Kai Bing Goh, William Delaporte, Yi Gao & Yixiang Gao
981436, 915100, 1208446, 990803
Department of Mechanical Engineering
The University of Melbourne
VIC 3010, Australia
kaibingg@student.unimelb.edu.au
wdelaporte@student.unimelb.edu.au
yigao4@student.unimelb.edu.au
yixiang1@student.unimelb.edu.au

Robotics Systems Assignment 3 Report
Trajectory Generation

MCEN90028 : Dr. Justin Fong & Dr. Tomislav Bacek
Due –*Friday, May 6, 2022*

Table of Content

Introduction

The purpose of this report is to illustrate the process of generating the trajectory for the end-effector of the robot designed , as well as the validation of feasibility of generated trajectory. Figure 1 shows a top-down view of the work space.

The board spaces are each $4cm \times 4cm$ in size, and there is a $2cm$ border around the board. The base of the robot is positioned in the center (y=0) and $6cm$ below the border of the board. This gives the workspace a total length of $L = 52cm$, and a total width $W = 42cm$. The board is assumed to have a thickness of $1cm$, and the maximum height of a chess piece is $6cm$.

The inertial frame $\{0\}$ is positioned at the base of the robot, as shown in figure 1. The end-effector within in the ranges of $X = [6, 52]$ and $Y = [-26, 26]$. Vertically, the range is $Z = [0, 15]$, the feasibility of which was validated in assignment 1. Noticeable,the inertial frame we set is different from that in assignment 3 illustration.

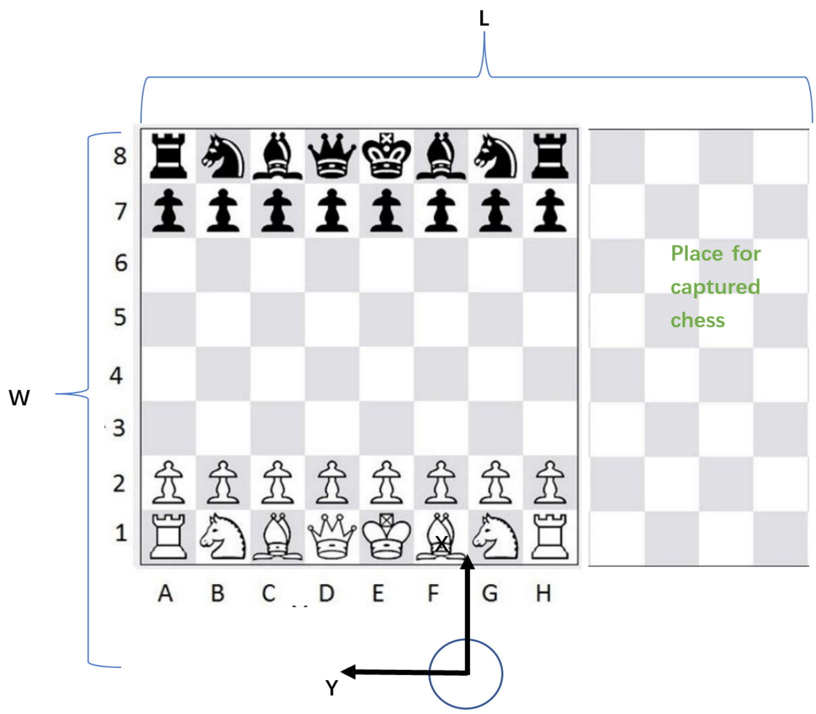

Figure 1: top view of working space

1 Generation of Trajectory

1.1 Position Trajectory

Since the position and velocity of both start and end points are provided, higher order polynomials are not required to generate the trajectory. To describe the motion of the robot arm in 3-Dimensional Space, a polynomial is needed for each axis. Each polynomial has 4 coefficients, shown in equation 1.

$$
\begin{aligned}
x(t) &= a_{x3}t^3 + a_{x2}t^2 + a_{x1}t + a_{x0} \\
y(t) &= a_{y3}t^3 + a_{y2}t^2 + a_{y1}t + a_{y0} \\
z(t) &= a_{z3}t^3 + a_{z2}t^2 + a_{z1}t + a_{z0}
\end{aligned}
\tag{1}
$$

These can be differentiated to produce the velocity polynomials shown in equation 2.

$$
\begin{aligned}
\dot{x}(t) &= 3a_{x3}t^2 + 2a_{x2}t + a_{x1} \\
\dot{y}(t) &= 3a_{y3}t^2 + 2a_{y2}t + a_{y1} \\
\dot{z}(t) &= 3a_{z3}t^2 + 2a_{z2}t + a_{z1}
\end{aligned}
\tag{2}
$$

Evaluating these polynomials for $t = 0$ provides the initial position and velocity in each axis, denoted by x_i, y_i, z_i, $\dot{x_i}$, $\dot{y_i}$, and $\dot{z_i}$. Doing the same at $t = t_f$ gives the final position and velocity, denoted by x_f, y_f, z_f, $\dot{x_f}$, $\dot{y_f}$, and $\dot{z_f}$. With these values known, we can generate the matrices shown in equation 3.

$$
\begin{gathered}
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix}
\begin{bmatrix} a_{x0} \\ a_{x1} \\ a_{x2} \\ a_{x3} \end{bmatrix}
=
\begin{bmatrix} x_i \\ x_f \\ \dot{x_i} \\ \dot{x_f} \end{bmatrix} \\
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix}
\begin{bmatrix} a_{y0} \\ a_{y1} \\ a_{y2} \\ a_{y3} \end{bmatrix}
=
\begin{bmatrix} y_i \\ y_f \\ \dot{y_i} \\ \dot{y_f} \end{bmatrix} \\
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix}
\begin{bmatrix} a_{z0} \\ a_{z1} \\ a_{z2} \\ a_{z3} \end{bmatrix}
=
\begin{bmatrix} z_i \\ z_f \\ \dot{z_i} \\ \dot{z_f} \end{bmatrix}
\end{gathered}
\tag{3}
$$

By choosing a value for t_f and solving these matrices for a given start and end point, the polynomials describing the position of the end-effector over the entire trajectory can be found.

1.2 Orientation Trajectory

In this part, we use the Euler parameters method to represent the change in orientation. The initial and final position of the end-effector are given by (x_i, y_i, z_i) and (x_f, y_f, z_f). Using the inverse kinematics derived in assignment 1, we can calculate the corresponding joint angles (q_1, q_2, q_3, q_4) required to reach these positions. These allow for the derivation of the rotation matrices R_{i0} and R_{f0}, which go from the end-effector frame to the inertial frame for each position. Following steps introduced in lecture, the rotation angles and Euler axis are as follows:

Rotation matrix from frame f to frame i: $R_{fi}{=}R_{i0}^T{*}R_{f0} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$

Rotation angle: $\theta_f = 2\cos^{-1}(\frac{1}{2}\sqrt{1 + r_{11} + r_{22} + r_{33}})$

Euler axis: $\hat{k} = \begin{bmatrix} k_x \\ k_y \\ k_z \end{bmatrix} = \frac{1}{2sin(\theta_f)} \begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$

Using θ_f, derived above, we can generate a cubic polynomial to represent the angle θ changing with time:

$$\theta(t) = a_{\theta 0} + a_{\theta 1} * t + a_{\theta 2} * t^2 + a_{\theta 3} * t^3 \tag{4}$$

Then, the coefficients can be determined using the same method as the position trajectory, with the constraints $\theta_i = 0$, $\dot{\theta_i} = 0$, $\theta_f = \theta_f$, and $\dot{\theta_f} = 0$. By solving the matrix equation 5 for a given start and end orientation, the coefficients can be determined.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \begin{bmatrix} a_{\theta 0} \\ a_{\theta 1} \\ a_{\theta 2} \\ a_{\theta 3} \end{bmatrix} = \begin{bmatrix} \theta_i \\ \theta_f \\ \dot{\theta_i} \\ \dot{\theta_f} \end{bmatrix} \tag{5}$$

Once the coefficients are determined, the Euler Parameters for any time can be calculated according to the equations given in 6. These can then be used to find the rotation matrix for the end effector orientation at that time, as shown in equation 7

$$\begin{aligned} e_1 = k_x \sin\frac{\theta(t)}{2} \quad e_2 = k_y \sin\frac{\theta(t)}{2} \\ e_3 = k_z \sin\frac{\theta(t)}{2} \quad e_4 = \cos\frac{\theta(t)}{2} \end{aligned} \tag{6}$$

$$ {}^i_tR(t) = \begin{bmatrix} 1-2e_2^2-2e_3^2 & 2(e_1e_2-e_3e_4) & 2(e_1e_3+e_2e_4) \\ 2(e_1e_2+e_3e_4) & 1-2e_1^2-2e_3^2 & 2(e_2e_3-e_1e_4) \\ 2(e_1e_3-e_2e_4) & 2(e_2e_3+e_1e_4) & 1-2e_1^2-2e_2^2 \end{bmatrix} \tag{7}$$

2 Implementation of Via Points

We set 2 via points directly above the initial and final position to ensure the chess piece can go over other nearby pieces. Since the height of a chess piece is at most 6cm, the end-effector would grab the chess piece from the bottom and lift it 7cm up to avoid obstacles, then move to 7cm above the final position, and finally lower down to place the chess piece on the board.
This results in 3 distinct segments to the trajectory. To obtain the coefficients for these segments in the x-axis, the following constraints are set:

$$x_1 = x_i,\ x_2 = x_i,\ x_3 = x_f,\ x_4 = x_f,\ \dot{x_1} = \dot{x_2} = \dot{x_3} = \dot{x_4} = 0$$

Similar constraints can be applied for the y- and z-axes:

$$y_1 = y_i,\ y_2 = y_i,\ y_3 = y_f,\ y_4 = x_f,\ \dot{y_1} = \dot{y_2} = \dot{y_3} = \dot{y_4} = 0$$

$$z_1 = z_i,\ z_2 = 7,\ z_3 = 7,\ z_4 = x_f,\ \dot{z_1} = \dot{z_2} = \dot{z_3} = \dot{z_4} = 0$$

Furthermore, because the trajectory is separated into 3 parts of different distances, we set the completion times to be different as well. The picking up and putting down part last 1 second each and the horizontal movement lasts 2 seconds, for a total duration of 4 seconds. In other words, the completion times for each segment are given by $t_{f1} = 1$, $t_{f2} = 2$ and $t_{f3} = 1$.

Then, using the algorithm for calculating coefficients of polynomial established above, we can get three discrete cubic curves for each of the three axes.

3 Trajectory Plot Generation

To verify these equations, we move a rook from position C4 (22,14,0) to H4 (22,-6,0) in our coordinates, moving four squares away from the initial position and over a knight positioned between the two endpoints. Noticeably, the inertial frame we defined is shown in figure 1 which is different from that given in assignment description, however the plots will show the plots of axes required in the assignment.

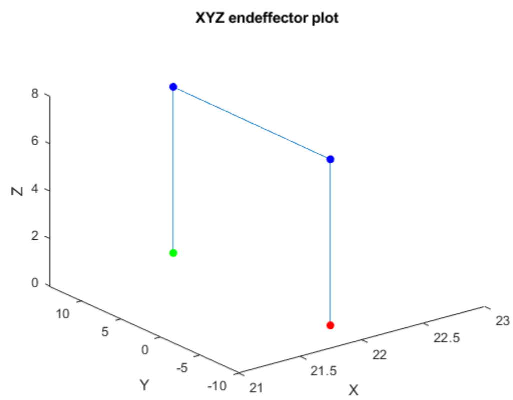

Figure 2: End-Effector Trajectory in 3 Dimensions

In figure 2, the blue points represents the location of via points used in the trajectory. The green dot represents the starting point (at $t = 0$) of the end-effector and the red dot represents the end point (at $t = t_f$).

As discussed above, we use two via points when moving the chess piece from one place to another, here set to (22,-14,7) and (22,6,7). As designed, the gripper lifts the chess piece 7 cm above and then move to a point at 7 cm above the final position and then lower down to place the chess piece.

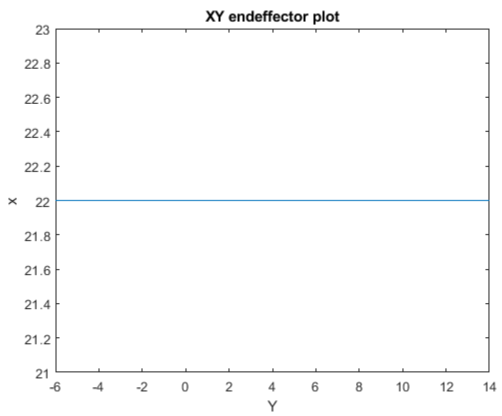

Figure 3: End-Effector Trajectory in X-Y Plane

Figure 3 shows the top view of the end-effector's motion in the X-Y plane. Although the internal code shows this as a change in y-coordinate from $y = 1$ to $y = -6$, this is movement along the x-axis defined in the assignment guidelines.

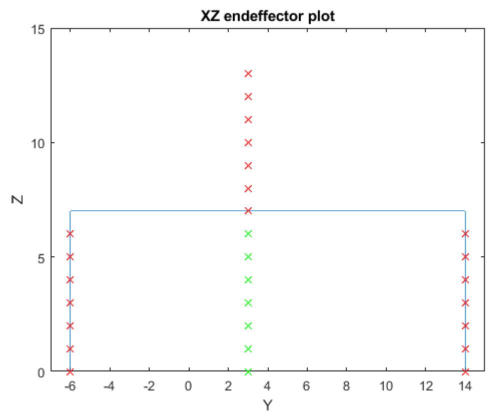

Figure 4: End-Effector Trajectory in X-Z Plane

The X-Z plane as defined in the assignment guidelines is shown in figure 4. The red crosses represent the position of the rook and the green crosses represent the knight, while the plotted line represents the displacement of the end-effector. It is observed that at location $y = 3$, the rook is sitting in the end-effector (where the gripper grips the chess piece's base) and is able to move over the knight.

As we set velocities at starting and end point of each cubic line is zero, the resulting trajectories are straight lines.

4 Feasibility demonstration

To demonstrate that the robot is capable of performing this trajectory, we plotted the positions the limbs would have to take in order to achieve the required end-effector pose. This was done through use of the inverse kinematics derived in assignment 2 to determine the joint-space representation of the limbs, which could then be used to find the position of each of the joints in task space. We plotted a sample of 11 configurations from across the example trajectory, resulting in the graph given in figure 5.

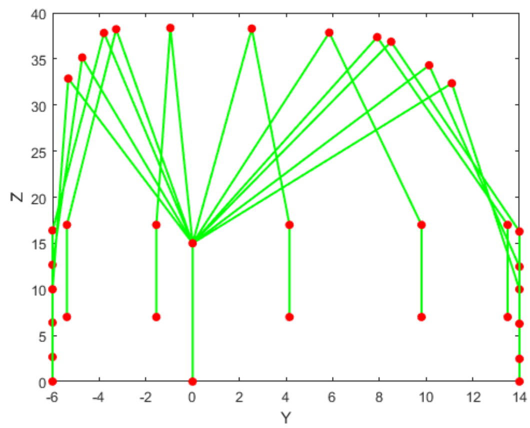

Figure 5: Demonstration of Robot Configurations

Conclusion

This report has shown the process of generating movement trajectories for our robotic arm. This includes an explanation of the algorithm, the way in which via points are taken into account, and a demonstration of the feasibility of the generated trajectory. The next step is to physically create the robot, and implement this trajectory algorithm into a controller that can successfully operate it.