

LAB4: Design and control of a high-speed rail pantograph

Objective:

- To learn how P, PI and PID controller is implemented.
- How to achieve different design specifications using PID controller.

Description of system: some high-speed rail systems are powered by electricity supplied to a pantograph on the train's roof from a catenary overhead, as shown in figure (1). The force applied by the pantograph to the catenary is regulated to avoid loss of contact due to excessive transient motion. A proposed method to regulate the force uses a closed-loop feedback system, whereby a force, F_{up} , is applied to the bottom of the pantograph, resulting in an output force applied to the catenary at the top. The contact between the head of the pantograph and the catenary is represented by a spring. The output force is proportional to the displacement of this spring, which is the difference between the catenary and pantograph head vertical positions.

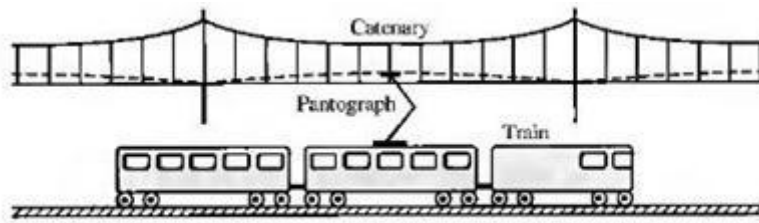


Figure: 1 High-speed rail system showing pantograph and catenary (O'Connor, 1997)

Pre lab: No pre-lab this time. Participation in lab discussion is counted towards pre-lab.

Background:

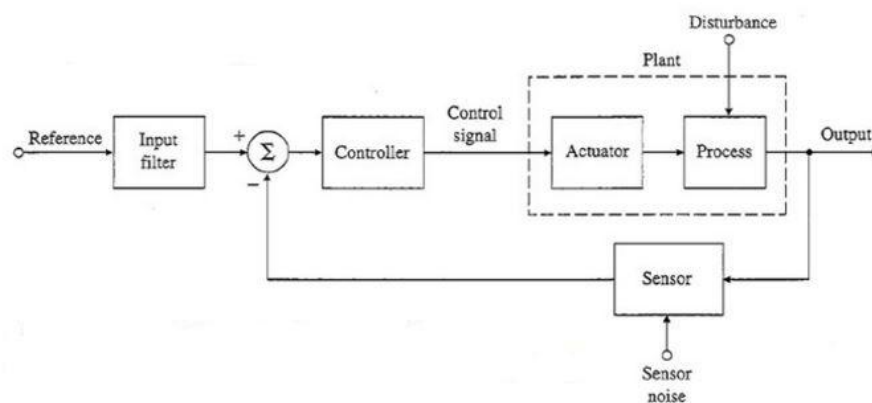


Figure: 2 Component block diagram of an elementary feedback control (Franklin, 2015)

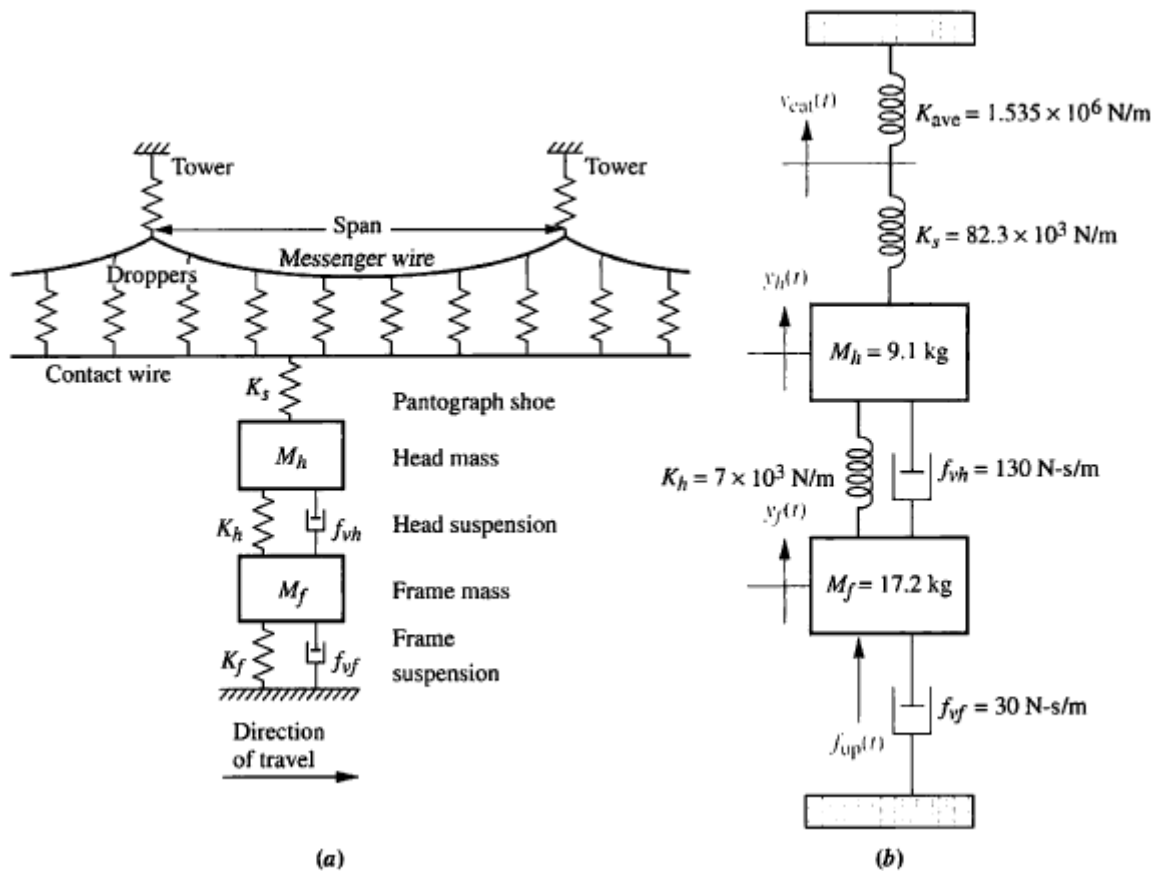


Figure: 3 (a) Coupling of pantograph and catenary; (b) simplified representation showing The active-control force (O'Connor, 1997)

Dominant poles: In the homogeneous (or transient) response of a given linear system, each pole (or complex pair of poles) contribute terms of the form $\exp(a^*t)$ where "a" is the value of the real part of the pole in question. If these poles are in the left-half plane, i.e. the system is stable, then the contribution to the response for a given value of "a" dies out faster if the pole is farther to the left (since $\exp(-a^*t)$ approaches zero faster for larger "a"). Therefore, the poles closest to the imaginary axis are the ones that tend to dominate the response since their contribution takes a longer time to die out. (Murray, Caltech).

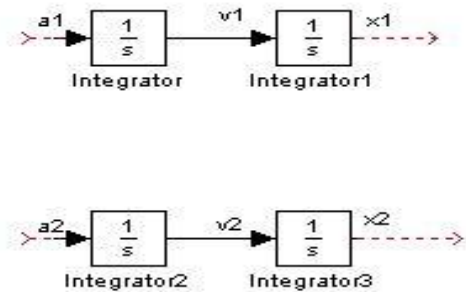


Figure: 4 starting blocks of Simulink block diagram

Stability Criteria and Routh-Hurwitz criteria:

(Adapted from https://en.wikibooks.org/wiki/Control_Systems/Routh-Hurwitz_Criterion)

The Routh-Hurwitz stability criterion provides a simple algorithm to decide whether or not the zeros of a polynomial are all in the left half of the complex plane (such a polynomial is called at times "Hurwitz"). A Hurwitz polynomial is a key requirement for a linear continuous-time invariant to be stable (all bounded inputs produce bounded outputs).

Necessary stability conditions: Conditions that must hold for a polynomial to be Hurwitz. If any of them fails - the polynomial is not stable. However, they may all hold without implying stability.

Sufficient stability conditions: Conditions that if met imply that the polynomial is stable. However, a polynomial may be stable without implying some or any of them. The Routh criterion provides conditions that are both necessary and sufficient for a polynomial to be Hurwitz.

The Routh-Hurwitz criterion is comprised of three separate tests that must be satisfied. If any single test fails, the system is not stable and further tests need not be performed. For this reason, the tests are arranged in order from the easiest to determine to the hardest.

The Routh Hurwitz test is performed on the denominator of the transfer function, the **characteristic equation**. For instance, in a closed-loop transfer function with $G(s)$ in the forward path, and $H(s)$ in the feedback loop, we have:

$$T(s) = \frac{G(s)}{1 + G(s)H(s)}$$

If we simplify this equation, we will have an equation with a numerator $N(s)$, and a denominator $D(s)$:

$$T(s) = \frac{N(s)}{D(s)}$$

The Routh-Hurwitz criteria will focus on the denominator polynomial $D(s)$.

Routh-Hurwitz Tests

Here are the three tests of the Routh-Hurwitz Criteria. For convenience, we will use N as the order of the polynomial (the value of the highest exponent of s in $D(s)$). The equation $D(s)$ can be represented generally as follows:

$$D(s) = a_0 + a_1s + a_2s^2 + \dots + a_Ns^N$$

Rule 1

All the coefficients a_i must be present (non-zero)

Rule 2

All the coefficients a_i must be positive (equivalently all of them must be negative, with no sign change)

Rule 3

If Rule 1 and Rule 2 are both satisfied, then form a Routh array from the coefficients a_i . There is one pole in the right-hand s -plane for every sign change of the members in the first column of the Routh array (any sign changes, therefore, mean the system is unstable).

The Routh array is formed by taking all the coefficients a_i of $D(s)$, and staggering them in array form. The final columns for each row should contain zeros:

$$\begin{array}{l} s^N \\ s^{N-1} \end{array} \left| \begin{array}{cccc} a_N & a_{N-2} & \dots & 0 \\ a_{N-1} & a_{N-3} & \dots & 0 \end{array} \right|$$

Therefore, if N is odd, the top row will be all the odd coefficients. If N is even, the top row will be all the even coefficients. We can fill in the remainder of the Routh Array as follows:

$$\begin{array}{l} s^N \\ s^{N-1} \\ \\ s^0 \end{array} \left| \begin{array}{cccc} a_N & a_{N-2} & \dots & 0 \\ a_{N-1} & a_{N-3} & \dots & 0 \\ b_{N-1} & b_{N-3} & \dots & \\ c_{N-1} & c_{N-3} & \dots & \\ \dots & & & \end{array} \right|$$

Now, we can define all our b , c , and other coefficients, until we reach row s^0 . To fill them in, we use the following formulae:

$$b_{N-1} = \frac{-1}{a_{N-1}} \left| \begin{array}{cc} a_N & a_{N-2} \\ a_{N-1} & a_{N-3} \end{array} \right|$$

And

$$b_{N-3} = \frac{-1}{a_{N-1}} \begin{vmatrix} a_N & a_{N-4} \\ a_{N-1} & a_{N-5} \end{vmatrix}$$

For each row that we are computing, we call the left-most element in the row directly above it the pivot element. For instance, in row b, the pivot element is a_{N-1} , and in row c, the pivot element is b_{N-1} and so on and so forth until we reach the bottom of the array.

To obtain any element, we negate the determinant of the following matrix, and divide by the pivot element:

$$\begin{vmatrix} k & m \\ l & n \end{vmatrix}$$

Where:

- **k** is the left-most element two rows above the current row.
- **l** is the pivot element.
- **m** is the element two rows up, and one column to the right of the current element.
- **n** is the element one row up, and one column to the right of the current element.

In terms of **k l m n**, our equation is:

$$v = \frac{(lm) - (kn)}{l}$$

Lab:

Task 1: Draw a functional block diagram showing the following signals: the desired output force as the input; the force F_{up} , applied to the bottom of the pantograph; the difference in displacement between the catenary and pantograph head; and the output contact force. Also, show blocks representing the input transducer, controller, actuator generating F_{up} , pantograph dynamics, spring described in the description of system and output sensor. All forces and displacements are measured from equilibrium.

Task 2: The diagram for the pantograph and catenary coupling is shown in figure (3(a)). Assume the simplified model shown in figure (3(b)), where the catenary is represented by the spring, K_{ave} . Find the transfer functions for the following cases below.

- a) $G_1(s) = Y_{cat}(s)/F_{up}(s)$, where $y_{cat}(t)$ is the catenary displacement and $f_{up}(t)$ is the upward force applied to the pantograph.

- b) $G_2(s) = Y_h(s) / F_{up}(s)$, where $y_h(t)$ is the pantograph head displacement.
- c) $G(s) = (Y_h(s) - Y_{cat}(s)) / F_{up}(s)$.

Task 3:

- a) For the transfer function you find in task 2 (c), Use the **dominant poles** from this transfer function and estimate percent overshoot, damping ratio, natural frequency, settling time, peak time, and rise time. Is the second order approximation valid?
- b) In Matlab, obtain a step response of $G(s)$ and compare the results calculated from above.

Task 4: Get the Simulink block diagram for figure 3(b) from model equations.

Task 5: Following the functional block diagram from task 1, we now want to create a pantograph active-control loop by adding the following components. (Input transducer $G_i(s) = 0.01$, controller $G_c(s) = K$, actuator $G_a(s) = 0.001$, pantograph spring $K_s = 82.3 \times 10^3$ N/m, and sensor $H_o(s) = 0.01$).

- a) Assemble a block diagram of the active pantograph control system from your functional block diagram from task 1 and pantograph dynamics found in task2.
- b) Find the **closed-loop transfer function** for transfer function for the block diagram in part (a) if $K=1000$.

Task 6: Get the Simulink block diagram for task 5 and get impulse and step response.

Task 7: Using your solution to task5 and **Routh-Hurwitz stability method**, find the range of controller gain, K that will keep the system stable.

Task 8: Using result from task5 (a)

- a) Find the system type
- b) Find the value of controller gain, K that minimizes the steady state force error.
- c) What is the minimum steady-state force error?

Task9: you have found the Impulse and step response for the desired system in task6.

- a) We want to design a feedback controller so that when the disturbance (W) is simulated by a unit step input, the output has a settling time less than 5 seconds and an overshoot less than 5%.
- b) Find the proportional, PI and PID controller parameters using the Ziegler-Nicholas transient-response method.

Lab Procedure:

1. Read the description of the system carefully and ask TA questions if you are not sure what to do in task 1. A sample functional block diagram of a feedback control is given in figure (2).
2. Write the model equations for figure 3(b) and take Laplace transform assuming initial conditions as 0. Find the respective transfer function.
3. Check the background section for dominant poles. Calculate the time domain specifications for the second order system. (refer to notes at the end of the text book)
4. Calculate the time domain design specs from Matlab output and compare with the ones calculated by hand.
5. Open Matlab-> Simulink, drag and drop necessary blocks and connect them appropriately starting from blocks shown in figure (4) and give input as step $u(t)$ for 1 to 1.5 secs. Save your graph and block diagram.
6. Get the closed-loop transfer function from functional block diagram in task 1 and dynamics of simplified system in task 2.
7. Open Matlab-> Simulink, drag and drop necessary blocks and connect them appropriately and give input as step $5t^2 u(t)$ for 1 to 10 secs. Save your graph and block diagram.
8. Check background section for theory about Routh-Hurwitz criteria. Find the range of K for which the system is stable.
9. Refer to web link about PID control design for task 9.

Post Lab: Write a report in abstract, objective, theory, procedure, results, conclusion and appendices format. All steps should be clearly mentioned. Include all plots and results.

References:

O'Connor, D.N., Eppinger, S.D., Seering, W.p., and Wormly, D.N. Active control of a High-Speed Pantograph. *Journal of Dynamic systems, Measurements and Control*, vol.119, March 1997, pp 1-4

Franklin, G.F., Powell, J.D., and Emami-Naeini, A. *Feedback Control of Dynamic Systems*. Pearson, 2015, 7/E

Nise, N.S., *Control Systems Engineering*. John Wiley & sons, Inc, 1995. 6/E

Useful Links

<http://vigir.ee.missouri.edu/~gdesouza/ece4310/index.htm> (intro slides)

http://vigir.ee.missouri.edu/~gdesouza/ece4310/Lab_Assignments/ECE_Feedback_Lab4.pdf (Lab 4 document)

http://ctms.engin.umich.edu/CTMS/index.php?aux=Basics_Matlab (Matlab basics)

http://ctms.engin.umich.edu/CTMS/index.php?aux=Basics_Simulink (Simulink basics)

https://en.wikibooks.org/wiki/Control_Systems/Routh-Hurwitz_Criterion (Stability Criteria)

<http://ctms.engin.umich.edu/CTMS/index.php?example=Introduction§ion=ControlPID> (Introduction to PID control design)