

School of Electrical and Electronic Engineering

**EEEN 40010**

**Control Theory**

|  |  |
| --- | --- |
| Student Name: | Paul Doherty |
| Student Number: | 10387129 |
| Date: | 21/10/2013 |
| Laboratory Title: | Root Locus |

# Design Problem 1:

A controller of the form was designed for a plant with a transfer function of to meet the following specifications:

(i) Closed-loop system is stable

(ii) Steady state error to a step input does not exceed 5%

(iii) 2% settling time does not exceed 0.75 sec.

(iv) PO% does not exceed 30%

This was done by first finding the damping ratio that would meet the percentage overshoot (PO) requirement. This was done by using the following equation: . Setting PO to the desired maximum overshoot and by utilising algebraic methods we solve for getting a damping ratio of .

An approximation for the real part of the dominant pole of the controller was then calculated using the approximate formula: . Subbing the calculated value of and the 2% settling time of that we are required to meet in the given specification we can solve to find that the real part of the dominant pole of the controller needs to be , it is also clear that the natural frequency is .

Now by employing MATLAB to view the step response of the closed loop system and by using the root locus drawing rules of the open loop system we can design a controller that will meet our specification.

Let us begin our design by first viewing the step response of the closed loop system of the plant without any control.

Since it was already concluded that the dominant pole of my controller would need to lie to the left on the real axis to meet the given specification the first step of my design was to place the controller pole at. Applying the drawing rules for the root locus in MATLAB we can see the following root locus plot of the open loop system:



It is clear that the closed loop system has a real pole and a dominant complex conjugate pair. The step response of the closed loop system is as follows.



It can be seen from the step response that this falls way short of the desired spec. but it is a start. A zero will need to be added to the controller at a place that agrees with the calculated approximate damping ratio from the specification. The zero will also need to be places to the left of but not so as to cancel our controller open loop pole. Placing the zero at yields the following root locus plot:



This root locus plot shows a much more desirable result than the previous plot as it can be seen the root locus begin and terminate on the real axis. However the chosen position for the zero is not far enough into the left half plane (LHP). Pushing it further in to the LHP will cause the root locus to enter the damping ratio region of . Through trial and error of experimenting with different zero locations I found that placing a zero at and then using the **rlocfind** command in MATLAB and selecting a suitable place for our gain we can successfully find all the elements in our zero/pole controller that will satisfy the specification:



This gives a gain value of .This yields a step response that satisfies the given specification:



The final controller used in the design is as follows:

# Design Problem 2:



Above is the chosen operating point of the aircraft in steady flight. This operating point makes perfect sense if the aircraft is in steady state flight because it would be expected that , the velocity of the aircraft in the x direction (longitudinal axis of the aircraft from tail to nose) would not be equal to as the plane is in steady flight. Using this intuition in an ideal world we would like the velocities in the y-axis and z-axis to be zero. i.e. and . In steady state ideal conditions it would also be desirable for there to be no angular velocity to be acting on the aircraft. (No roll), (No pitch), (No yaw).

For the force acting on the aircraft during steady state flight it is expected that there would be a force acting on the aircraft in the as it is expected that the aircraft is moving forward in steady state flight. Similarly we would expect a force to be acting on the aircraft in the direction during steady state flight as we assume the aircraft is in the air during flight so a non-zero force in this direction is a reasonable operating point parameter. In an ideal world we would like force in the y-direction to be zero i.e. . So this is also a reasonable parameter for the chosen operating point during steady state flight. This will commonly offset in non-steady state flight due to cross wind.

In steady state flight it reasonable to say that there will be no moments acting on the system and so , , and . Similarly in steady state flight we can say that the Euler angles, i.e. the angles that relate the pitch, yaw and roll of the aircraft to the body fixed frame of the aircraft. Because of this we can that because the pitch, yaw and roll angle are all equal to zero in our chosen operating point then we can assume that the Euler angles are also all equal to zero. It for these reasons the chosen operating point of the aircraft in steady state flight makes sense.

# Design Problem 3:

# Design Problem 4:

The offset of the angle of attack is related to and to the sideslip angle and the offset of the sideslip angle is related to the offset of *v* byemploying the Maclaurin Series.

# Design Problem 5a:

The longitudinal dynamics of the aircraft can be described as:



When we assume that the thrust is constant so that the offset from the operating point of thrust setting *T* is zero and we make the steady flight assumption that *u* is constant so that the offset of *u* from the operating point value is also zero. The following parameters are then selected for a certain Boeing aircraft travelling at a suitable speed:



The transfer function of the linear, time-invariant system with input equal to elevator offset setting and output equal to offset pitching angle was found by applying the Laplace Transform as follows:

# Design Problem 5b:

The actuator which controls the elevator setting is described by the simple model:



The transfer function of the actuator was derived by again employing the Laplace Transform:

The transfer function is:

Thus the transfer function of the series combination of the plant and the actuator is:

# Design Problem 6:

The poles of the system are at: , ,.

There is one zero of the system at: .

The step response of the closed loop system is as follows:



This step response indeed looks strange. But when the loop-the-loop manoeuvre is considered it can be see how this relates to the pitch of the aircraft. The first peak responds to loop-the-loop manoeuvre as the pitch increases to a maximum as the aircraft approaches the point where it is completely “upside-down” . The trough that follows represents the aircraft returning to the normal flight pattern. Finally the ripples that can be seen after this are the aircraft oscillating before it reaches its steady-state flight point.

# Design Problem 7:

By employing the root locus design procedure, a PID controller was designed for the plant; which models the pitch of the aircraft, to behave as follows:

(i) The aircraft pitch tracks step changes in the desired pitch angle offset with zero steady state error

(ii) The closed-loop system has an overshoot not exceeding 50% and a 2% settling time not exceeding 10 sec.

The first step is to approximate the damping ratio and the dominant pole placement of the controller pole. This is done with the same procedure used in design problem 1. . Subbing the required percentage overshoot into the equation and solving for we find that the damping ratio is to be .

Using the approximate equation for the dominant pole placement of the controller pole we find.

Beginning by examining the root locus of the closed loop system without control gives the following root locus plot:



It is clear to see the system not fail safe. From the above calculation the controller pole was chosen to be placed at . As the open loop system now has four poles, three real poles and a complex conjugate pair, it is desired that two zeros be introduced in order to keep the asymptotes in desired position to meet the system specification. A PID controller is needed: . The two zeros will need to be positioned relatively close to the conjugate pair to keep the system fail safe. Let us begin by trying we find the following root locus plot:



It is clear that this system is not fail safe either as it can be seen that the root locus asymptotes drift into the right half plane for a period of time. Values of the controller zeros will need to be chosen closer to the selected pole of the controller.

After some experimentation it can be seen that choosing the value zeros to be the closed loop system meets the desired specification and we achieve the following root locus plot and employing the MATLAB command **rlocfind()** we get a estimation for the gain .



This gives me an estimated gain of . Utilising this and viewing the closed loop system step response the behaviour of the system can be viewed:



From this step response we see that the and the steady state error meets the specification but it can be seen that the system is too slow. The gain need to be raised. By experimentally raising the gain it is found that a gain of causes the system to meet the speed requirements without violating the. Below is the step response of the system:



This step response of the closed loop system meets our specification. The settling time is nicely inside the requirement and the is comfortably under the requirement.

# Design Problem 8:

If the controller gain fails and the gain becomes zero then the closed loop step response of the system becomes:



If the controller gain goes to zero it can be seen that the closed loop system will simply not react to anything and completely stop and come ‘offline’.

If the controller gain fails and becomes really big then the closed loop step response of the system becomes:



If the controller gain becomes too big then it can be seen that the closed loop system is in fact fail safe despite reacting extremely radically, as can be seen by the system overshoot and the speed of the response, before it comes to rest at unity.