

MECH 467 — Prelab 3

# Simulation of Contouring Performance

Ryan Edric Nashota (ID: 33508219)

## Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Part A – Trajectory Generation</b>	<b>1</b>
2.1	Trapezoidal Velocity Profile Equations . . . . .	1
2.2	Toolpath . . . . .	2
2.3	Motion Profiles . . . . .	2
2.4	Axis Commands . . . . .	3
<b>3</b>	<b>Part B – Two-Axis Controller Design</b>	<b>4</b>
3.1	Controller Design Methodology . . . . .	4
3.2	Bode Plots . . . . .	5
3.3	Closed-Loop Analysis . . . . .	6
<b>4</b>	<b>Part C – Contouring Performance Simulation</b>	<b>7</b>
4.1	Simulation Results (Handout Trajectory) . . . . .	7
4.2	Independent Creative Section: Custom Trajectory . . . . .	9
4.3	Simulation of Custom Trajectory . . . . .	11

## Table of Figures

1	Generated reference toolpath. . . . .	2
2	Displacement, Feedrate, and Tangential Acceleration profiles. . . . .	3
3	Axis Position, Velocity, and Acceleration commands. . . . .	4
4	Bode plots for Case 1 (LBW): X Axis (left) and Y Axis (right). . . . .	5
5	Bode plots for Case 2 (HBW): X Axis (left) and Y Axis (right). . . . .	6
6	Bode plots for Case 3 (Mismatch): X Axis (left) and Y Axis (right). . . . .	6
7	Simulated contouring performance (LBW Controller). . . . .	8
8	Tracking errors in X and Y axes during the trajectory. . . . .	9
9	Custom "M" shape trajectory. . . . .	10
10	Motion profiles for the custom trajectory. . . . .	11
11	Simulated contouring performance for the custom "M" trajectory. . . . .	12
12	Tracking errors for the custom trajectory. . . . .	13

## List of Tables

1	Closed-Loop System Analysis	7
---	-----------------------------	---

# 1 Introduction

This prelab focuses on the design and simulation of a two-axis motion control system. The objectives include generating trajectories for a specified toolpath, designing Lead-Lag controllers to meet specific bandwidth and phase margin requirements, and simulating the contouring performance of the system. Additionally, a custom trajectory is designed for experimental implementation.

## 2 Part A – Trajectory Generation

### Part A Deliverables

- Generated axis commands for the specified toolpath (two lines and a full circle).
- Plotted the toolpath, displacement/feedrate/acceleration profiles, and axis command profiles.

The reference toolpath consists of two linear segments and one full circular segment. The trajectory was generated using a trapezoidal velocity profile with  $F = 200 \text{ mm/s}$  and  $A = D = 1000 \text{ mm/s}^2$ . The interpolation period was  $T_i = 0.1 \text{ ms}$ .

### 2.1 Trapezoidal Velocity Profile Equations

The trajectory generation is based on the trapezoidal velocity profile. For a given total displacement  $S_{total}$ , desired feedrate  $F$ , and acceleration/deceleration  $A, D$ , the acceleration time  $T_a$ , deceleration time  $T_d$ , and constant velocity time  $T_c$  are calculated as follows:

$$T_a = \frac{F}{A}, \quad T_d = \frac{F}{D} \quad (1)$$

The distances covered during acceleration ( $S_a$ ) and deceleration ( $S_d$ ) are:

$$S_a = \frac{1}{2}FT_a, \quad S_d = \frac{1}{2}FT_d \quad (2)$$

If  $S_a + S_d > S_{total}$ , the profile becomes triangular, and the peak velocity is reduced. Otherwise, the constant velocity distance is  $S_c = S_{total} - S_a - S_d$ , and the time at constant velocity is  $T_c = S_c/F$ .

The displacement  $s(t)$ , velocity  $\dot{s}(t)$ , and acceleration  $\ddot{s}(t)$  profiles are then constructed by integrating the acceleration profile:

$$\ddot{s}(t) = \begin{cases} A & 0 \leq t < T_a \\ 0 & T_a \leq t < T_a + T_c \\ -D & T_a + T_c \leq t \leq T_a + T_c + T_d \end{cases} \quad (3)$$

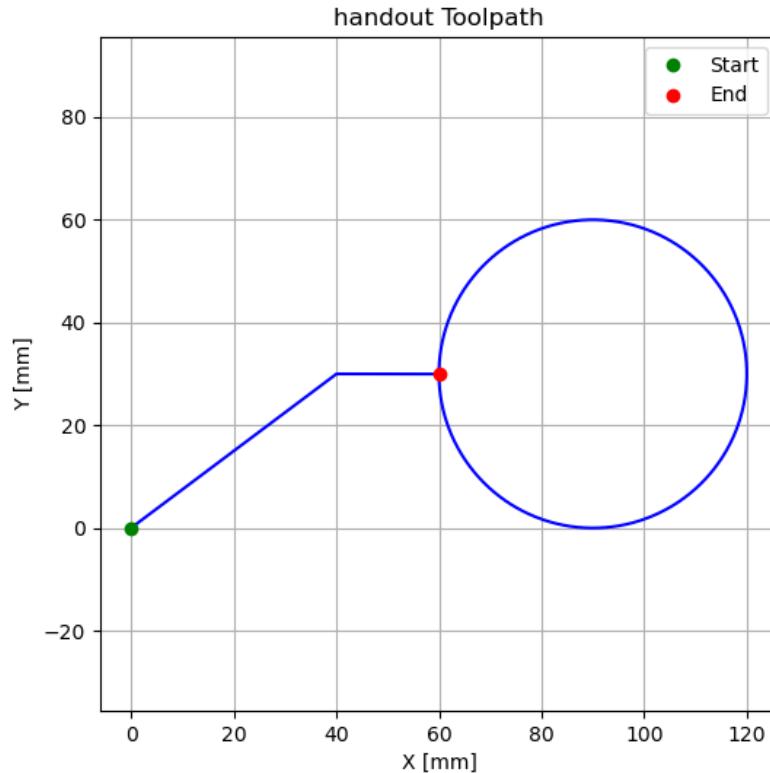
For circular interpolation, the tangential motion follows the same profile along the arc length  $S = R\theta$ . The axis commands are derived as:

$$x(t) = x_c + R \cos(\theta(t)), \quad y(t) = y_c + R \sin(\theta(t)) \quad (4)$$

where  $\theta(t) = \theta_{start} \pm s(t)/R$ .

## 2.2 Toolpath

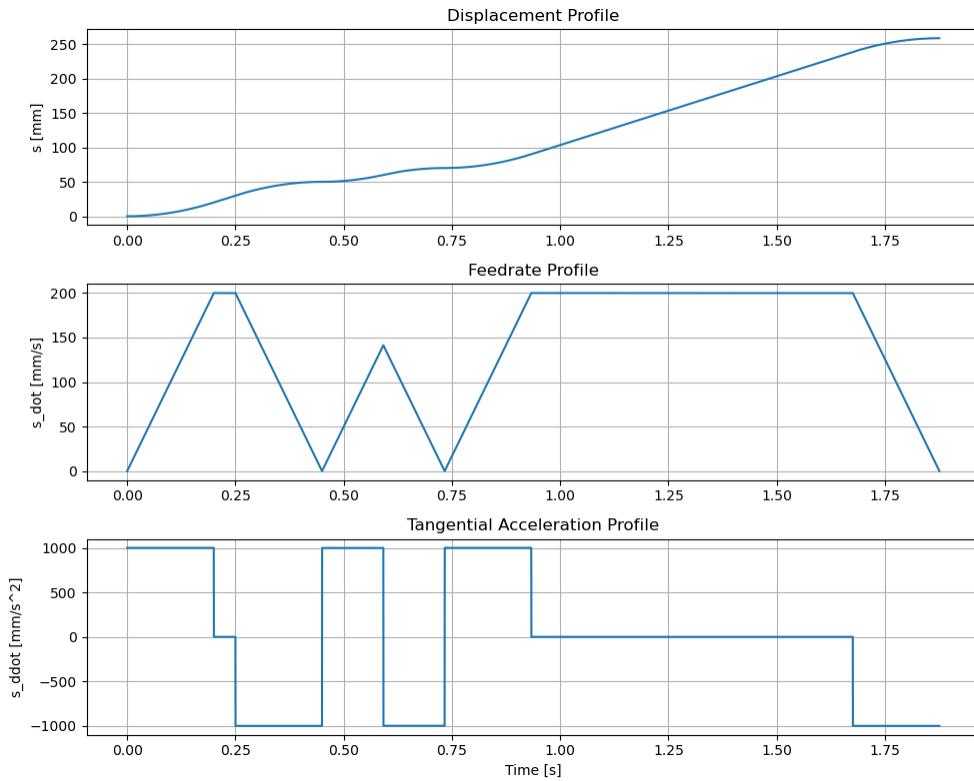
Figure 1 shows the generated toolpath in the XY plane. The path starts at (0, 0), moves linearly to (40, 30), then to (60, 30), and finally executes a full circle centered at (90, 30).



**Figure 1.** Generated reference toolpath.

## 2.3 Motion Profiles

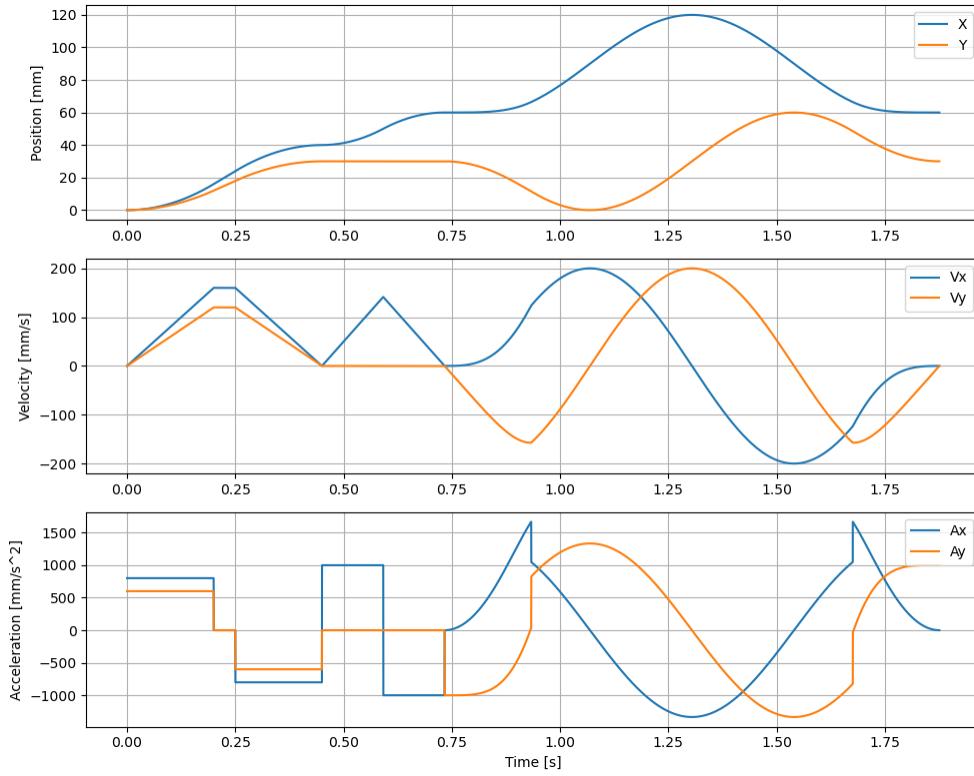
Figure 2 displays the displacement ( $s$ ), feedrate ( $\dot{s}$ ), and tangential acceleration ( $\ddot{s}$ ) profiles along the path. The trapezoidal nature of the velocity profile is clearly visible, with constant acceleration and deceleration phases.



**Figure 2.** Displacement, Feedrate, and Tangential Acceleration profiles.

## 2.4 Axis Commands

Figure 3 shows the position, velocity, and acceleration commands for the X and Y axes. The circular segment introduces sinusoidal variations in the axis velocities and accelerations.



**Figure 3.** Axis Position, Velocity, and Acceleration commands.

## 3 Part B – Two-Axis Controller Design

### Part B Deliverables

- Designed Lead-Lag controllers with integral action for three cases (LBW, HBW, Mismatched).
- Generated Bode plots for open-loop systems.
- Analyzed closed-loop poles, zeros, rise time, and overshoot.

### 3.1 Controller Design Methodology

The plant transfer function for each axis is given by:

$$G(s) = \frac{K_a K_t K_e}{s(J_e s + B_e)} \quad (5)$$

A Lead-Lag compensator is designed to boost the phase margin at the desired crossover frequency  $\omega_c$ . The compensator form is:

$$C_{LL}(s) = K \frac{\alpha \tau s + 1}{\tau s + 1} \quad (6)$$

The required phase contribution  $\phi_{max}$  is calculated as:

$$\phi_{max} = -180^\circ + PM_{target} - \angle G(j\omega_c) \quad (7)$$

The lead parameters are then found using:

$$\alpha = \frac{1 + \sin(\phi_{max})}{1 - \sin(\phi_{max})}, \quad \tau = \frac{1}{\omega_c \sqrt{\alpha}} \quad (8)$$

The gain  $K$  is chosen to set the open-loop magnitude to unity at  $\omega_c$ :

$$K = \frac{1}{|G(j\omega_c)|\sqrt{\alpha}} \quad (9)$$

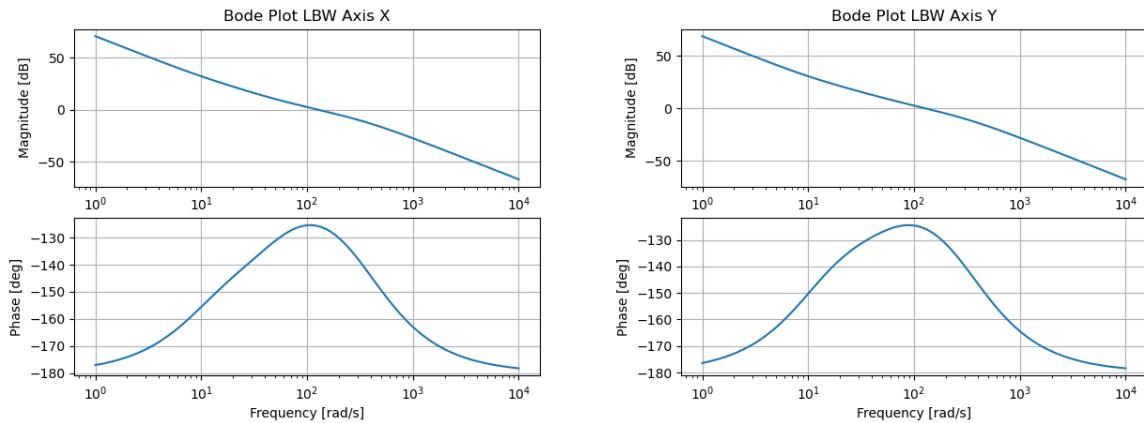
To eliminate steady-state error, an integral action is added:

$$C_I(s) = \frac{s + K_i}{s}, \quad K_i = \frac{\omega_c}{10} \quad (10)$$

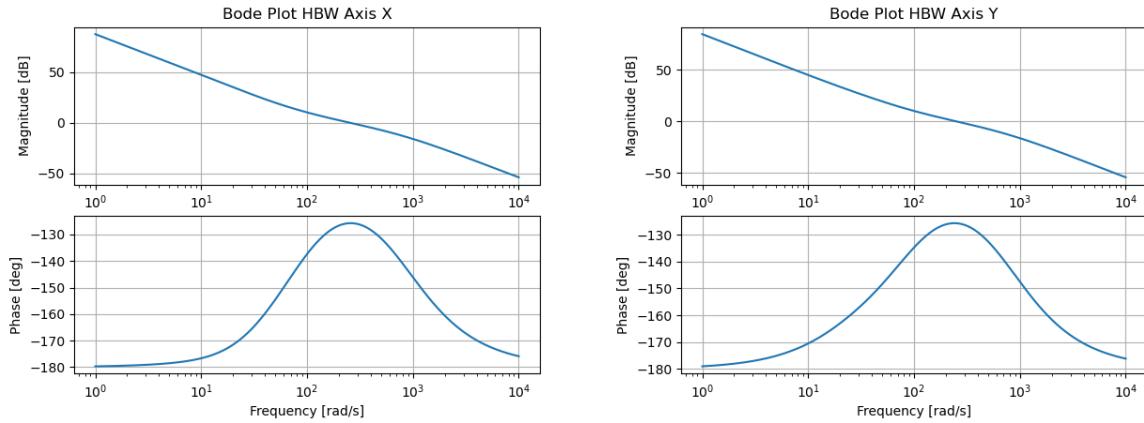
The final controller  $C(s) = C_{LL}(s)C_I(s)$  is discretized using Tustin's approximation for implementation.

## 3.2 Bode Plots

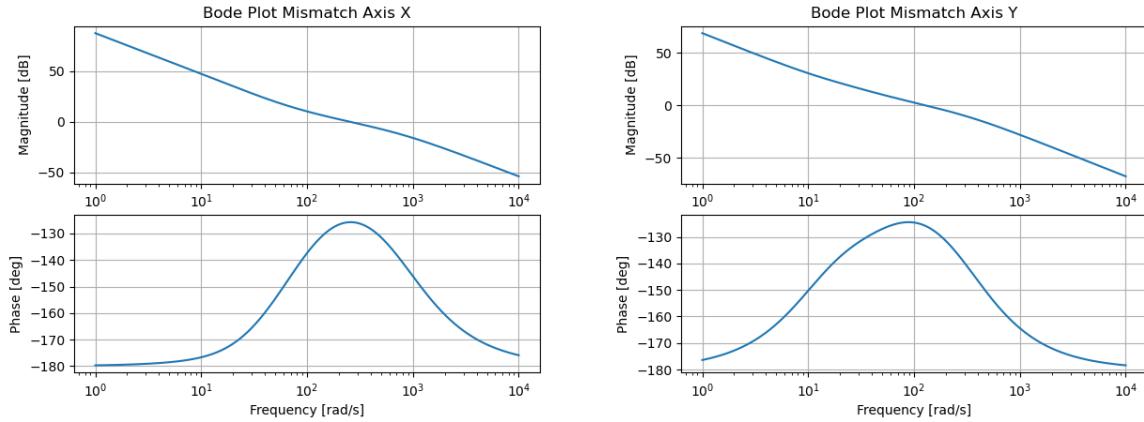
Figures 4 to 6 show the open-loop Bode plots for the designed controllers. The plots confirm that the target crossover frequencies and phase margins are achieved.



**Figure 4.** Bode plots for Case 1 (LBW): X Axis (left) and Y Axis (right).



**Figure 5.** Bode plots for Case 2 (HBW): X Axis (left) and Y Axis (right).



**Figure 6.** Bode plots for Case 3 (Mismatch): X Axis (left) and Y Axis (right).

### 3.3 Closed-Loop Analysis

Table 1 summarizes the closed-loop performance metrics and pole/zero locations for the designed controllers. The higher bandwidth (HBW) cases show faster rise times but slightly higher overshoot. The mismatched case combines the fast response of the X axis with the slower response of the Y axis.

**Table 1.** Closed-Loop System Analysis

Case	Axis	Rise Time [ms]	Overshoot [%]	Dominant Poles	Zeros
LBW	X	9.00	22.37	$0.986 \pm 0.008j$	-0.999, 0.999, 0.995
	Y	9.30	21.17	$0.987 \pm 0.010j$	-0.999, 0.999, 0.995
HBW	X	4.40	24.61	0.998, 0.982	-0.999, 0.997, 0.992
	Y	4.50	23.73	0.997, 0.984	-0.999, 0.997, 0.992
Mismatch	X	4.40	24.61	0.998, 0.982	-0.999, 0.997, 0.992
	Y	9.30	21.17	$0.987 \pm 0.010j$	-0.999, 0.999, 0.995

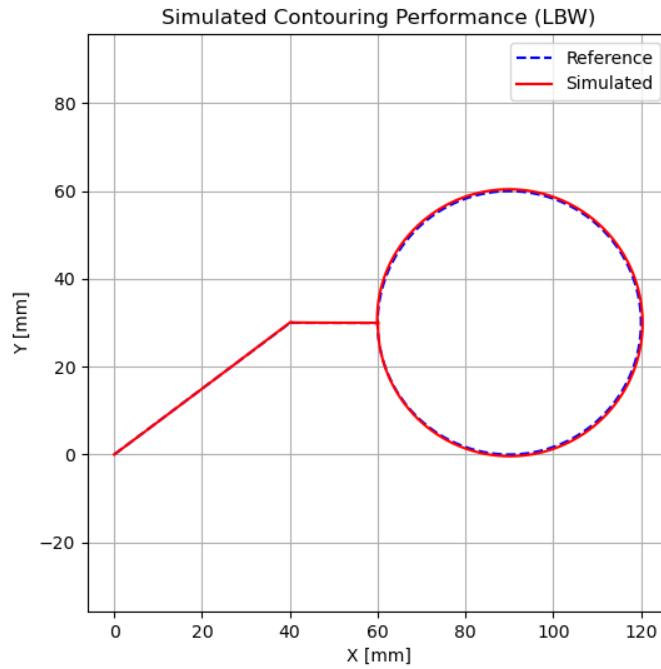
## 4 Part C – Contouring Performance Simulation

### Part C Deliverables

- Simulated the contouring performance using the discrete controllers and continuous plant model.
- Designed a custom toolpath for experimental implementation.

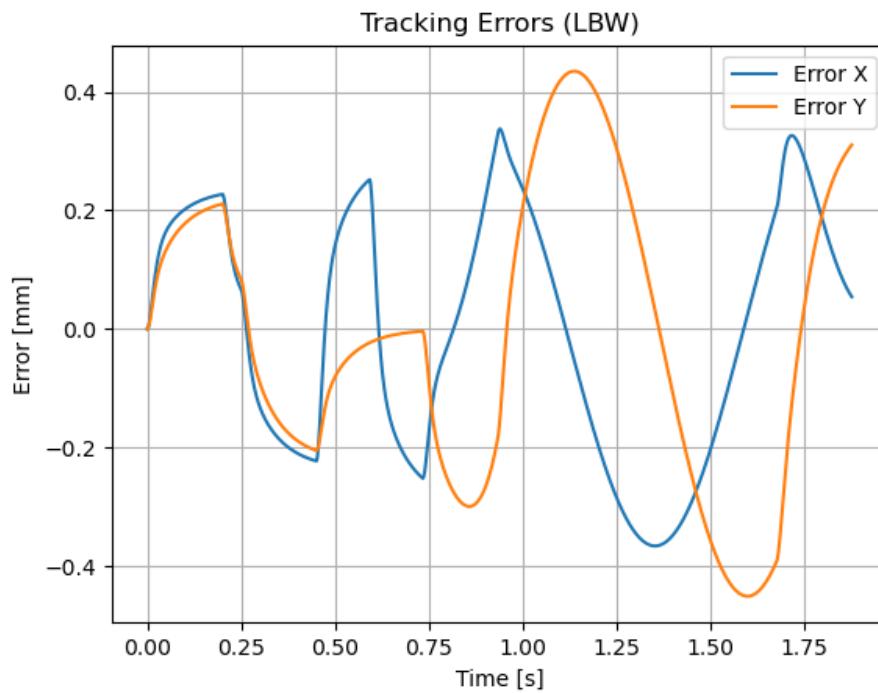
### 4.1 Simulation Results (Handout Trajectory)

The contouring performance was simulated using the Case 1 (LBW) controllers. Figure 7 shows the simulated path overlaid on the reference path. The system tracks the reference path closely, with deviations primarily occurring at corners and during the circular segment where the axis dynamics introduce lag.



**Figure 7.** Simulated contouring performance (LBW Controller).

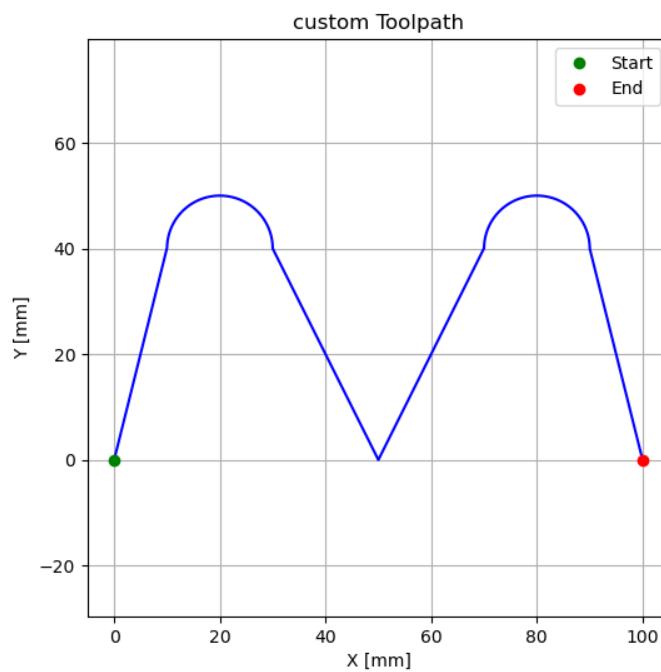
Figure 8 shows the tracking errors for both axes. The errors are largest during acceleration and deceleration phases, as expected for a Type 1 system tracking a ramp input (velocity).



**Figure 8.** Tracking errors in X and Y axes during the trajectory.

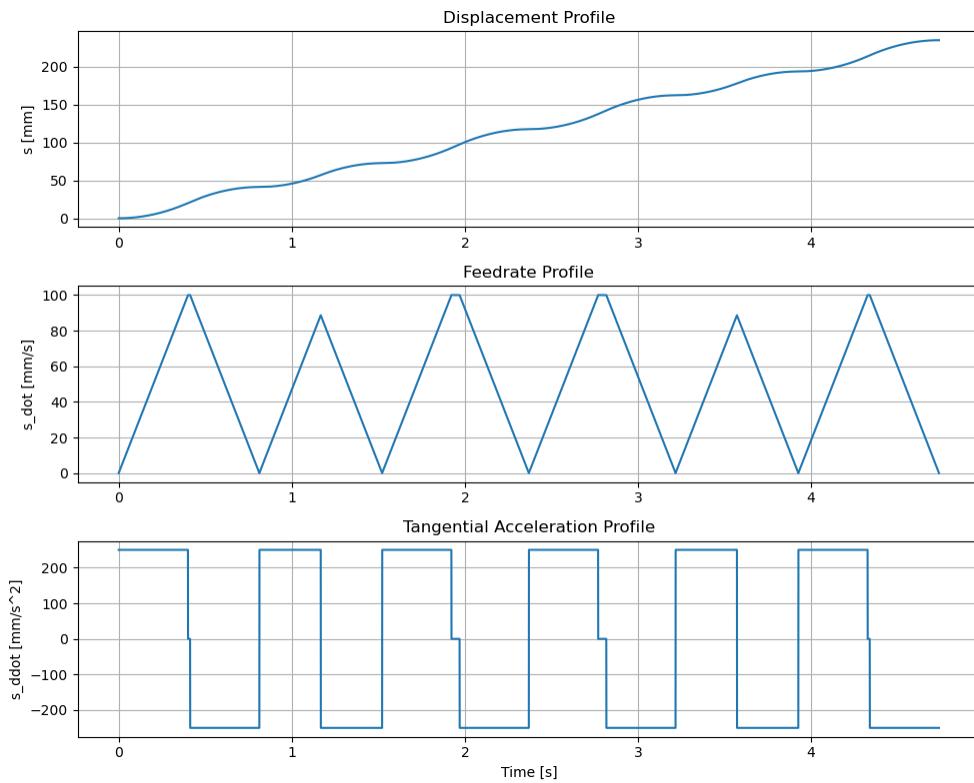
## 4.2 Independent Creative Section: Custom Trajectory

A custom trajectory was designed for the experimental session. The shape resembles the letter "M" with rounded tops, constructed using linear and circular segments.



**Figure 9.** Custom "M" shape trajectory.

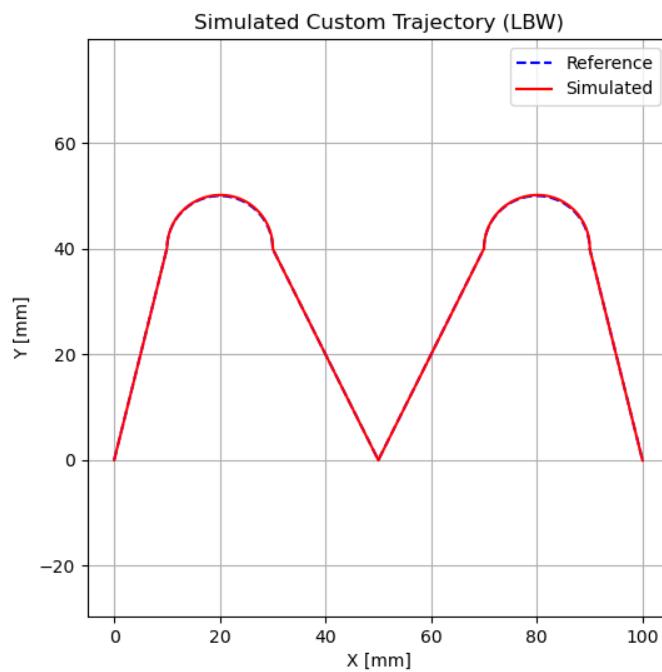
Figure 10 shows the motion profiles for the custom trajectory.



**Figure 10.** Motion profiles for the custom trajectory.

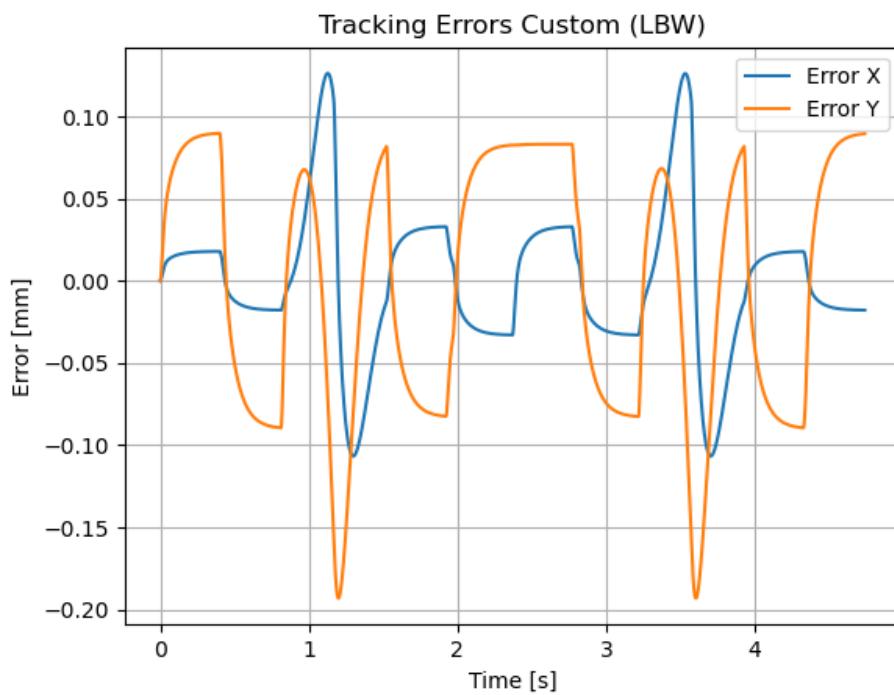
### 4.3 Simulation of Custom Trajectory

The custom trajectory was also simulated using the LBW controllers to verify feasibility and performance. Figure 11 shows the simulated path tracking the reference "M" shape.



**Figure 11.** Simulated contouring performance for the custom "M" trajectory.

Figure 12 displays the tracking errors. Similar to the handout trajectory, errors peak at the sharp corners where the velocity changes instantaneously in the ideal path (though smoothed by the trapezoidal profile in time, the geometric corner still presents a challenge for the tracking system).



**Figure 12.** Tracking errors for the custom trajectory.