EECE 460 : Control System Design

SISO Pole Placement

Guy A. Dumont

UBC EECE

January 2011

1 Preview

2 Polynomial Pole Placement

3 Constraining the Solution

4 PID Design

5 Smith Predictor

Preview

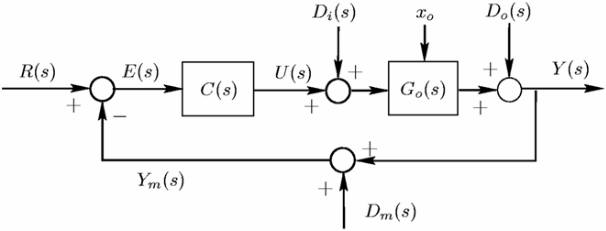
- Formal control design method
- The key synthesis question is:

Given a model, can one systematically synthesize a controller such that the closed-loop poles are in predefined locations?

- This is indeed possible through pole assignment
- In EECE360, we saw a method for assigning closed-loop poles through state feedback (see review notes)
- Here we shall see a polynomial approach based on transfer functions
- This material is covered in Chapter 7 of the textbook

Polynomial Pole Placement

- Consider the nominal plant $G_0(s) = \frac{B_0(s)}{A_0(s)}$ and the controller $C(s) = \frac{P(s)}{L(s)}$ in a simple feedback configuration as below



$$P(s) = p_{n_p}s^{n_p} + p_{n_p-1}s^{n_p-1} + \ldots + p_0$$
$$L(s) = l_{n_l}s^{n_l} + l_{n_l-1}s^{n_l-1} + \ldots + l_0$$
$$B_o(s) = b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \ldots + b_0$$
$$A_o(s) = a_n s^n + a_{n-1}s^{n-1} + \ldots + a_0$$

The Design Objective

- Consider a desired closed-loop polynomial

$$A_{cl}(s) = a^c_{n_c} s^{n_c} + a^c_{n_c-1} s^{n_c-1} + \cdots + a^c_0$$

Design Objective

Given A_0 and B_0, can we find P and L such that the closed-loop characteristic polynomial is $A_{cl}(s)$?

- The characteristic equation $1 + G_0 C = 0$ gives the characteristic polynomial $A_0 L + B_0 P$. Does there exist $P(s)$ and $L(s)$ such that

$$A_0(s)L(s) + B_0(s)P(s) = A_{cl}(s)$$

A Simple Example

- Let $A_0(s) = s^2 + 3s + 2$ and $B_0(s) = 1$
- Consider $P(s) = p_1 s + p_0$ and $L(s) = l_1 s + l_0$
- Then

$$A_0(s)L(s) + B_0(s)P(s) = (s^2 + 3s + 2)(l_1 s + l_0) + (p_1 s + p_0)$$

- Let $A_{cl}(s) = s^3 + 3s^2 + 3s + 1$
- Equating coefficients gives

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 3 & 1 & 0 & 0 \\ 2 & 3 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{bmatrix} \begin{bmatrix} l_1 \\ l_0 \\ p_1 \\ p_0 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ 3 \\ 1 \end{bmatrix}$$

A Simple Example

- The above 4×4 matrix is non-singular, hence it can be inverted and we can solve for p_0, p_1, l_0 and l_1
- Doing so, we find $l_0 = 0$, $l_1 = 1$, $p_0 = 1$ and $p_1 = 1$
- Hence the desired characteristic polynomial is achieved by the controller

$$C(s) = \frac{P(s)}{L(s)} = \frac{s+1}{s}$$

- Note that this is a PI controller!
- Note that in order to solve for the controller, a particular matrix has to be non-singular
- This matrix is known as the Sylvester matrix

Sylvester's Theorem

Theorem (Sylvester's Theorem)

Consider the two polynomials

$$A(s) = a_n s^n + a_{n-1} s^{n-1} + \ldots + a_0$$
$$B(s) = b_n s^n + b_{n-1} s^{n-1} + \ldots + b_0$$

and the eliminant matrix (also known as Sylvester matrix)

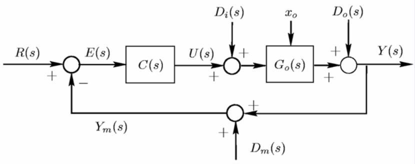
$$\mathbf{M_e} = \begin{bmatrix} a_n & 0 & \cdots & 0 & b_n & 0 & \cdots & 0 \\ a_{n-1} & a_n & \cdots & 0 & b_{n-1} & b_n & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_0 & a_1 & \cdots & a_n & b_0 & b_1 & \cdots & b_n \\ 0 & a_0 & \cdots & a_{n-1} & 0 & b_0 & \cdots & b_{n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_0 & 0 & 0 & \cdots & a_0 \end{bmatrix}$$

Then $A(s)$ and $B(s)$ are coprime (i.e. have no common factors) if and only if $\det(\mathbf{M_e}) \neq \mathbf{0}$

Pole Placement Design

Lemma (Pole Placement Design)

- *Consider the feedback loop*



with $G_0(s) = \frac{B_0(s)}{A_0(s)}$ *and* $C(s) = \frac{P(s)}{L(s)}$

- *Assume* $A_0(s)$ *and* $B_0(s)$ *are coprime with* $n = \deg A_0(s)$
- *Let* $A_{cl}(s)$ *be an arbitrary polynomial of degree* $n_c = 2n - 1$.

Then, there exist $P(s)$ *and* $L(s)$ *with degrees* $n_p = n_l = n - 1$ *such that*

$$A_0(s)L(s) + B_0(s)P(s) = A_{cl}(s)$$

The Diophantine Equation

- The equation

$$A_0(s)L(s) + B_0(s)L(s) = A_{cl}(s)$$

  is called Diophantine equation after Greek mathematician Diophantus[1].
- In its matrix form, it is also referred to as the Bezout Identity. It plays a central role in modern control theory.
- The previous Lemma presents the solution for the minimal complexity controller.
- The controller is said to be biproper, i.e. $\deg P(s) = \deg L(s)$
- Usually, the polynomials $A_0(s)$, $L(s)$ and $A_{cl}(s)$ are monic i.e. coefficient of highest power of s is unity.

[1]Diophantus, often known as the 'father of algebra', is best known for his Arithmetica, a work on the solution of algebraic equations and on the theory of numbers. However, essentially nothing is known of his life and there has been much debate regarding the date at which he lived.

Forcing Integration

- As seen before, it is very common to force the controller to contain an integrator
- To achieve this, we need to rewrite the denominator of the controller as

$$L(s) = s\bar{L}(s)$$

- The Diophantine equation

$$\begin{aligned} A_0(s)L(s) + B_0(s)P(s) &= A_{cl}(s) \\ A_0(s)s\bar{L}(s) + B_0(s)P(s) &= A_{cl}(s) \end{aligned}$$

can then be rewritten as

Forcing Integration

$$\bar{A}_0(s)\bar{L}(s) + B_0(s)P(s) = A_{cl}(s) \quad \text{with} \quad \bar{A}_0(s) = sA_0(s)$$

Forcing Integration

- Note that because we have increased the degree of the l.h.s. by 1, now we have $\deg A_{cl} = 2n$ and need to add a closed-loop pole.
- Also, since $\deg L = \deg \bar{L} + 1$, to keep a bi-proper controller, we should have $\deg P = \deg L = \deg \bar{L} + 1$

Forcing Pole/Zero Cancellation

- Sometimes it is desirable to force the controller to cancel a subset of stable poles or zeros of the plant model
- Say we want to cancel a process pole at $-p$, i.e. the factor $(s+p)$ in A_0, then $P(s)$ must contain $(s+p)$ as a factor.
- Then the Diophantine equation has a solution if and only if $(s+p)$ is also a factor of A_{cl}
- To solve the resulting Diophantine equation, the factor $(s+p)$ is simply removed from both sides

Example 1

Consider the system

$$G_0(s) = \frac{3}{(s+1)(s+3)}$$

- The unconstrained problem calls for $\deg A_{cl} = 2n-1 = 3$. Choose $A_{cl}(s) = (s^2+5s+16)(s+40)$. The polynomials P and L are of degree 1. The Diophantine equation is

$$(s+1)(s+3)(s+l_0)+3(p_1 s+p_0) = (s^2+5s+16)(s+40)$$

which yields $l_0 = 41$, $p_1 = 49/3$ and $p_0 = 517/3$ and the controller $C(s)$ is

$$C(s) = \frac{49s+517}{3(s+41)}$$

Example 1

- Consider now the constrained case where we want the controller to contain integration. We need to add a closed-loop pole. Let us say that we want the controller to cancel the process pole at -1. The Diophantine equation then becomes

$$(s+1)(s+3)s(s+l_0)+3(s+1)(p_1 s+p_0)=(s+1)(s^2+5s+16)(s+40)$$

which after eliminating the common factor $(s+1)$ yields $l_0=42$, $p_1=30$ and $p_0=640/3$ and the controller $C(s)$ is

$$C(s)=\frac{(s+1)(90s+640)}{3s(s+42)}$$

Example 2

- Assume the plant with nominal model $G_0(s) = \frac{1}{s-1}$
- We want a controller that stabilizes the plant and tracks with zero steady-state error a 2 rad/s sinusoidal reference of unknown amplitude and phase
- The requirement for zero steady-state error at 2 rad/s implies

$$T_0(\pm j2) = 1 \Rightarrow S_0(\pm j2) = 0 \Leftrightarrow G_0(\pm j2)C(\pm j2) = \infty$$

- This can be satisfied if and only if $C(\pm j2) = \infty$, i.e. if $C(s)$ has poles at $\pm j2$
- Thus

$$C(s) = \frac{P(s)}{L(s))} = \frac{P(s)}{(s^2+4)\bar{L}(s)}$$

Example 2

- Since we have added 2 poles to the controller, we now need $\deg A_{cl} = 2n - 1 + 2 = 3$, $\deg P(s) = 2$ and $\deg \bar{L} = 0$
- This leads to $\bar{L}(s) = 1$ and $P(s) = p_2 s^2 + p_1 s + p_0$
- Choosing $A_{cl}(s) = (s^2 + 4s + 9)(s + 10)$, the Diophantine equation is

$$\bar{A}_0(s)\bar{L}(s) + B_0(s)P(s) = A_{cl}(s$$

$$(s-1)(s^2+4) + (p_2 s^2 + p_1 s + p_0) = (s^2 + 4s + 9)(s + 10)$$

$$s^3 + (p_2 - 1)s^2 + (p_1 + 4)s + p_0 - 4 = s^3 + 14s^2 + 49s + 90$$

- This yields $p_2 = 15$, $p_1 = 45$, $p_0 = 94$, i.e.

$$C(s) = \frac{15s^2 + 45s + 94}{(s^2 + 4)}$$

- This is not a PID!

Pole-Placement PID Design

- We already have seen how to derive a PID controller using a model-based technique such as the Dahlin controller, which is actually a special case of pole placement. Here we generalize this design

Lemma

The controller $C(s) = \frac{n_2 s^2 + n_1 s + n_0}{d_2 s^2 + d_1 s}$ and the PID $C_{PID}(s) = K_P + \frac{K_I}{s} + \frac{K_D s}{\tau_D s + 1}$ are equivalent when

$$
\begin{aligned}
K_P &= \frac{n_1 d_1 - n_0 d_2}{d_1^2} \\
K_I &= \frac{n_0}{d_1} \\
K_D &= \frac{n_2 d_1^2 - n_1 d_1 d_2 + n_0 d_2^2}{d_1^3} \\
\tau_D &= \frac{d_2}{d_1}
\end{aligned}
$$

Pole-Placement PID Design

- To obtain a PID controller, we then need to use a delay-free second-order nominal model for the plant and design a controller forcing integration
- We thus choose

$$\begin{aligned} \deg A_0(s) &= 2 \\ \deg B_0(s) &\leq 1 \\ \deg \bar{L}(s) &= 1 \\ \deg P(s) &= 2 \\ \deg A_{cl} &= 4 \end{aligned}$$

Example 3

- Consider the nominal plant

$$G_0(s) = \frac{2}{(s+1)(s+2)}$$

- Synthesize a PID that gives closed-loop dynamics dominated by (s^2+4s+9)
- Adding the factor $(s+4)^2$ to the above so that $\deg A_{cl} = 4$
- With $B_0(s) = 2$ and $A_0(s) = s^2+3s+2$ the following controller is obtained

$$C(s) = \frac{P(s)}{s\bar{L}(s)} = \frac{14s^2+59s+72}{s(s+9)}$$

- From the previous Lemma we find that this a PID controller with

$$K_P = 5.67 \qquad K_I = 8 \qquad K_D = 0.93 \qquad \tau_D = 0.11$$

Example 4

$$G_o(s) = \frac{B_o(s)}{A_o(s)} = \frac{2}{(s-1)(s+2)} = \frac{2}{s^2+s-2} \tag{1}$$

Find a controller of minimal complexity which stabilizes the plant, yields zero steady state error for step disturbances and generates a closed loop with natural modes which decay at least as fast as e^{-3t}.

Solution 7.1. *With experience, design problems such as this can be tackled by trial and error, using tools such as MATLAB* **rltool**. *However, a systematic approach gives a direct solution. Here we follow the latter approach.*

We begin by recalling that we can choose an arbitrary set of closed loop natural frequencies if the closed loop characteristic polynomial, $A_{cl}(s)$ has degree at least equal to 2n-1 (the degree of $A_o(s)$, the plant nominal model denominator). However, since we want to force zero steady state error at d.c., an additional degree-of-freedom is needed. We are aiming for a minimum complexity controller; hence, we choose the degree of $A_{cl}(s)$ equal to 4.

In addition, the roots of $A_{cl}(s)$ should be to the left of $s = -3$ to ensure that the response time specification is met. Say we choose

$$A_{cl}(s) = (s^2+6s+9)(s+4)(s+5) = s^4 + 15s^3 + 83s^2 + 201s + 180 \tag{12}$$

Example 4

We can now solve the Diophantine equation

$$\underbrace{(s-1)(s+2)}_{A_o(s)}\ \underbrace{s(s+\ell_0)}_{L(s)} + \underbrace{2}_{B_o(s)}\ \underbrace{(p_2 s^2 + p_1 s + p_0)}_{P(s)} = s^4 + 15s^3 + 83s^2 + 201s + 180 \tag{13}$$

The solution to this equation can found using the MATLAB function ***paq.m*** *distributed with the book and available on the web site.*

```
>>Ao=[1 1 -2];Am=[Ao 0];Bo=2;Acl=[1 15 83 201 180];
>> [Lm,P]=paq(Am,Bo,Acl); L=[Lm' 0];C=tf(P',L);
```

We finally obtain

$$P(s) = 35.5s^2 + 114.5s + 90 \tag{14}$$

$$L(s) = s^2 + 14s \tag{15}$$

$$C(s) = \frac{35.5s^2 + 114.5s + 90}{s^2 + 14s} \tag{16}$$

The Smith Predictor

- Of course, in general pole placement design will not yield a PID
- This is particularly the case when the plant model contains time delay
- Time delays are common in many applications, particularly in process control applications
- For such cases, the PID controller is far from optimal
- For the case of open-loop stable plants with delay, the Smith predictor provides a useful control strategy

Handling Time Delay

- Consider a delay-free model $\bar{G}_0(s)$ with a controller $\bar{C}(s)$
- The complementary sensitivity function is then

$$\bar{T} = \frac{\bar{G}_0\bar{C}}{1+\bar{G}_0\bar{C}}$$

- Assume we now have the plant

$$G_0(s) = e^{-\tau s}\bar{G}_0(s)$$

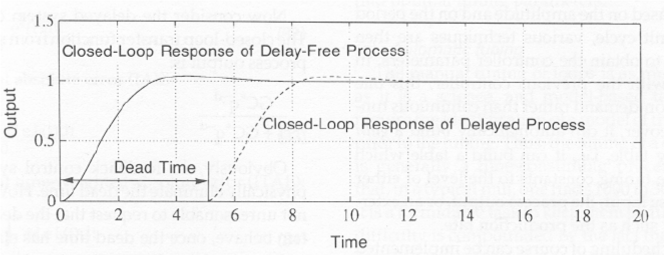
- How can we get a controller $C(s)$ for $G_0(s)$ from $\bar{C}(s)$?

Handling Time Delay

- The complementary sensitivity function for the delayed plant is

$$T = \frac{G_0 C}{1 + G_0 C}$$

- It is intuitively appealing to design the controller for the delayed system such that the closed-loop response for the delayed system is simply the delayed closed-loop response of the delay-free system



- Mathematically, it means that we want

$$T(s) = e^{-\tau s} \bar{T}(s)$$

The Smith Predictor

- The previous relationship between the two complementary sensitivities means

$$\frac{G_0 C}{1 + G_0 C} = e^{-\tau s} \frac{\bar{G}_0 \bar{C}}{1 + \bar{G}_0 \bar{C}}$$

i.e.

$$\frac{e^{-\tau s} \bar{G}_0 C}{1 + e^{-\tau s} \bar{G}_0 C} = e^{-\tau s} \frac{\bar{G}_0 \bar{C}}{1 + \bar{G}_0 \bar{C}}$$

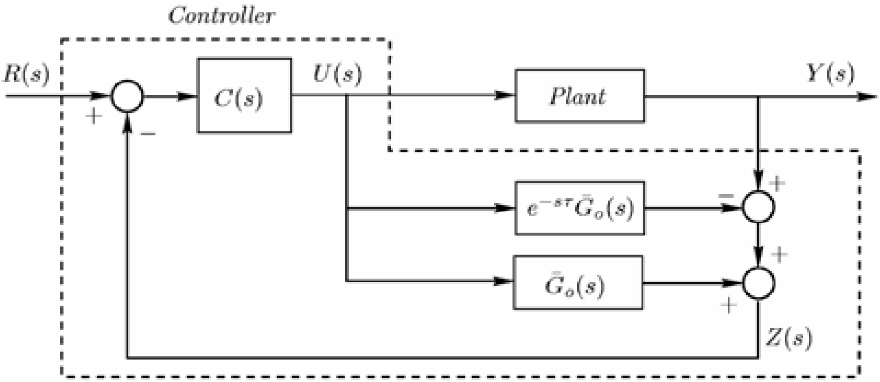
- Solving for C yields

The Smith Predictor

$$C(s) = \frac{\bar{C}}{1 + \bar{G}_0 \bar{C}(1 - e^{-\tau s})}$$

The Smith Predictor

- The Smith predictor corresponds to the structure below



The Smith Predictor

- One cannot use the above architecture when the open loop plant is unstable. In the latter case, more sophisticated ideas are necessary.
- There are significant robustness issues associated with this architecture. These will be discussed later.
- Although the scheme appears somewhat ad-hoc, its structure is a fundamental one, relating to the set of all possible stabilizing controllers for the nominal system.
- We will generalize this structure when covering the so-called Youla parametrization or Q-design.

Example 5

Solved Problem 7.5. *Consider a plant having a nominal model given by*

$$G_o(s) = \frac{e^{-2s}}{s+1} = e^{-2s}\bar{G}_o(s) \tag{7}$$

Build a Smith predictor so that the settling time for a step reference is no more than 3 [s]. *Assume that the reference and disturbances are step like signals*

Since the plant has a 2 [s] *pure delay, we cannot aim for a settling time less than that if we want a robust closed loop. Say we choose the settling time equal to 3* [s]. *We recall that the settling time is usually defined as the pure delay plus four times the dominant time constant. This implies that the dominant pole should be located to the left of* -4; *this will yield a dominant time constant equal to* 0.25 [s]. *Say we then choose (for the rational part i.e.,* $\bar{G}_o(s)$*)*

$$A_{cl}(s) = s^2 + 8s + 20 \tag{33}$$

Then the corresponding Diophantine equation is

$$(s+1)\underbrace{s}_{L(s)} + 1 \cdot \underbrace{(p_1 s + p_0)}_{P(s)} = s^2 + 8s + 20 \tag{34}$$

Then $C(s)$, *in Figure 7.1 of the book, is given by*

$$C(s) = \frac{7s+20}{s} \tag{35}$$

And the resultant complementary sensitivity is given by:

$$T_o(s) = \frac{7s+20}{s^2+8s+20}e^{-2s} \tag{36}$$