MAE 4421 Design Project

Hydraulic Control Spool Valve System Design Project

9 May 2025

Authors: Daniel Buckingham, Matthew Schnese, Tupou Tupou

Introduction:

The goal of this project was to characterize a hydraulic control spool valve similar to one which might be found in an automotive transmission and design a position control system for it. The control system was required to be quick and efficient as possible while limited by several constraints. These constraints included the displacement of the spool valve limited to less than $0.05\ m$, the acceleration of the linkage limited to less than $1.5\ m/s^2$, and the actuator force limited to $12\ N$. These system requirements can be met through powered electronics using the output signal and a compensator to augment the system response to produce a more desirable output as shown in the block diagram in Figure 1.

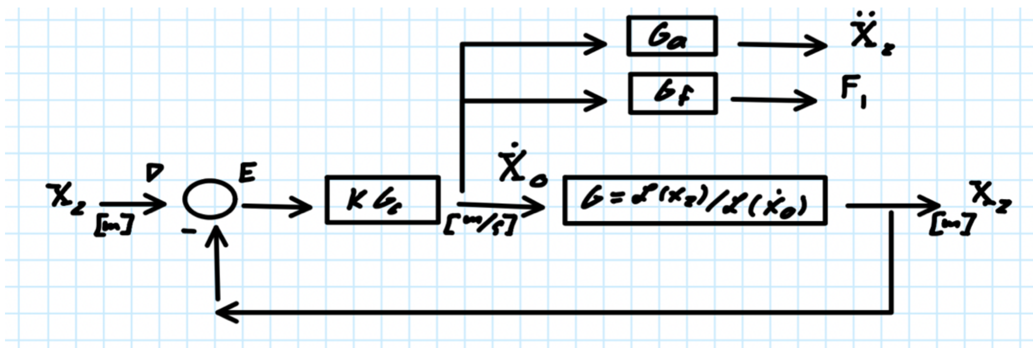

Figure 1: Block diagram of hydraulic spool valve position control system.

System Analysis:

Before the system could be analyzed a state space model was built using the simplified diagram of the system in Figure 2. With the input for the system being the flow velocity in the hydraulic actuator $\dot{x}_0$ the needed to fully characterize the system response were the displacement of mass two x_2, the acceleration of mass two $\ddot{x}_2$, and the force of the actuator (mass one) F_1.

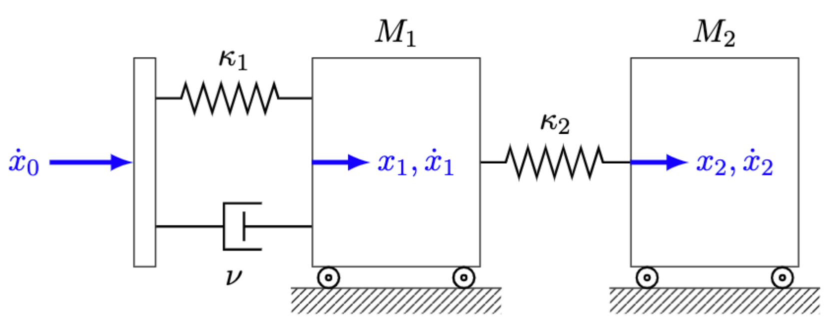

Figure 2: Simplified diagram of hydraulic control spool valve.

In order to build a state space model of this system the bond graph in Figure 3 was created. Through this bond graph the state space model shown in Figure 4 was developed which was in turn used to generate transfer functions for each of the desired outputs.

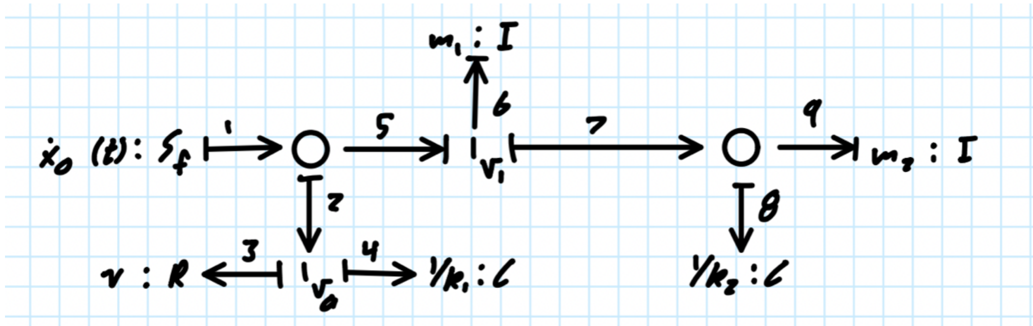

Figure 3: Bond graph created to assist in the creation of a state space model.

$$\dot{x} = \begin{bmatrix} \dot{\delta}_1 \\ \dot{p}_1 \\ \dot{\delta}_2 \\ \dot{p}_2 \end{bmatrix} = \begin{bmatrix} 0 & -1/m_1 & 0 & 0 \\ k_1 & -\nu/m_1 & -k_2 & 0 \\ 0 & 1/m_1 & 0 & -1/m_2 \\ 0 & 0 & k_2 & 0 \end{bmatrix} \begin{bmatrix} \delta_1 \\ p_1 \\ \delta_2 \\ p_2 \end{bmatrix} + \begin{bmatrix} 1 \\ \nu \\ 0 \\ 0 \end{bmatrix} \dot{x}_0(t)$$

$$y = \begin{bmatrix} x_2 \\ F_1 \\ \ddot{x}_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & k_2 & 0 \\ 0 & 0 & k_2/m_2 & 0 \end{bmatrix} \begin{bmatrix} \delta_1 \\ p_1 \\ \delta_2 \\ p_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \dot{x}_0(t)$$

Figure 4: State space model used for derivation of transfer functions for each output.

This state space model was created in MATLAB and used to create the transfer functions in Equations 1 through 3 as well as the step response of the unmodified system in Figure 5.

$$x_2 = \frac{s^3 + 11s^2 + 418s}{s^4 + 11s^3 + 418s^2 + 2000s + 25000} \qquad [1]$$

$$F_1 = \frac{8000s^2 + 100000s}{s^4 + 11s^3 + 418s^2 + 2000s + 25000} \qquad [3]$$

$$\ddot{x}_2 = \frac{2000s^2 + 25000s}{s^4 + 11s^3 + 418s^2 + 2000s + 25000} \qquad [2]$$

Test Plan:

With the transfer functions derived from the state space model for the position and acceleration of the spool valve and the force of the hydraulic actuator a locus plot and uncompensated step response was created for each. The locus plots of the force and acceleration transfer functions in Figure 5 revealed that the gain in the compensator would have to remain low.

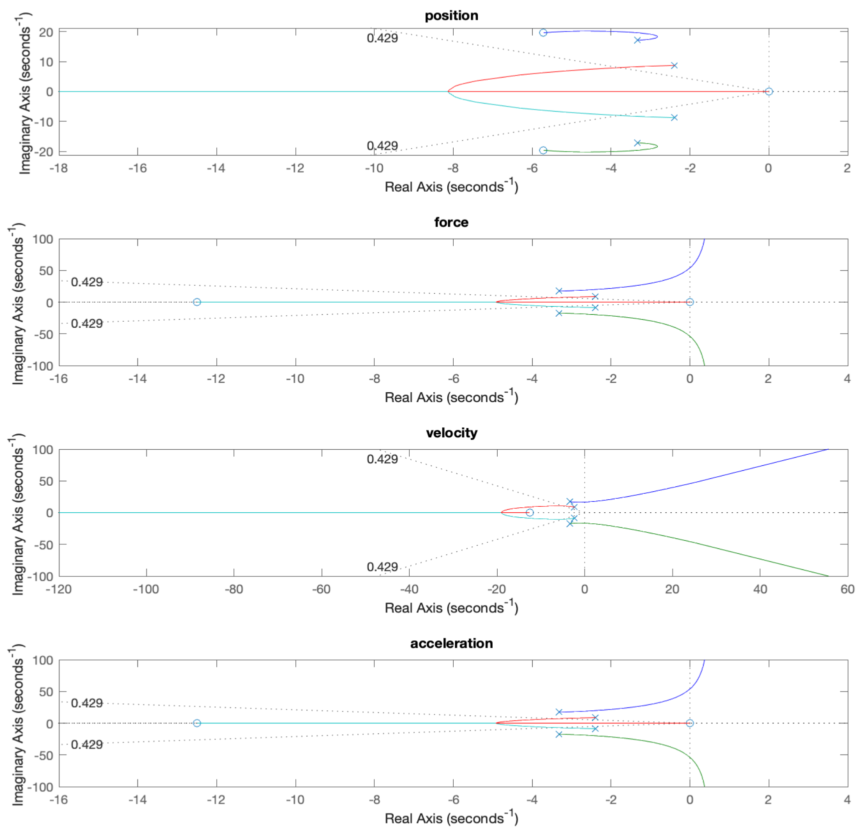

Figure 5: Locus plots.

On the other hand, the uncompensated step responses in Figure 6 revealed that a PID compensator would be the ideal method for adjusting this systems steady state response. A PID compensator would allow the system order to be increased and eliminate the steady state error.

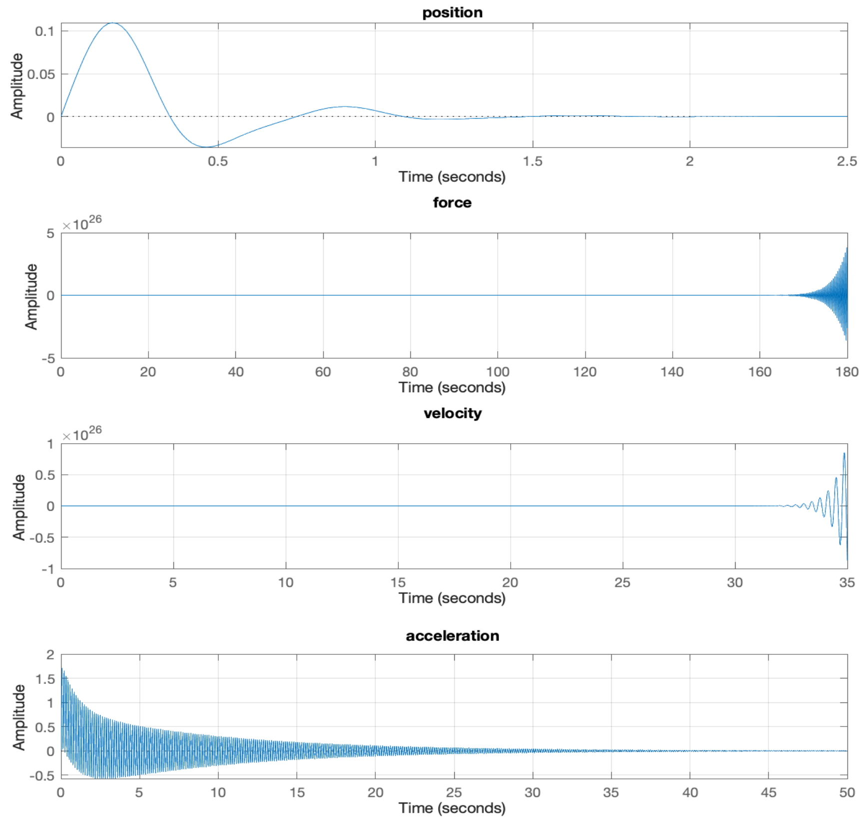

Figure 6: Uncompensated step responses.

For the controller, several tests were performed for a PID compensator setup. These tests were performed to analyze the stability of the system's three transfer functions when different parameters of the PID were changed. These tests were performed with a unit step input, and the time scale that was analyzed was from zero to two seconds. The system was analyzed in MATLAB's Simulink toolbox, which allowed for changes to the compensator to be made quickly. The system was tested for the four constraints placed on the actuator: Minimizing the leak time, position not exceeding 0.05 meters, acceleration not exceeding 1.5m/s^2, and the force requested by the system not exceeding 12 Newtons. Additionally, Simulink was used to find the amount of time spent in the region between x=0.005 and x=0.035 meters. The focus of the analysis was finding gain values that were stable and fulfilled the maximum position, acceleration, and force constraints then tweaking the values by increasing or decreasing each gain individually. This process allowed the time spent between 0.005 and 0.035 meters to be minimized while conserving the stability of the system. The PID system created using Simulink is below in Figure 7.

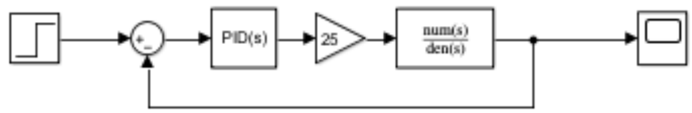

Figure 7: PID feedback system.

Results:

The results of the compensator tests were a PID compensator with gains K_p=0.0055, K_I=0.1, and K_d=0.00122 followed by a gain K of 25. This compensator setup followed all four constraints, with a maximum position of 0.0465 m, a max acceleration of 0.998 m/s^2, a maximum force of 0.9177 N, and the time spent between x=0.005 and x=0.035 meters being 0.03 seconds. The position of the linkage also settles to 0.4 meters in steady state. Position, acceleration, and force plotted versus time can be seen below in Figure 8.

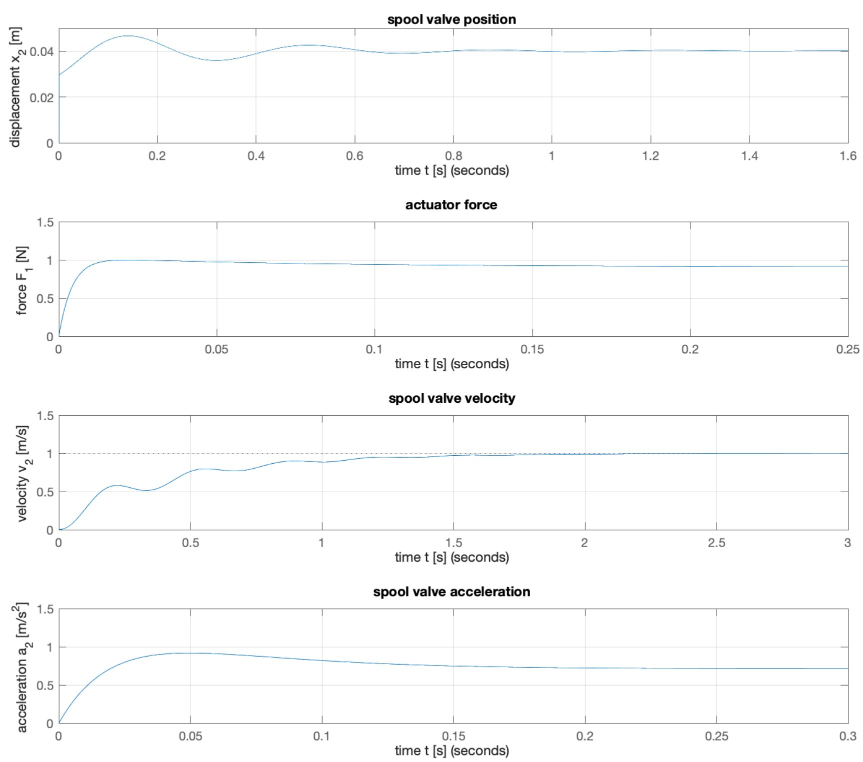

Figure 8: Compensated step responses.

Table 1: Final compensated system outputs

Compensated Outputs		
maximum displacement	maximum acceleration	maximum force
0.0465 m	0.998 m/s^2	0.918 N

Conclusion:

The completed control system, designed using a PID compensator derived from Root Locus analysis, successfully stabilized the hydraulic spool valve and met all design specifications. The final compensated system achieved a maximum spool displacement of 0.0465 m, a peak acceleration of 0.998 m/s^2, and a peak actuator force of only 0.918 N, which was within the limits of 0.05 m, 1.5 m/s^2, and 12 N, respectively. Additionally, the time spent within the leakage zone (0.005 – 0.035 meters) was reduced to just 0.03 seconds, minimizing backflow and overall system inefficiency.

These results demonstrate the effectiveness of the PID controller in regulating both transient and steady-state behavior. Through iterative gain tuning and performance evaluation in Simulink, the system achieved fast rise time, minimal overshoot, and precise steady-state tracking, confirming its suitability for application in a transmission style hydraulic spool valve.

Critique:

One of the primary challenges faced during this project was the lack of in-person collaboration. Without the full team working together in the same room, communication delays, misinterpretations, and inconsistent assumptions occasionally impacted the flow and clarity of the work. This separation made it harder to rapidly align on decisions, particularly during the controller tuning phase and transfer function validation. As a result, duplicate efforts were sometimes made, and key analysis elements, such as the interpretation of simulation outputs or system assumptions, took longer to converge. Despite this, the team adapted by actively using group chats, file sharing, and discussion threads to keep progress moving. On the technical side, the system analysis was thorough, but early results lacked alignment across team members regarding expected steady-state behavior and proper constraint tracking. With better synchronization, the PID tuning process and validation could have been more efficient. However, the project ultimately produced a stable, verified design that satisfied all constraints, demonstrating resilience despite the logistical hurdles.

Appendix A: MATLAB code

```
% Daniel Buckingham
% Matthew Schnese
% Tupou Tupou
% MAE 4421
% system design project

clear;clc;close all;

% physical constants

m1 = 7;
m2 = 4;
k1 = 1000;
k2 = 700;
v = 80;

A = [0 (-1/m1) 0 0 ; k1 (-v/m1) -k2 0 ; 0 (1/m1) 0 (-1/m2); 0 0 k2 0];
B = [1 ; v ; 0 ; 0];
C = [1 0 1 0 ; 0 0 k2 0 ; 0 0 0 (1/m2) ; 0 0 (k2/m2) 0];
D = [0 ; 0 ; 0 ; 0];

system = ss(A,B,C,D);
[num,den] = ss2tf(A,B,C,D);

p = tf(num(1,:),den);
f = tf(num(2,:),den);
v = tf(num(3,:),den);
a = tf(num(4,:),den);

z = 0.429;
w = 0;

figure(1);
subplot(4,1,1);
rlocus(p);
title('position');
sgrid(z,w);
subplot(4,1,2);
rlocus(f);
title('force');
sgrid(z,w);
subplot(4,1,3);
rlocus(v);
title('velocity');
sgrid(z,w);
subplot(4,1,4);
rlocus(a);
title('acceleration');
sgrid(z,w);

figure(1);
subplot(4,1,1);
```

```
step(feedback(p,1));
title('position');
grid on
subplot(4,1,2);
step(feedback(f,1));
title('force');
grid on
subplot(4,1,3);
step(feedback(v,1));
title('velocity');
grid on
subplot(4,1,4);
step(feedback(a,1));
title('acceleration');
grid on

t = 0.00122/100;
c = pid(0.0055,0.1,0.00122,t);
K = 25;
pc = c*K*p;
fc = c*K*f;
vc = c*K*v;
ac = c*K*a;

figure(3);
subplot(4,1,1);
step(feedback(pc,1));
ylim([0 0.05]);
ylabel('displacement x_{2} [m]');
xlabel('time t [s]');
title('spool valve position');
grid on
subplot(4,1,2);
step(feedback(fc,1));
ylim([0 1.5]);
ylabel('force F_{1} [N]');
xlabel('time t [s]');
title('actuator force');
grid on
subplot(4,1,3);
step(feedback(vc,1));
ylim([0 1.5]);
ylabel('velocity v_{2} [m/s]');
xlabel('time t [s]');
title('spool valve velocity');
grid on
subplot(4,1,4);
step(feedback(ac,1));
ylim([0 1.5]);
ylabel('acceleration a_{2} [m/s^{2}]');
xlabel('time t [s]');
title('spool valve acceleration');
grid on
```