**ENGR 4020: Mechatronics System Design PID Tuning**

***Tuning Rules***

If a mathematical model of the plant is available, it is always possible to apply various classical control design techniques, such as root locus or Bode plots (later), in order to determine controller parameters that meet desired design specifications. On the other hand, if the plant is so complex that an accurate mathematical model cannot be derived easily, these classical control design techniques become difficult to apply in designing a general linear controller. The alternative is to apply experimentally proven techniques for choosing the controller parameters, resulting in at least a starting point from which the controller can then be tuned.

In the early 1940’s, Ziegler and Nichols [1] published a work on the topic of tuning rules for determining PID controller gains based on observable behaviors of the system, such as the transient response of a given plant. The motivation was primarily for applications to systems where the mathematical model of the plant was not known precisely, so that tuning of the PID parameters can be performed on site via experimental analysis with the plant. These tuning rules are equally applicable to systems with known models. The Ziegler-Nichols tuning rules consist basically of two methods, both of which are aimed at obtaining MP(%) = 25% in the step response.

**Tuning Method**: The second Ziegler-Nichols method requires the following *a priori* information.

Assumption: In a unity feedback configuration, the plant output exhibits sustained oscillations, that is, resembles a pure sinusoid for some proportional gain value KP.

This method consists of increasing only the proportional gain, KP in a unity feedback configuration precisely to the point where the output first exhibits sustained oscillations. Recall from the root locus analysis that the value for KP at which the axis crossover occurs and is referred to as the critical gain, Kcr. When KP=Kcr, the system output will behave as a sinusoid at the frequency *fo* Hz. The tuning rules for the controller are given in Table 1.

Table 1: Tuning rules for second Ziegler-Nichols method

|  |  |  |  |
| --- | --- | --- | --- |
| Controller Type | KP | KI | KD |
| P |  | 0 | 0 |
| PI |  |  | 0 |
| PID |  |  |  |

Plants for which the Ziegler-Nichols, the tuning rules can be applied will exhibit maximum closed loop step response percent overshoot values ranging between between 10% and 60%. The average maximum percent overshoot is 25% based on the tables given.

**Procedure**

1. For the plant

the transfer function of the compensator

and the typical unity feedback configuration.

**Design Objective:** Design a PID compensator such that the resultant closed loop system will have zero steady-state error, , a percent overshoot and a 2% settling time second.

**Procedure Answer Sheet**

1. e. Tuning method

Kcr = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *fo* = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

KP= *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* KI= *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* KD= *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

1. g. Tuned PID parameters

KP= *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* KI= *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_* KD= *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

**References**

[1] J.G. Ziegler and N.B. Nichols, “Optimum Settings for Automatic Controllers,” *Transactions of the A.S.M.E.*, vol. 64, pp. 759-768, November 1942.

[2] J.K. Hurtig, “ PID Lab”, ECE 445.

**Simulink Based PID Tuning**

1. Type the command simulink into the command window of MATLAB.
2. Open a blank model
3. Click on the library browser (red square)

A screenshot of a cell phone

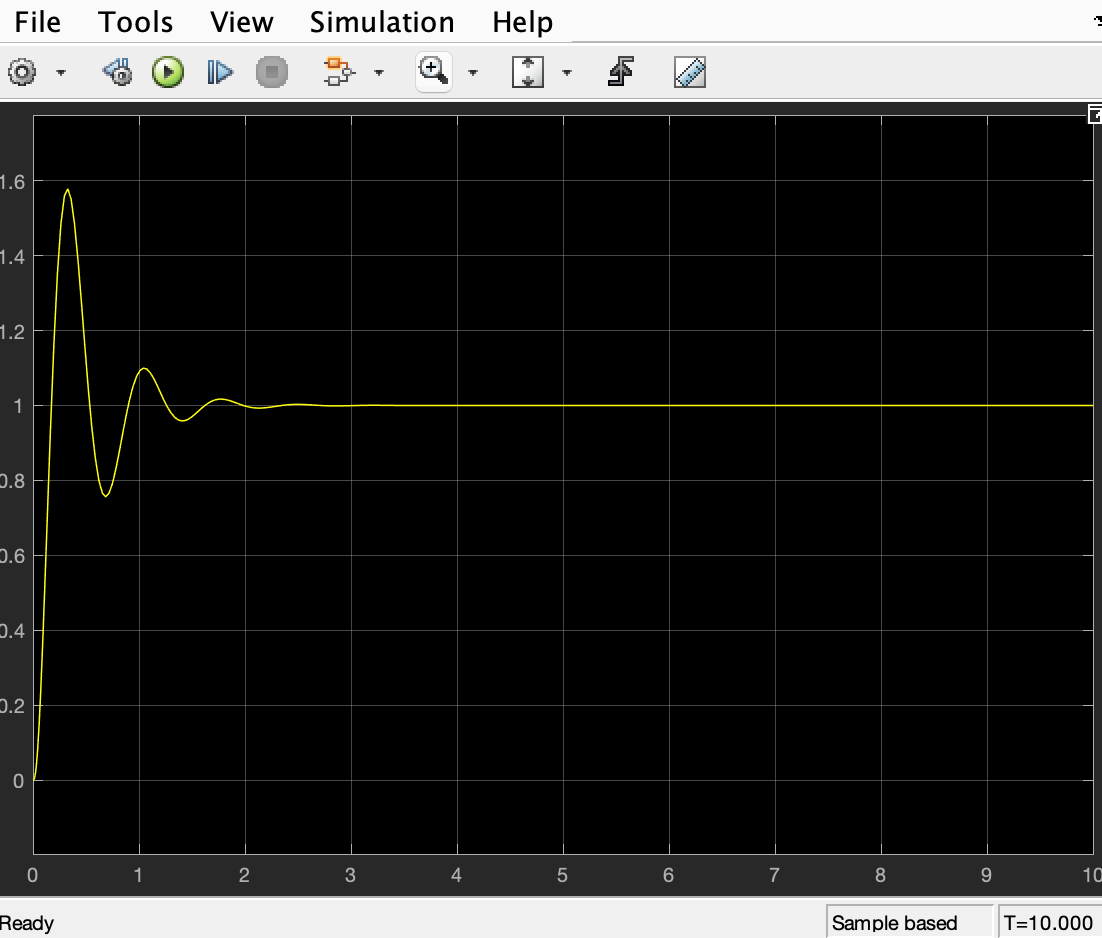
Description automatically generated

1. Use the blocks from this library browser to construct the following block diagram

A screenshot of a cell phone

Description automatically generated

1. The summer signs can be changed by double clicking the summer (should be |+-)
2. Set step time to 0 in the step
3. The transfer function is set by double clicking and entering vectors for the numerator and denominator
4. Put in our initial guesses for PID gains from Ziegler Nichols Method
5. Double click the scope, then click run (the play button). You should get a response like the figure below



1. Double click the PID controller
2. Click Tune
3. Play with the GUI to tune your controller. The GUI should look like the figure below

A screenshot of a computer

Description automatically generated

1. You may want to select options and choose to focus on reference tracking. Don’t be scared to reset if needed! Here’s my tuned response.

A close up of a map

Description automatically generated

1. Finally, click update block, and run the Simulink simulation again to view the result on the scope. You can also send data to workspace to make nicer plots in MATLAB if you want.