Robust Control Systems Design Using Matlab

Zainal Abidin[*)]

Electrical Department Islamic University of Lamongan
email : zainalabidin@unisla.ac.id

ABSTRACT

A steady control system is carried out through optimal design and also using estimation theory, controls designed for process models generally work well, but also sometimes experience failure in a system.
The most important thing in the feedback control test model is to improve the robustness performance of the system. Strong control systems make it possible to determine more or less directly the uncertainty of a system, and make it possible to predict possible trade-offs between robustness and closed-loop performance.

Keywords* : *robust, control system, Matlab

I. INTRODUCTION

Creating a highly accurate control system design in the presence of inaccuracies is a common problem. Usually industrial system parameters are accurate and in accordance with system parameters. However, practically the parameters of a system can not be known with certainty and varies based on time. A good control system must be designed more adequately in line with the variation in the parameters of a system installation. The robust control system maintains stability and performance over varying parameters and from various system disturbances.

In this discussion we will consider the commonly used Proportional Integral Derivative (PID) controls. The system design system feedback setting is shown in fig. 1. The system has a Gp prefilter. The role of prefilter in system optimization has been discussed and implemented [1].

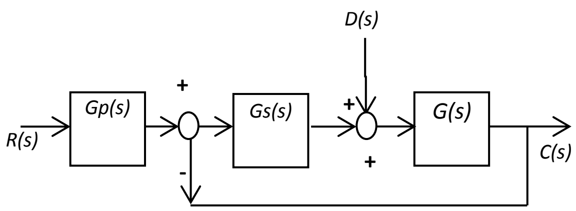


Fig.1. Feedback control system with reference and disturbance and a Prefilter

II. Methodology

Robust PID Control

In the discussion we has the case :

$$G_c(s) = \frac{K_3\,s^2 + K_1 s + K_2}{s} \qquad (1)$$

The control system with Proportional Integral Derivative (PID) is not a rational function. It will be difficult when inserting the PID controller into Matlab by making standard numerators and denominators. The problem can be solved by a conv function rather than a series function in system modification.

The determination is to select the parameters K_1, K_2, and K_3 to meet the system specifications and have stable resistance properties. However, it is still uncertain how to select parameters on the PID controller to obtain certain robustness characteristics. In this case some illustrative examples will be shown which make it possible to select parameters iteratively and verify robustness with a simulation model. Matlab simulation can help this process because all designs and simulations can be run using program listings and can easily be run repeatedly.

.

III. MODELLING AND RESULT

For modelling discussion, model system implementation using Matlab modelling.

Case study : Robust control of temperature

Look at the feedback control system in figure 1, where :

$$G(s) = \frac{1}{(s+c_0)^2} \quad (2)$$

and the nominal value of c_0 is , $c_0 = 1$

The compensator design model is based on c0 = 1 and considers the resistance model by simulation. Our design specifications are proposed as follows:

(i) Settling time Ts = 0.5 seconds
(ii) ITAE performance for step input [1]

In this simulation, it will not use a prefilter to meet specifications (ii) but on the contrary will show that the system performance is acceptable with low overshoot and is obtained by increasing system gain.

The closed-loop transfer function is :

$$T(s) = \frac{K_3\, s^2 + K_1 s + K_2}{s^3 + (2+K_3)s^2 + (1+K_1)s + K_2} \quad (3)$$

Characteristic equation formula is :

$$1 + K^*\left(\frac{s^2+as+b}{s^3}\right) = 0$$

Where

$$K^* = K_3 + 2$$
$$a = \frac{1+K_1}{2+K_3}$$
$$b = \frac{K_2}{2+K_3}$$

Settling time requirement Ts < ½ leads us choose that roots of $s^2 + as + b$ to the left of the s= $-\zeta\omega_n$ = -8 line in the s-plane, as shown in figure 2, to ensure that the locus travels into the required performance region. We have chosen a = 16 dan b =70 to ensure the locus travels past the s=-8 line.

We select a point on the root locus in the performance region, and using the ***rlocfind*** function, we find gain K^* and value of ω_n of formulation . For the point have chosen that, K^*=118

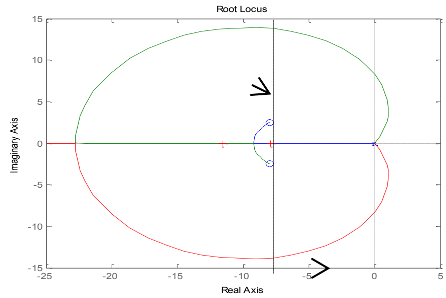


Figure 2. Root locus for the PID compensated temperature controller

Then, with K^*, a, and b we can solve for the PID coefficient as follows :

$K_3 = K^* - 2 = 116$
$K_1 = a(2+K_3) - 1 = 1187$
$K_2 = b(2+K_3) = 8260$

To achieve system overshoot performance it is necessary to follow the following steps, We use the K cascade strengthening which will be selected by an iterative method using the step function. This is illustrated in figure 3. The step response corresponding to K = 5 has an overshoot of 2%. With the added gain of K = 5, the final PID controller approach is:

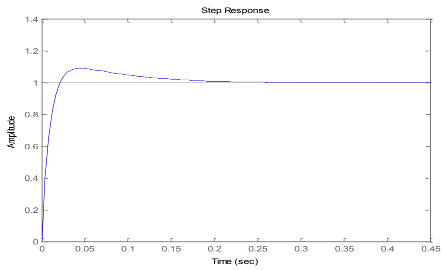
$$G_c(s) = K\frac{K_3 s^2 + K_1 s + K_s}{s} \quad (4)$$
$$= 5\frac{116s^2 + 1187s + 8260}{s}$$

n this case the prefilter is not used as in the previous design. [1]. Another solution is to increase system gain for excellent transient response. Next we can consider the problem of resistance to changes in the system parameter c0.

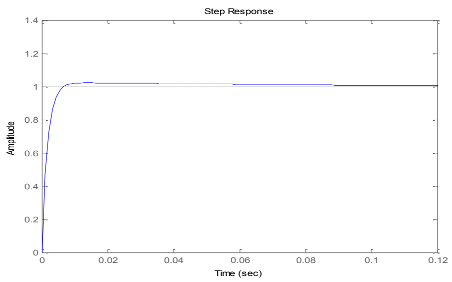
Investigations in maintaining robustness of our design consisted of response analysis using the PID control given in equation (4) for a wide variety of system parameters c0∈ [0,1,10]. The simulation results are shown in Figure 4. A program script is written to calculate the step response for a particular c0. Better to place c0 input at command prompt level to make scripts more interactive.

The simulation results show that the PID design is strong against changes in c0. The step difference in response for c0 ∊ [0,1,10] is barely noticeable in the plot. If the results show otherwise, it may be possible to redesign until acceptable performance is achieved.

There are several control design methods that incorporate robustness right into the design process. Matlab's interactive capabilities make it possible to check the robustness of the system by simulation, although this is clearly not the most desirable approach when it comes to designing control systems.



a.Uncompensated control



b. compensated control

Figure 3. Step response for the PID temperature controller

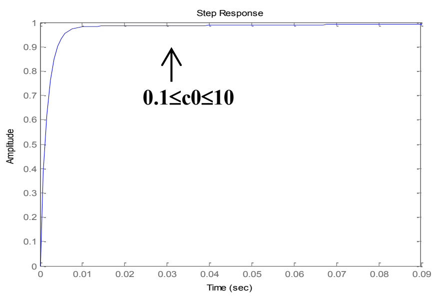


Figure 4. Robust PID controller analysis with variation C_0

IV. CONCLUSION

There are several control design methods for incorporating robustness directly into the design process. By using Matlab, the design process is more interactive which allows to perform system endurance tests with simulations, although in fact this is not the best thing in doing system design.

REFERENCE :

[1] Richard C. Dorf, Robert H. Bishop.2010 . *Modern Control System. 12th Edition*

[2] Robert H. Bishop. 2005.*Modern Control System Analysis and Design Using Matlab*. Wesley Publishing Company

[3] Katsuhito, Ogata. 2010. Modern Control Engineering.Prentice Hall

ISSN : 25020986 | E-ISSN : 26860635