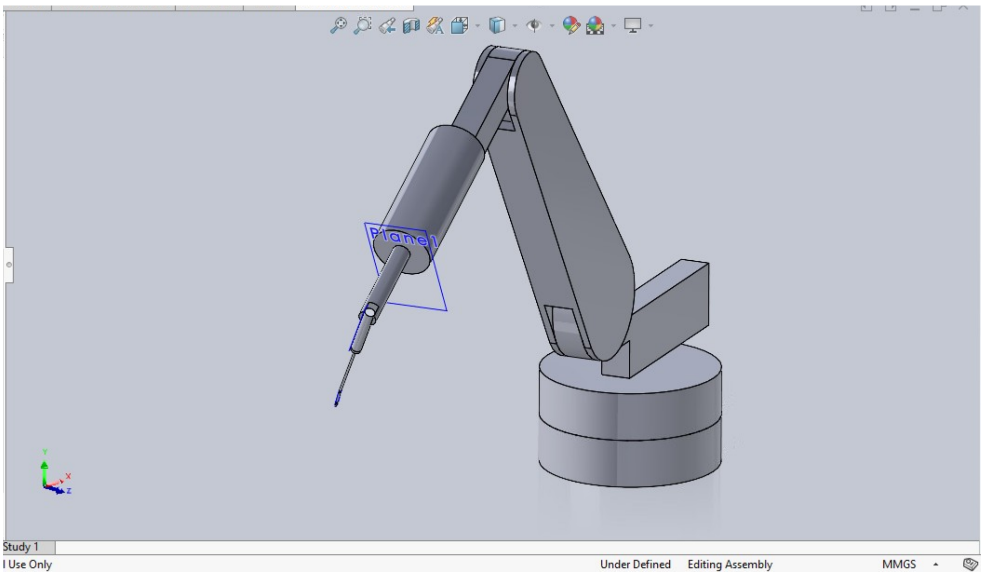


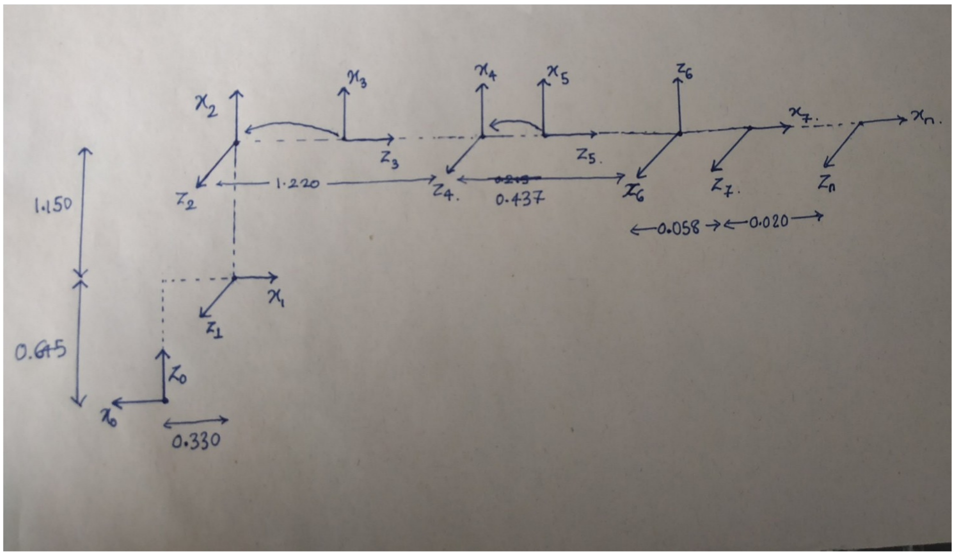
Figure 3: CAD Assembly

6. DH Parameters

The DH parameters are shown in the table below:

Link_i	a_i (m)	Alpha_i (degrees)	d_i (m)	Theta_i (degrees)
0 ->1	0.330	90	0.645	180+theta_1
1 ->2	1.15	0	0	90 + theta_2
2 ->3	0	90	0	theta_3
3 ->4	0	-90	1.22	theta_4
4 ->5	0	90	0	theta_5
5 ->6	0	90	0.437	90 + theta_6
6 ->7	0.058	90	0	90 + theta_7
7 ->n	0.02	0	0	theta_8

These DH parameters were obtained by considering frames according to Spong convention as shown in Drawing 1



Drawing 1: DH Frames

7. Forward Velocity Kinematics

Based on the DH parameters, we calculate the Transformation matrices using Sympy in the python code provided. Since it is a 8 d.o.f. system, it is difficult to make sense of the final transformation matrix and the Jacobian in its parametric form, we are printing the values at initial configuration for the sake of this report. At the initial position, the Transformation Matrix is printed as:

Final Transformation Matrix at start configuration:

Matrix([

[-0.252048316318927, -0.96771465124839, 3.48208312818468e-17, -1.54097244214484],

[2.12587128802748e-16, -1.93873230913241e-17, 1.0, 2.92422103840674e-16],

[-0.96771465124839, 0.252048316318927, 2.10610221352347e-16, 0.764329514302685],

[0, 0, 0, 1]])

8. Inverse Velocity Kinematics

The Jacobian is calculated using Sympy and the 1st method taught in class using cross products. The Jacobian matrix at the initial configuration is:

Jacobian Matrix at start configuration:

Matrix([

[-2.92422103840674e-16, 0.119329514302685, 0.00799044145567452, -1.07713058535892e-16, -0.498373045392921, 4.62192373554343e-18, -7.33794857552748e-18, -0.0193542930249678],

[-1.54097244214484, -8.87643270298757e-17, -7.68918328526846e-17, -0.507294418124996, 5.30848387640567e-17, -1.90819582357449e-17, -0.078, -3.87746461826483e-19],

[0, 1.21097244214484, 1.23975804371174, -5.61500285039659e-17, 0.129804882904248, -1.20381363886077e-18, -1.52237836266223e-17, 0.00504096632637854],

[0, 1.22464679914735e-16, 1.22464679914735e-16, -0.909797672793022, 1.22464679914735e-16, -0.252048316318927, -0.96771465124839, 3.48208312818468e-17],

[0, 1.0, 1.0, 1.47235813227799e-16, 1.0, 1.5135478884538e-16, 4.18450168660435e-17, 1.0],

[1, 6.12323399573677e-17, 6.12323399573677e-17, 0.415052038400488, 6.12323399573677e-17, -0.96771465124839, 0.252048316318927, 2.10610221352347e-16]])

9. Forward and Inverse Velocity Kinematics Validation

We have created a Sympy code similar to HW4 to make the robot move in a straight line in XZ plane with a slope. The output of the code in matplotlib is as shown in the Figure 4

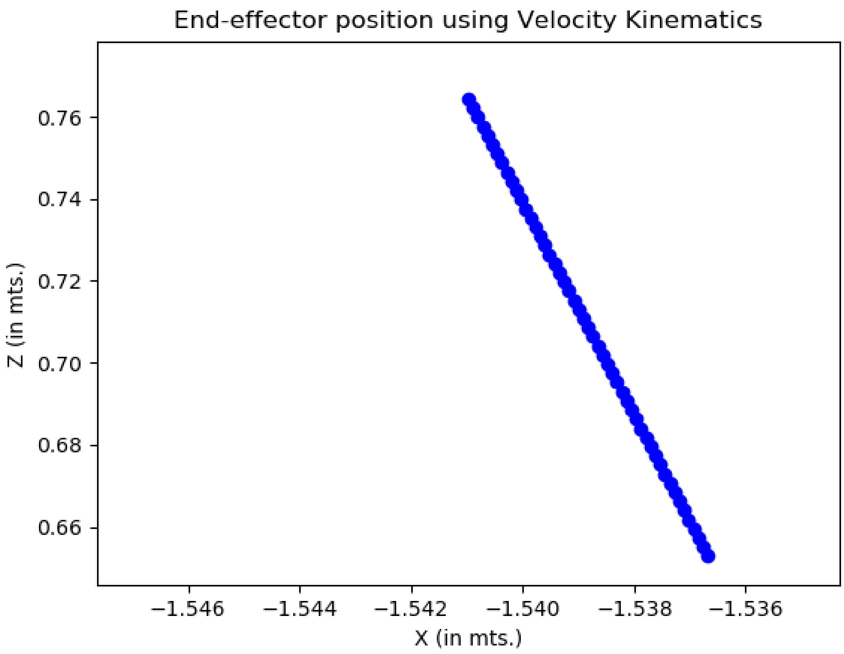


Figure 4: Validation Plot for end-effector position

10. Workspace Study

We plotted the points by incrementing various joint angles to get a set of points in the workspace using final transformation matrix. The python code is attached in the submission. The resulting scatter plot of the workspace is as shown in Figure 5. As expected, it's spherical.

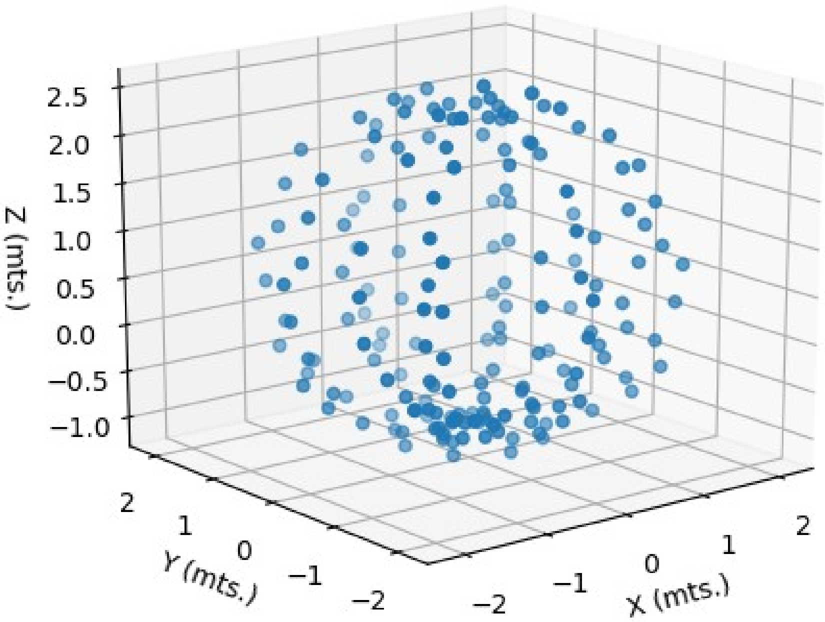


Figure 5: Workspace of the manipulator + instrument system

11. Assumptions

- All links and objects are rigid.
- We have assumed some frames to be clustered at the same position to find a solution for the 8 dof system.
- We do not consider the haptic joystick, various encoder and sensor outputs.
- Friction and other external disturbances are not taken into account.
- The solution for the path of the surgical instrument is not unique i.e there might be another optimal solution.
- Although the robot used in paper [1] is DENSO VS-6556G, we assume dimensions similar to KUKA KR120 2700 manipulator since it is more widely used.

12. Control Method

We used PID controller using config_controllers.yaml file in our catkin package.

13. MoveIt Simulation

Since we had some problems with controlling the robot in Gazebo due to gravity, we have used MoveIt as a simulation platform. We publish the joint anges at each iteration obtained in our code using Inverse Kinematics and the formula **q_current = q_previous + q_dot * dt** to MoveIt. We make the robot follow 3 consecutive trajectories: 1) A Line simulating the entering of the instrument into the cut in the body, 2) An Arc simulating the reorientation of the tool within the cut and 3) A Circle demonstrating the reachability of the instrument within the cut. **Link for the simulation video:** [Simulation Video]

The trajectory followed in simulation is shown in Figure 6.

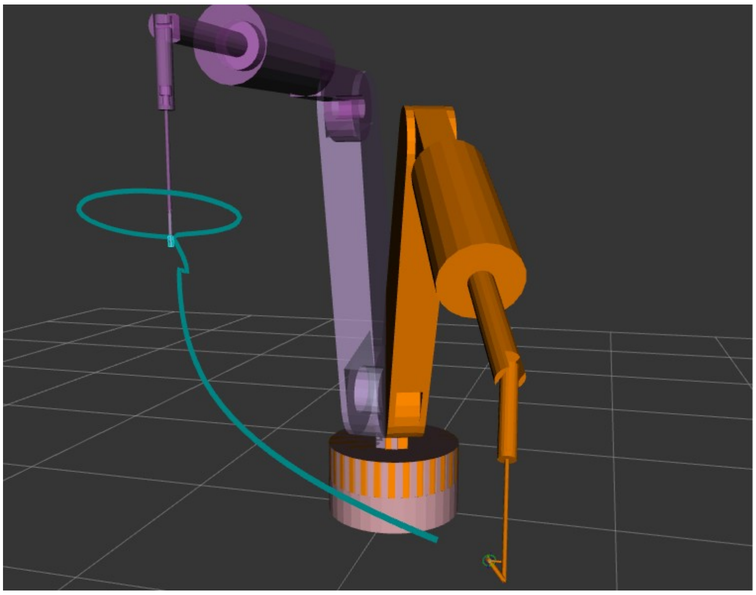


Figure 6: Trajectory followed in Simulation

Note that the manipulator follows the trajectory but does not take the configuration expected because this is using inverse kinematics which can have multiple solutions. Also, there are no workspace constraints.

14. Problems Faced

- Robot was falling in gazebo due to gravity. We added controllers and tried to tune the pid and joint effort values but tuning it for 8 joints was difficult so we used MoveIt instead of Gazebo.

- Model configuration in MoveIt was different than the home position used for calculation of DH parameters so we had to account for this joint angle errors.
- Moveit is showing the trajectory of the joint between end effector and the previous link so the path shown in Results section might be an approximation of the path followed by the tip of the actual end – effector.
- Since multiple solutions might exist for the inverse kinematics problem, the joint motion shown in Moveit might not be the one which is expected in surgery.

15. Lessons Learned

- Understood and derived the Forward and Inverse Kinematics of the manipulator and laparoscopic instrument system.
- Learnt how to simulate real-world tasks in MoveIt & Rviz.
- Validated and verified Forward and Inverse Kinematics by making the end-effector follow desired trajectories.

16. Conclusions

- Only kinematics is not enough to simulate the exact motion of the tip of the surgical instrument.
- Dynamics, Constraints in the workspace should also be taken into account.
- We used a basic PID controller for this project but in such applications where precision is of utmost importance advanced controllers are used.
- Since the robot has 8 DOF, there are multiple joint angle configurations which can give the same position of the tip of the end effector (surgical instrument).

17. Future Work

- We can try to implement the same in Gazebo and fix the gravity-related problems.
- We can try to make the manipulator perform a pick and place task.
- We can try to derive the dynamics of the robot.

References

[1] C. Lee *et al.*, "Pneumatic-type surgical robot end-effector for laparoscopic surgical-operation-by-wire," *Biomed. Eng. Online*, vol. 13, no. 1, 2014.

[2] KUKA KR 120 2700-2 Datasheet

[3] https://www.webmd.com/digestive-disorders/laparoscopic-surgery

[4] https://www.healthline.com/health/laparoscopy

[5] https://www.medstarfranklinsquare.org/our-services/surgical-services/treatments/robotic-surgery/how-does-robotic-surgery-differ-from-laparoscopic-surgery/