

Problem Wk.6.1.1: Proportional Delay

Part 1: delayPlusPropModel

We wish to analyze the behavior of the delay-plus-proportional controller described by

$$\text{omega}[n] = k1 * e[n] + k2 * e[n-1].$$

Use the functions and methods associated with the [sf module](#) to construct a procedure called `delayPlusPropModel` that takes two required arguments:

- the proportional gain k_1 applied to the error at time n , and
- the delay gain k_2 applied to the error at time $n-1$

and which returns a system function for a system whose input is the desired distance and whose output is the actual distance.

You can debug these in Idle by using the file `designLab06Work.py`.

Part 2: Gains

Consider four different values of k_1 : 10, 30, 100, and 300. For each value of k_1 , use `optimize.optOverLine` (from the [optimize module](#)) to determine the value of k_2 that minimizes the magnitude of the least stable pole.

Enter 1 decimal place of accuracy for k_2 and 2 decimal places for the pole magnitude.

1.	k1	k2	magnitude of dominant pole
	10	<input type="text"/>	<input type="text"/>
	30	<input type="text"/>	<input type="text"/>
	100	<input type="text"/>	<input type="text"/>
	300	<input type="text"/>	<input type="text"/>

MIT OpenCourseWare
<http://ocw.mit.edu>

6.01SC Introduction to Electrical Engineering and Computer Science
Spring 2011

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.