PROBLEM 1

1a

For randomly selected system parameters the provided matlab code generates these results:

$$a_1 = 3 \quad \text{(Given system coefficient)}$$
$$a_2 = 2 \quad \text{(Given system coefficient)}$$
$$b = 5 \quad \text{(Plant gain)}$$

PD controller gains:

$$K_p = 10 \quad \text{(Proportional gain)}$$
$$K_d = 2 \quad \text{(Derivative gain)}$$

Closed-loop Transfer Function

$$T_{CL}(s) = \frac{10s + 50}{s^2 + 13s + 52}$$

System Properties

- **Natural Frequency** (ω_0): 7.211 rad/s
- **Damping Ratio** (ζ): 0.901

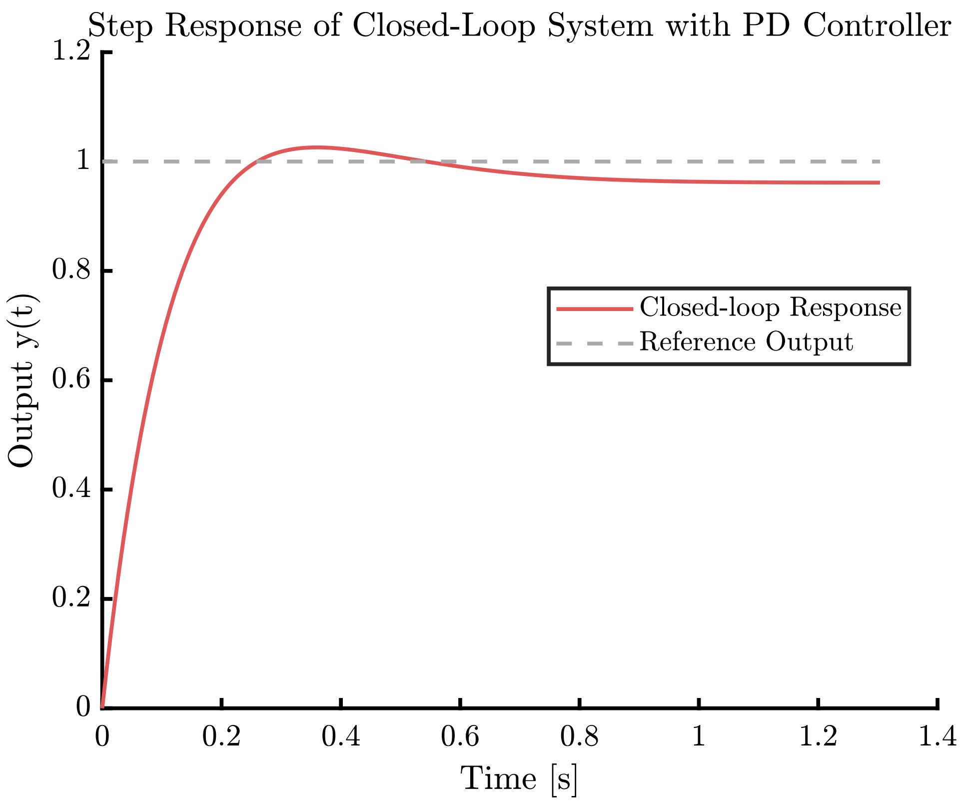

Figure 1: problem 1a simulation

1b

Computed PD Gains and System Performance

The computed PD controller gains are:

$$K_p = 106.333 \quad \text{(Proportional Gain)}$$
$$K_d = 4.511 \quad \text{(Derivative Gain)}$$

System Performance Metrics

- **Settling Time**: 0.548 s
- **Overshoot**: 34.787%
- **Rise Time**: 0.057 s

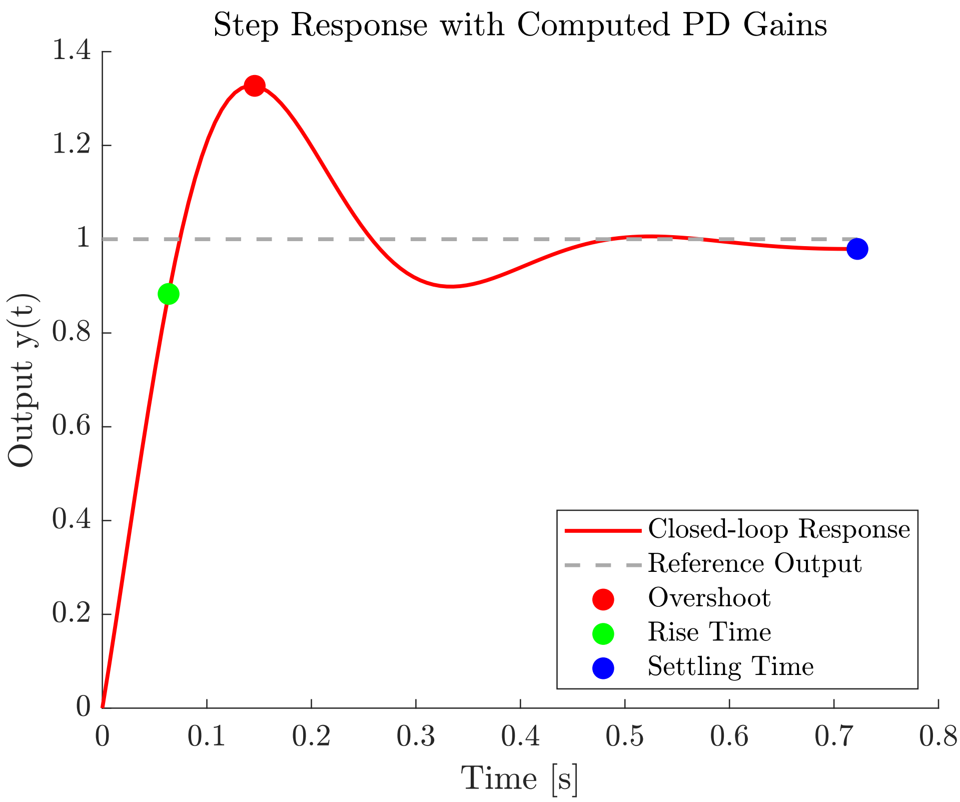

Figure 2: problem 1b simulation

As we can see the desired condition for overshoot is not satisfied, It was supposed to be less than 25%

1c

Tuned PD Gains and System Performance

The tuned PD controller gains are:

$$K_p = 109.333 \quad \text{(Proportional Gain)}$$
$$K_d = 9.000 \quad \text{(Derivative Gain)}$$

System Performance Metrics

- **Settling Time**: 0.274 s
- **Overshoot**: 17.976%
- **Rise Time**: 0.046 s

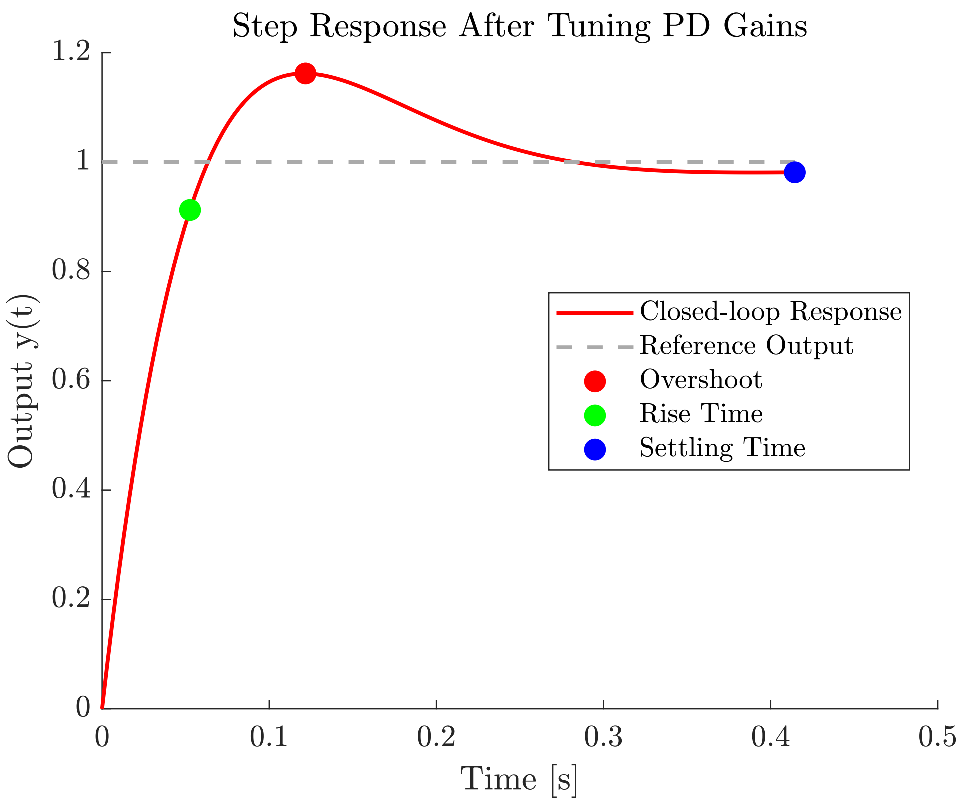

Figure 3: problem 1c simulation

After tuning the PD controller to the different values, all System Performance Metrics satisfy the desired conditions.

PROBLEM 2

2a

PID Controller Gains

Parameter	**H_2 (Water Level)**	**T_2 (Temperature)**
K_p (Proportional Gain)	2.5982	13.7632
K_i (Integral Gain)	0.0332	0.2754
K_d (Derivative Gain)	29.0047	153.4397
Crossover Frequency (rad/s)	0.0134	0.0309
Phase Margin (°)	74.08	69.53
Stability	Yes	Yes

Table 1: PID Controller Gains

Time-Domain Performance

Metric	**H_2 (Water Level)**	**T_2 (Temperature)**
Rise Time (s)	122.7720	51.2394
Transient Time (s)	421.2066	161.4242
Settling Time (s)	421.2066	161.4242
Overshoot (%)	6.5687	6.2582
Peak Time (s)	260.1271	105.8050

Table 2: Time-Domain Performance

Steady-State Error Constants

System	Step Response (K_s)	Ramp Response (K_v)	Parabolic Response (K_a)
H_2 (Water Level)	1.000	0.000	0.000
T_2 (Temperature)	1.000	0.000	0.000

Table 3: Steady-State Error Constants

Stability Margins

Metric	H_2 (Water Level)	T_2 (Temperature)
Phase Margin (°)	146.02	144.42
Delay Margin (s)	285.74	118.99
Stability	Yes	Yes

Table 4: Stability Margins

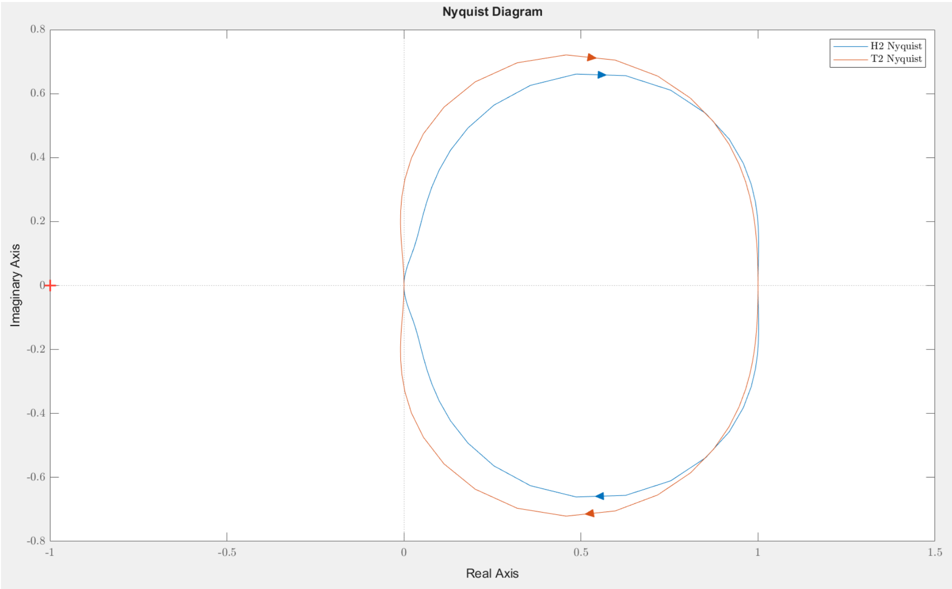

Figure 4: Nyquist plot

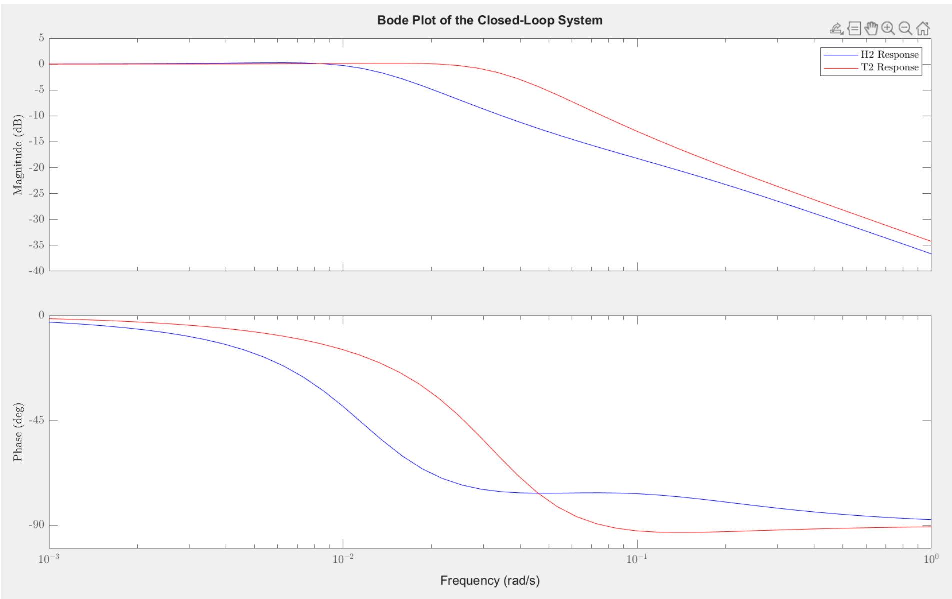

Figure 5: Bode plots for Water Tank System

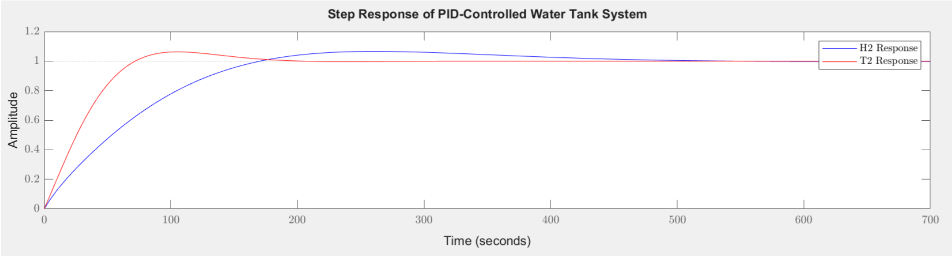

Figure 6: Step response of PID controller Water Tank System

2b

Simulink results

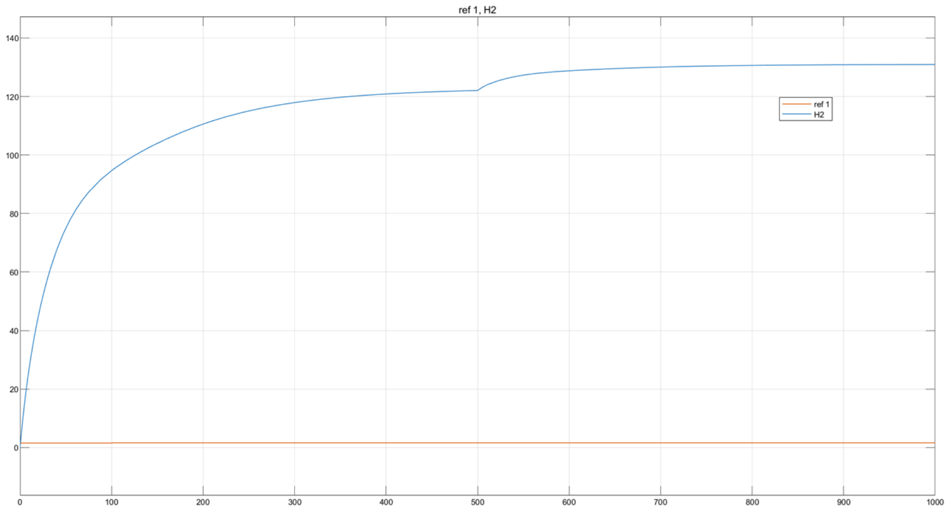

Figure 7: Simulation result

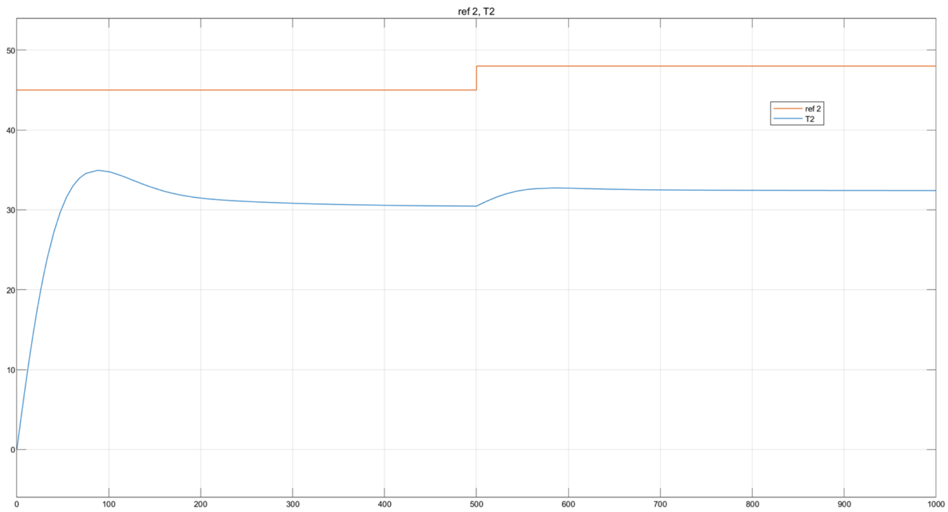

Figure 8: Simulation result

Python result

https://colab.research.google.com/drive/1PgQT3lyIQOKjHOaA7xTR-p6A5aHB1Uxd?usp=sharing

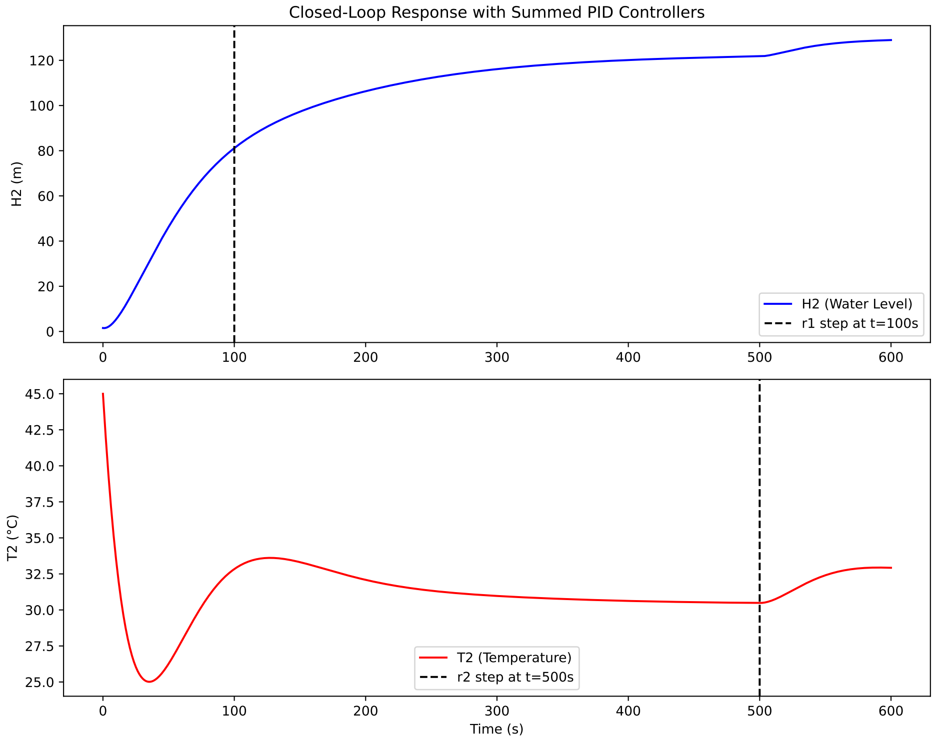

Figure 9: Simulation result

I think that the combined method demonstrates that both outputs (water level and temperature) can be controlled using a single input, but there are a few possible concerns.

1) Because the system is linearized around a single operating point, substantial fluctuations in level and temperature may occur outside of the linear model's accuracy range.

2)To use only one actuator for two outputs might result in powerful interactions: altering one can unintentionally impact the other.

3) When tuned in MATLAB, the comparatively big derivative gains might cause severe transients or noise sensitivity.