LQG/LTR Control

NDSU ECE 463/663

Lecture #31

Inst: Jake Glower

Please visit Bison Academy for corresponding

lecture notes, homework sets, and solutions

Linear Quadratic Gaussian / Loop Transfer Recovery

Pole Placement:

- Allows you to place the poles wherever you like
- Easy to get a specific response
- Tends to produce large feedback gains

LQR:

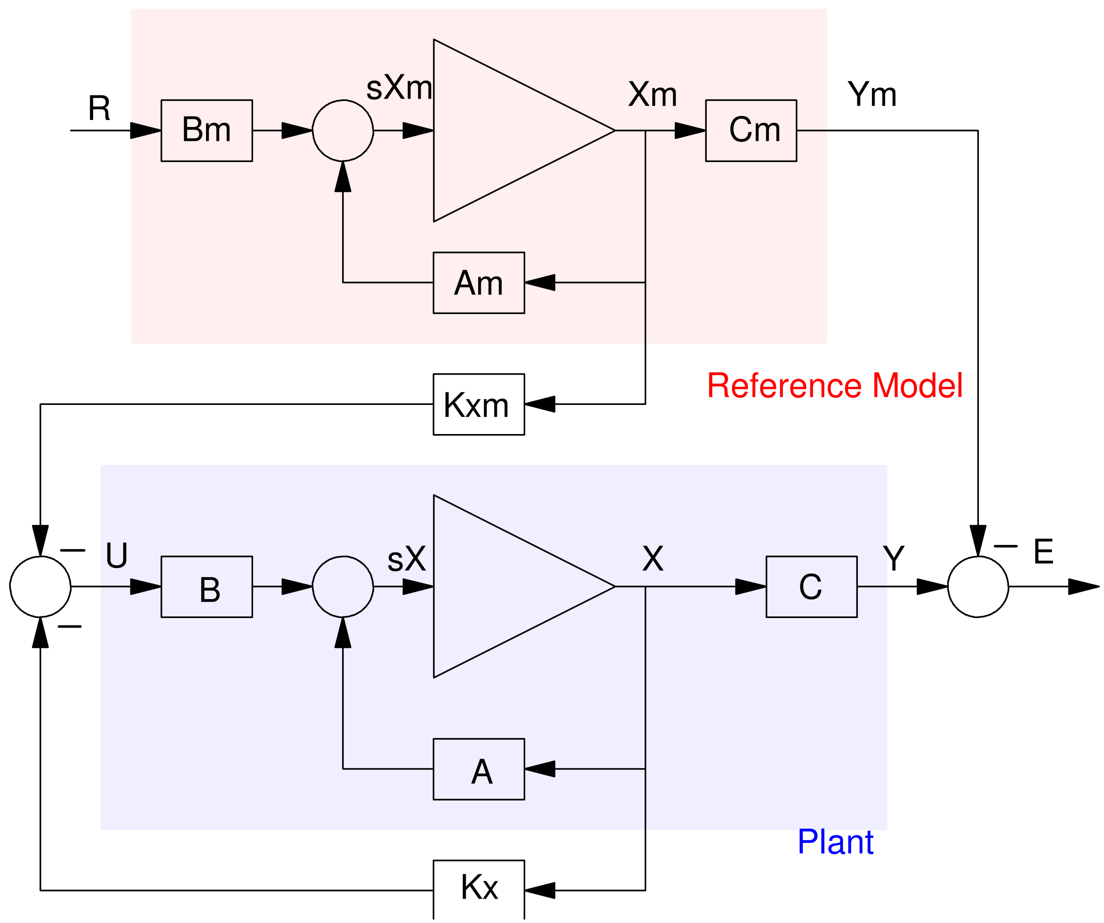
- Smaller feedback gains
- More robust designs
- Harder to get a specific response

LQG/LTR

- Uses LQR techniques
- To get a specific response

LQG/LTR Formulation

- Define a reference mode: how the plant *should* behave:



In state-space, the augmented system is:

$$s\begin{bmatrix} X \\ X_m \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & A_m \end{bmatrix}\begin{bmatrix} X \\ X_m \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} U + \begin{bmatrix} 0 \\ B_m \end{bmatrix} R$$

$$E = Y - Y_m$$

$$E = \begin{bmatrix} C & -C_m \end{bmatrix}\begin{bmatrix} X \\ X_m \end{bmatrix}$$

Design with LQR controller where

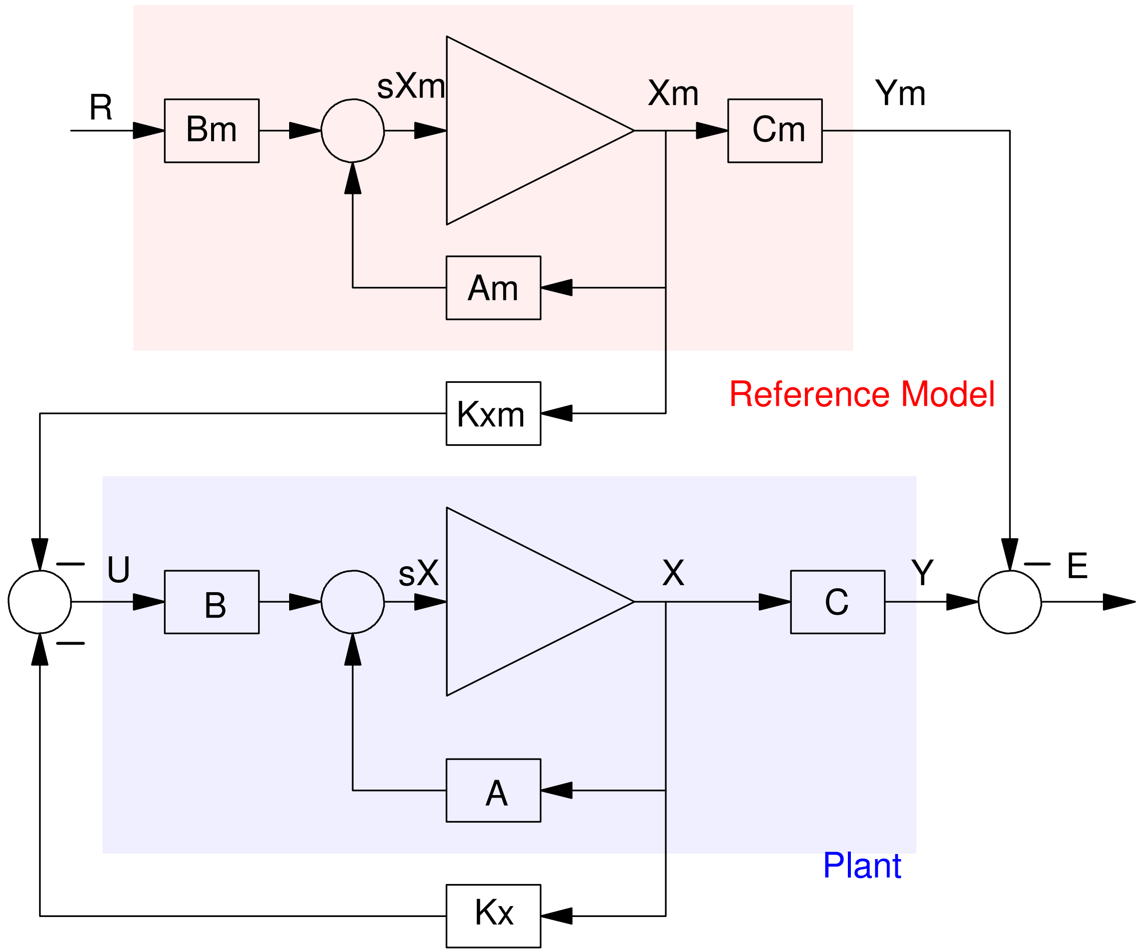
$$Q = \alpha \begin{bmatrix} C & -C_m \end{bmatrix}^T \begin{bmatrix} C & -C_m \end{bmatrix}$$

$$\mathrm{R} = 1$$

Let $\alpha \to \infty$. This forces $X \to X_m$.

The full-state feedback gains will be

$$U = -\begin{bmatrix} K_x & K_{xm} \end{bmatrix} \begin{bmatrix} X \\ X_m \end{bmatrix}$$



Example: Heat Equation

Force a 4-stage RC filer

$$sX = \begin{bmatrix} -2 & 1 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} X + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} U$$

$$Y = \begin{bmatrix} 0 \; 0 \; 0 \; 1 \end{bmatrix} X$$

to behave as

$$Y_m = \left(\frac{10}{s^2+2s+10} \right) R$$

Solution: Define the reference model to be

$$sX_m = \begin{bmatrix} 0 & 1 \\ -10 & -2 \end{bmatrix} X_m + \begin{bmatrix} 0 \\ 10 \end{bmatrix} R$$

$$Y_m = \begin{bmatrix} 1 & 0 \end{bmatrix} X_m$$

Define the augmented system

$$s\begin{bmatrix} X \\ X_m \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & A_m \end{bmatrix} \begin{bmatrix} X \\ X_m \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} U$$

$$Y = E = \begin{bmatrix} C & -C_m \end{bmatrix} \begin{bmatrix} X \\ X_m \end{bmatrix}$$

To force the error to zero, define $Q = C^T C$

```
Q = C6'*C6
R = 1;
```

Case 1: $Q = 10^6 \ C^T C$

```
Q6 = 1e6 * C6'*C6
K6 = lqr( A6,  B6,  Q6, R )
```

8.7795   56.0993 248.6748 685.4467 -246.0652 -188.6547

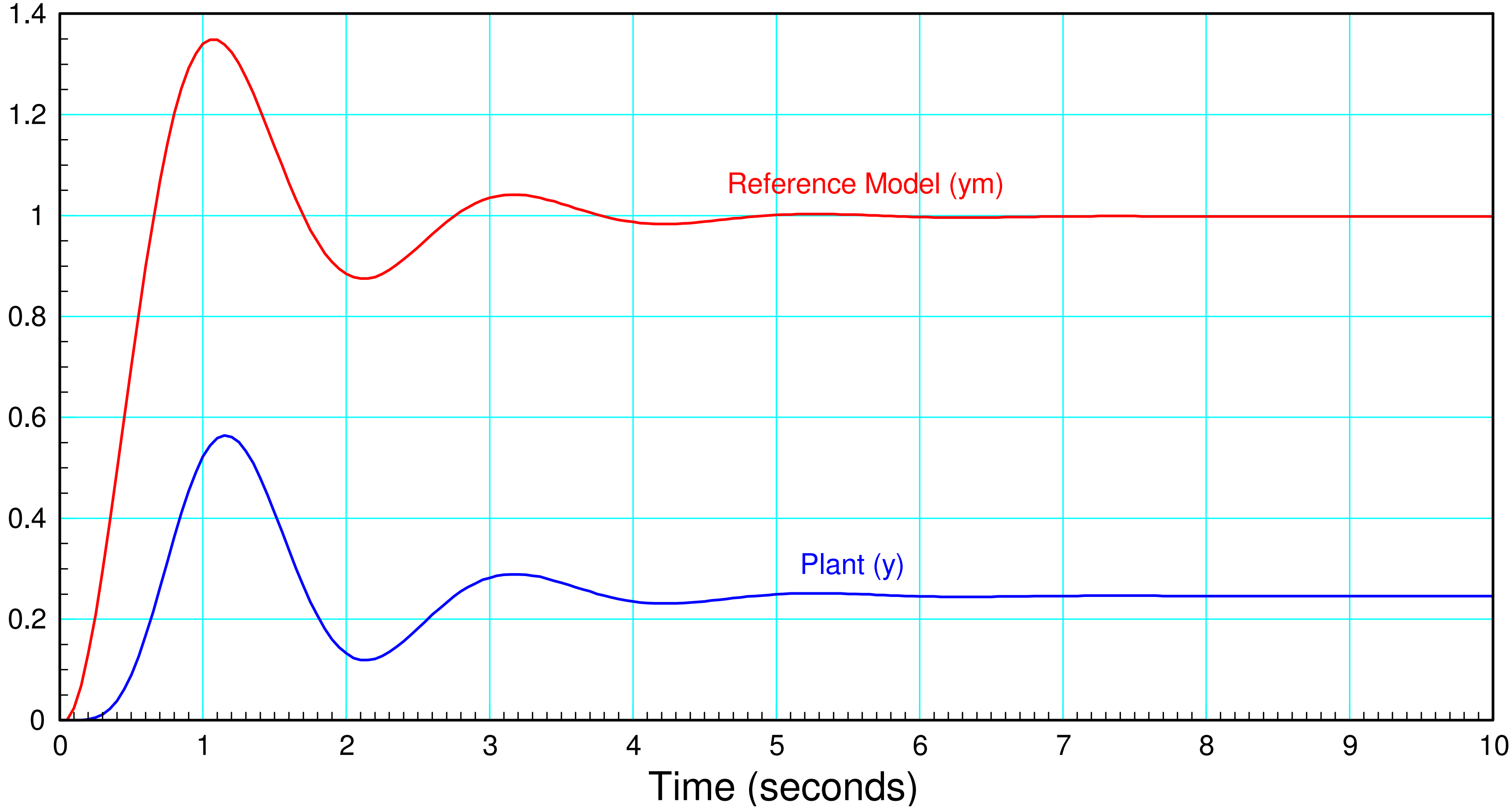
The poles of the closed-loop system are

```
eig(A6-B6*K6)

  - 2.2976458 + 4.8124056i   Dominant Pole
  - 2.2976458 - 4.8124056i
  - 5.5921276 + 1.9727695i
  - 5.5921276 - 1.9727695i
  - 1. + 3.i                Reference Model's Poles
  - 1. - 3.i
```

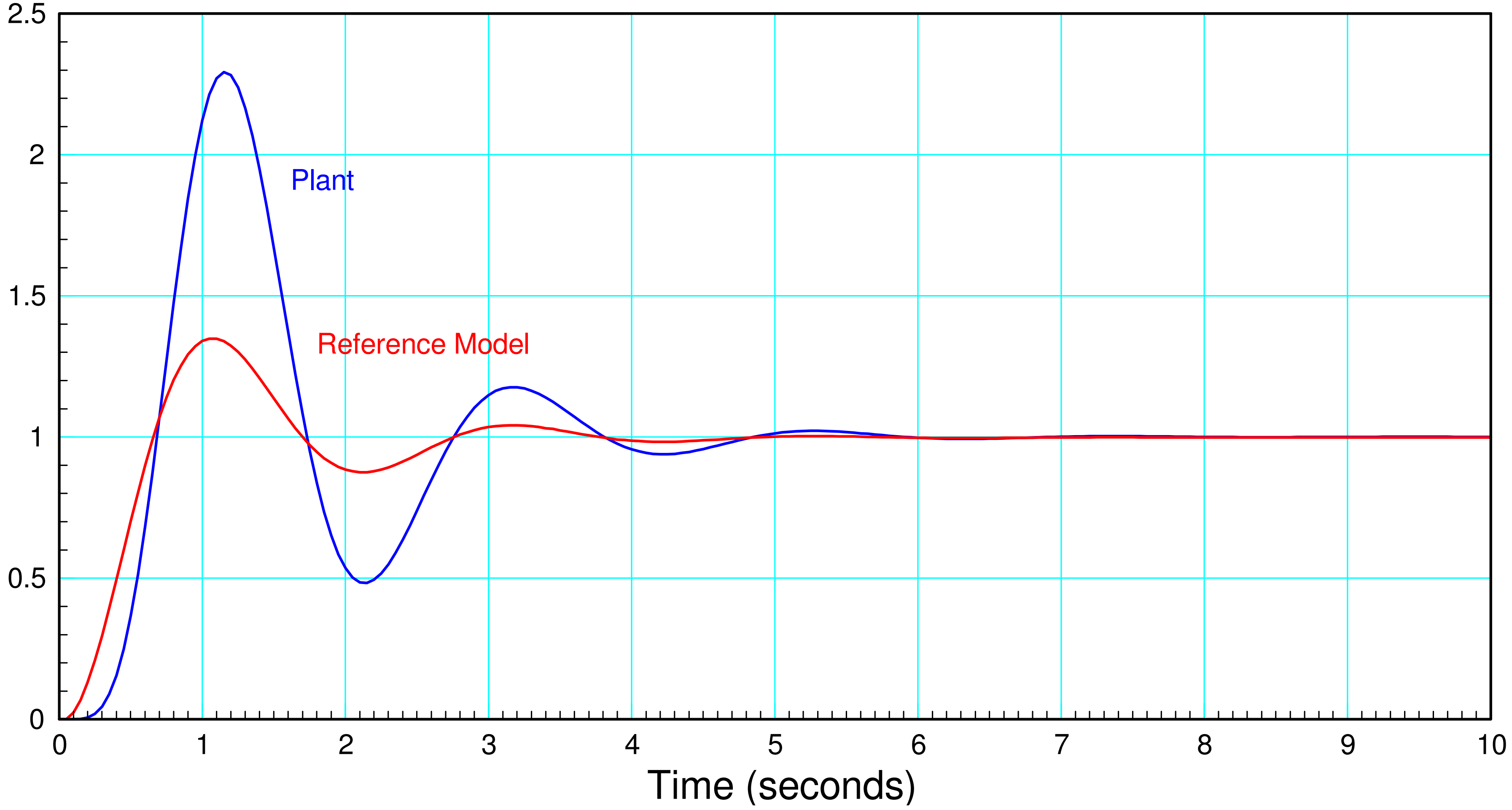
Note

- Two poles are at $-1 \pm j3$ - which is the reference model
- The other poles are a little faster - meaning it will sort of track the reference model. They're not that much faster, however, so tracking will be poor.

Plotting the step response



Adjust the DC gain to 1.000



Note that the response is sort of following the reference model. It's not great, however.

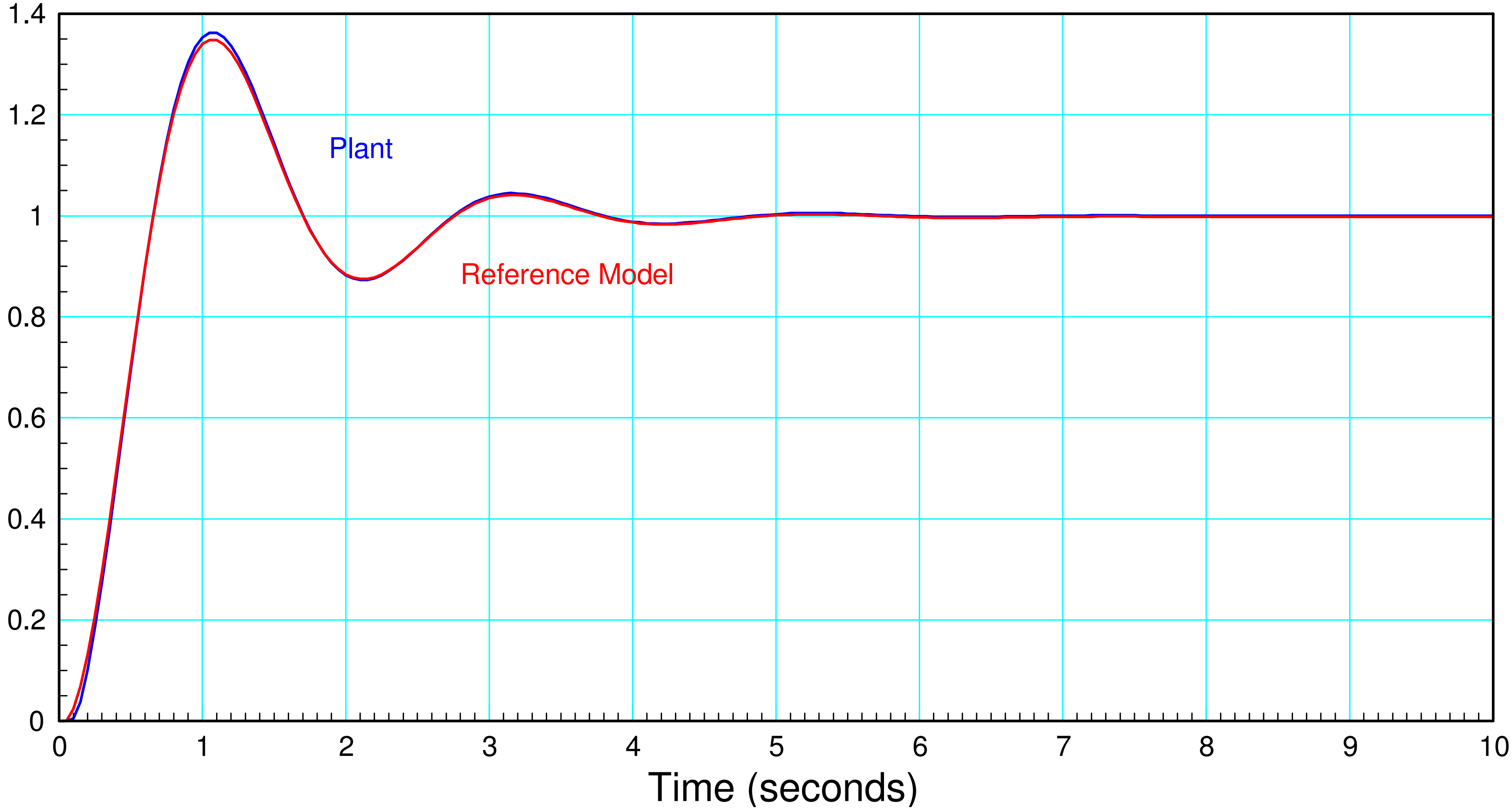
The response is the "optimal" response, however, for the Q and R selected.

To get a better response, increase Q

```
Q6 = 1e12 * C6'*C6;
R6 = 1;
K6 = lqr(A6, B6, Q6, R)

75.8304   3026.789  73170.323   923726.06  -967507.21  -75408.799

eig(A6-B6*K6)

  - 12.13016 + 29.146288i
  - 12.13016 - 29.146288i
  - 29.285062 + 12.072677i
  - 29.285062 - 12.072677i
  - 1. + 3.i
  - 1. - 3.i
```

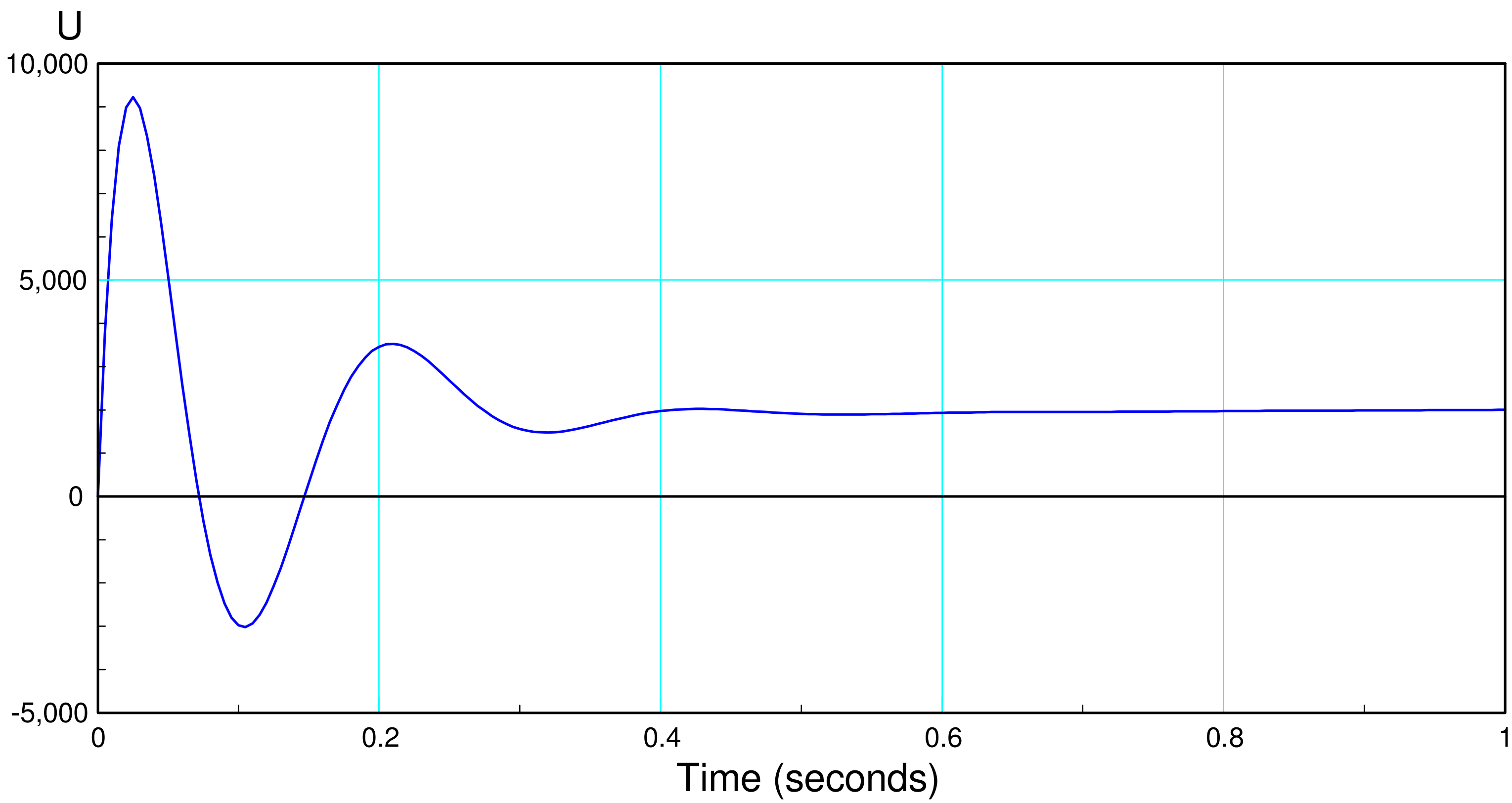
- The feedback gains are much larger. It takes larger gains to force the desired response.
- The closed-loop poles are more reasonable. The pole at $-1 \pm j3$ is the reference model. The other poles are the plant - which are much faster than the reference model (meaning it should be able to track the model)

Step Response (Q = 1e12 C'C), DC gain set to 1.000



Note

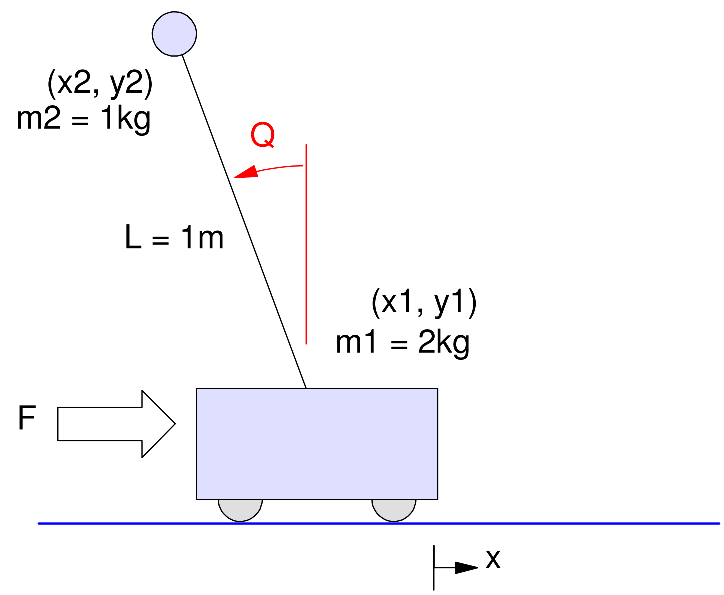
- The plant behaves almost identical to the reference model - even though you are trying to make a heat equation oscillate.
- This comes at a cost: the input is quite large:



Example 2: Cart & Pendulum

Use LQG/LTR Techniques for the cart and pendulum system

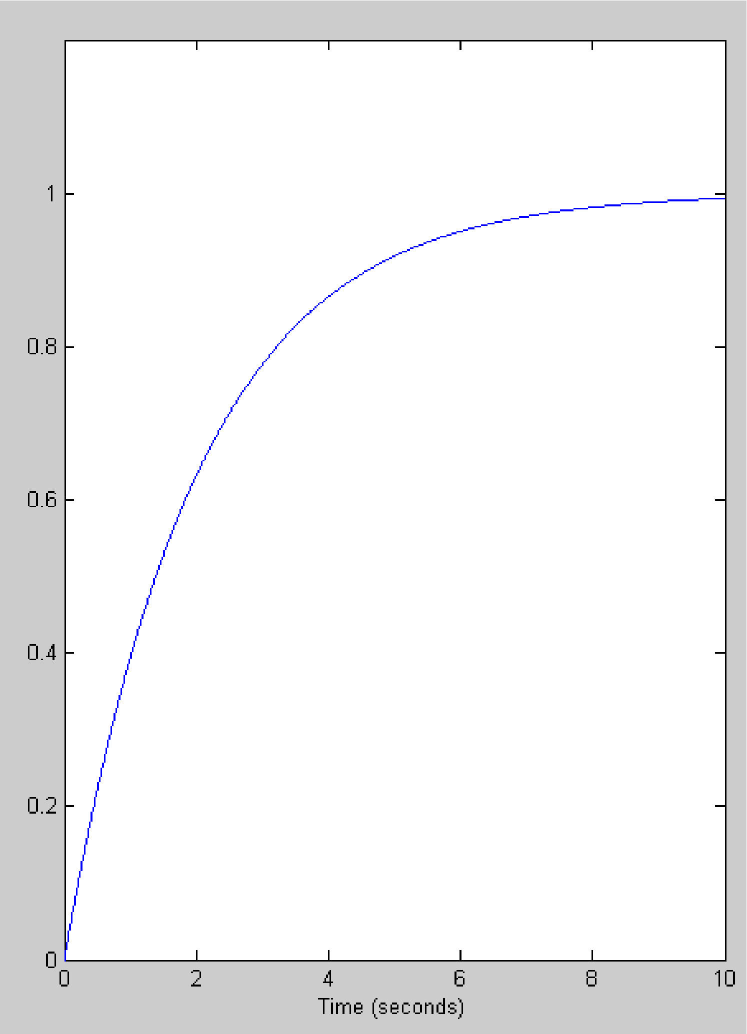
$$s\begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -4.9 & 0 & 0 \\ 0 & 14.7 & 0 & 0 \end{bmatrix}\begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0.5 \\ -0.5 \end{bmatrix}F$$



Step 1: Define the reference model

- 8 second settling time
- No overshoot

```
%Reference Model

Gm = zpk([],[-0.5],1);

DC = evalfr(Gm,0);
Gm = Gm / DC;
X = ss(Gm);
Am = X.a;
Bm = X.b;
Cm = X.c;

[n,m] = size(Am);

X = zeros(4,1);
Xm = zeros(n,1);
```



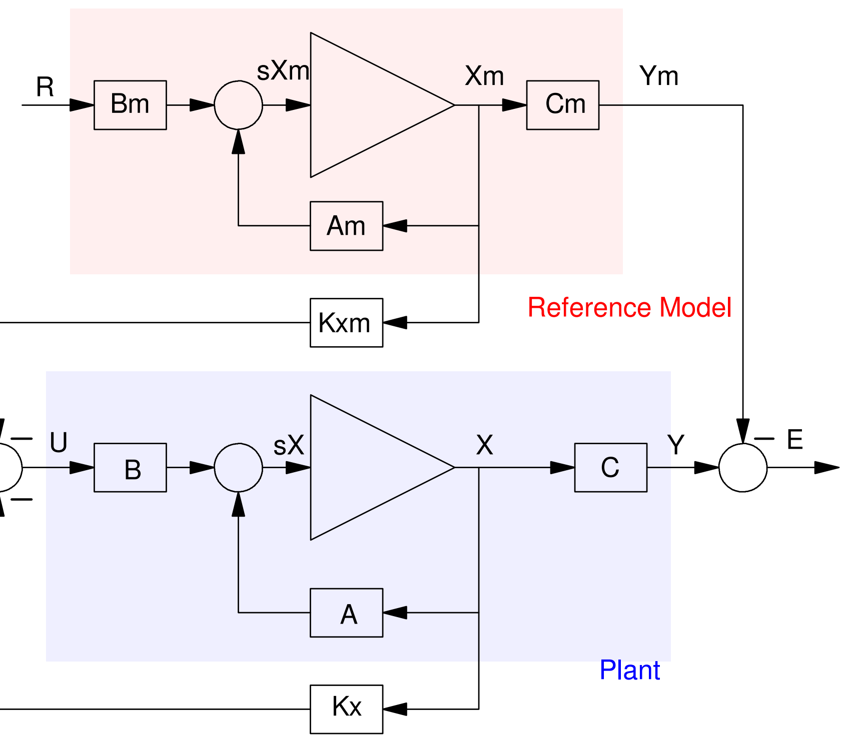
Step 2: Find the feedback gains

Create the augmented system

```
Aa = [A, zeros(4,n) ; zeros(n,4), Am];
Ba = [B; zeros(n,1)];
Ca = [C, -Cm];
```

Find the feedback gains

```
Q = Ca' * Ca;
R = 1;

Ka = lqr(Aa, Ba, Q*1e4, 1);

Kx = Ka(1:4);
Km = Ka(5:4+n);
```



Step 3: Adjust the DC gain

As Q goes to infinity, the DC gain goes to 1.000

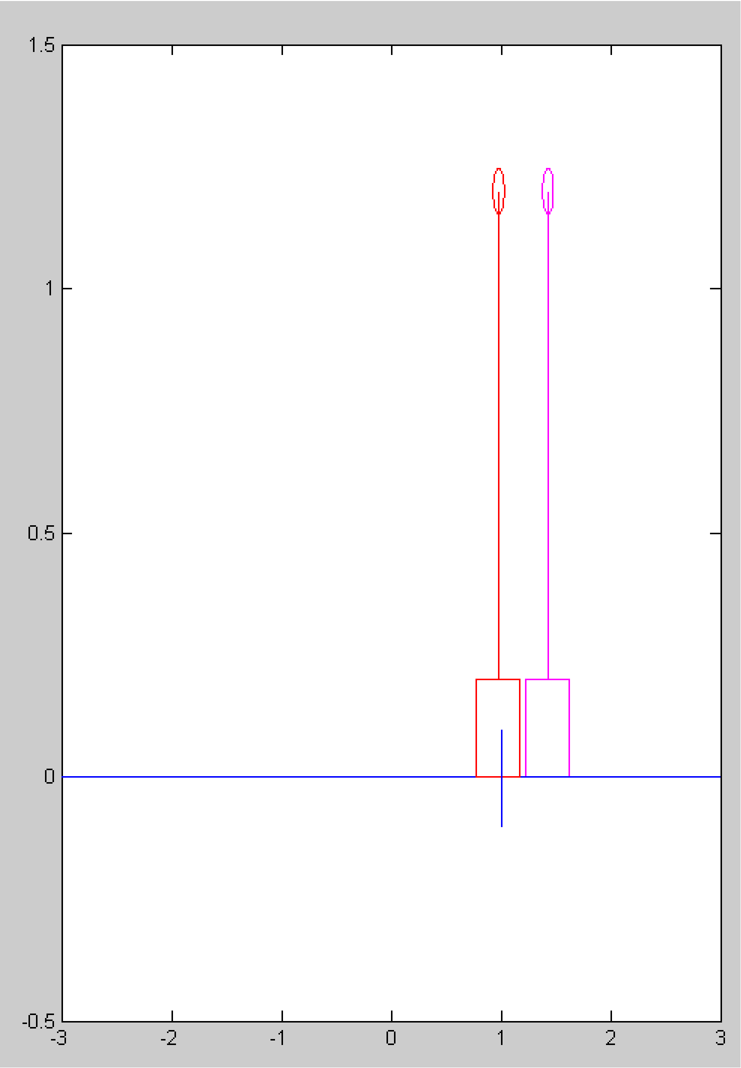
To use a finite Q, add a gain (similar to Kr) to the input

- Pick Kr so that the DC gain is 1.000

```
DC = -[C,0*Cm]*inv(Aa-Ba*Ka)*[0*B;Bm];
Bm = Bm/DC;
```

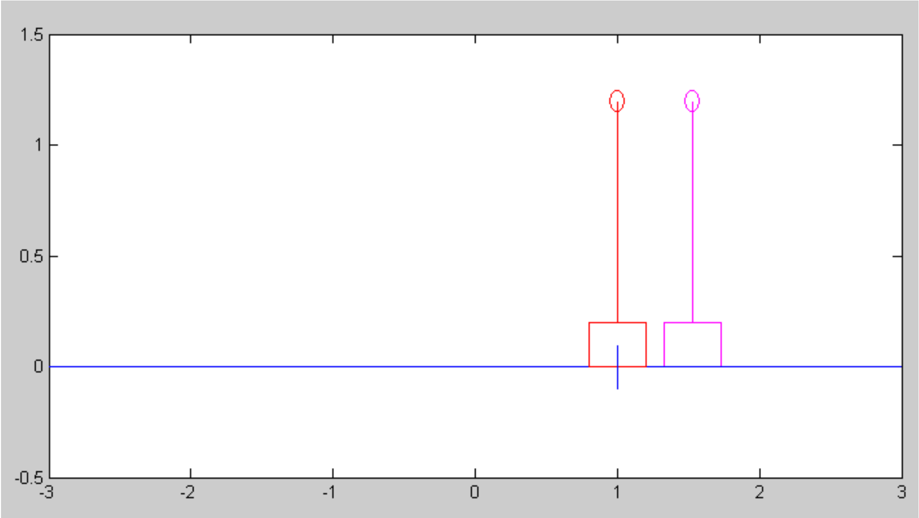
Main Loop

```
while(t < 29.9)
 Ref = sign(sin(0.2*t));
 U = -Km*Xm- Kx*X;

 dX = CartDynamics(X, U);
 dXm = Am*Xm + Bm*Ref;

 X = X + dX * dt;
 Xm = Xm + dXm * dt;

 t = t + dt;
 n = mod(n+1, 5);
 if(n == 0)
    CartDisplay(X, [Cm*Xm;0;0;0], Ref);
 end
 y = [y ; X(1), Cm*Xm, Ref];
end
```



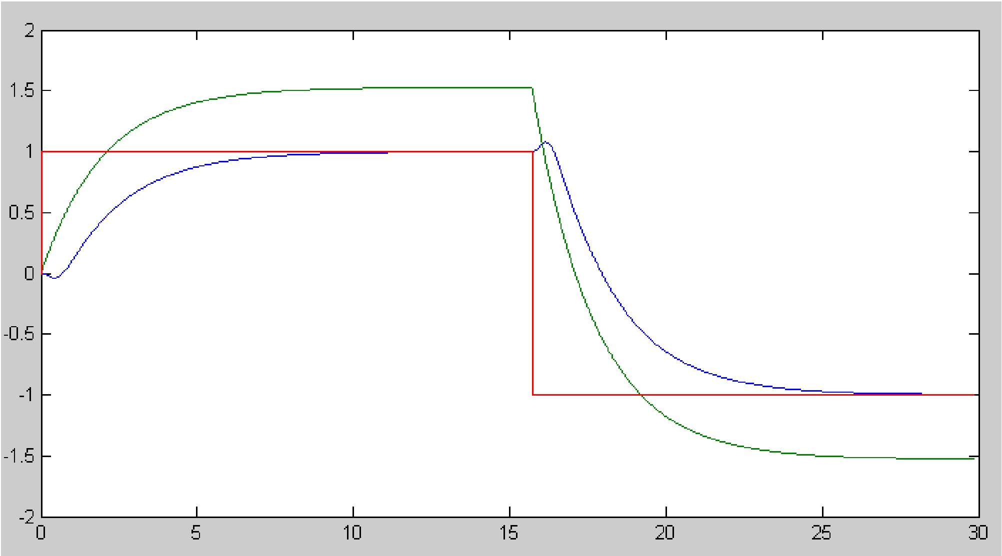
Response:

$$G_m = \left(\frac{0.5}{s+0.5}\right)$$



Note

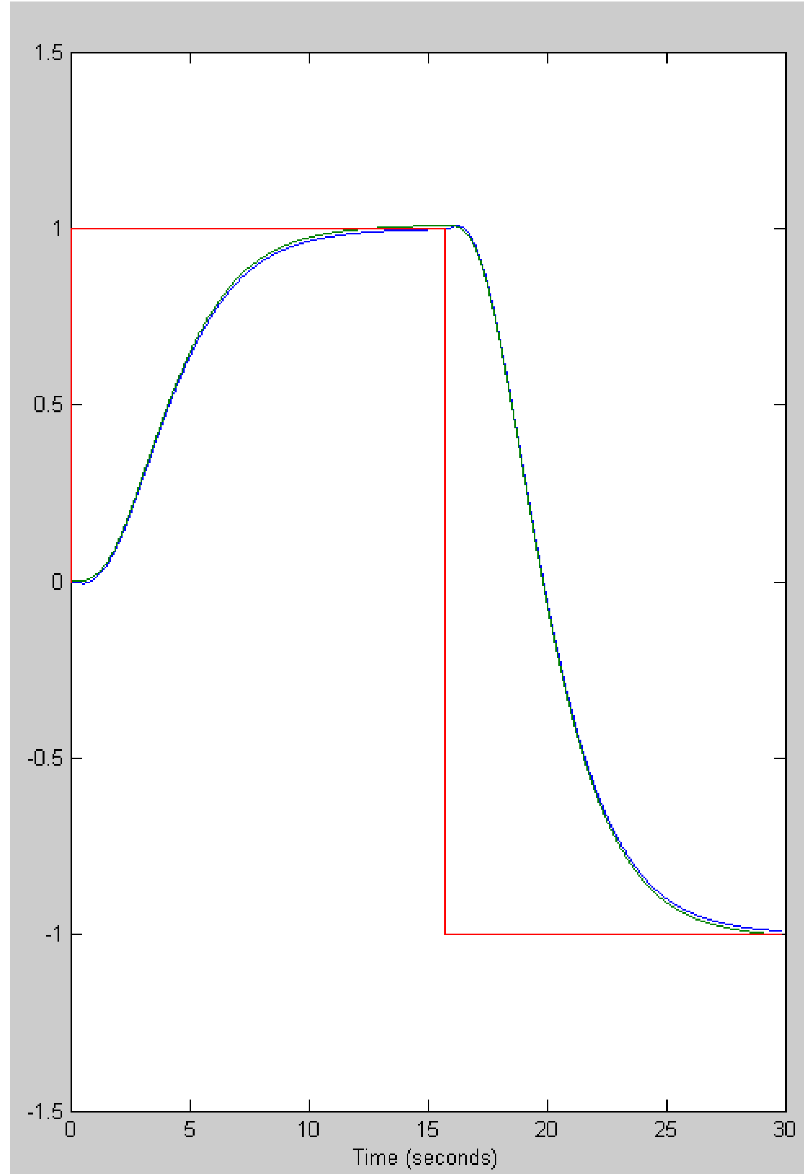
- The cart is a 4th-order system
- The reference model is a 1st-order system
- It's trying, but it's hard to make a 4th order system behave like a 1st order one



Response

$$G_m = \left(\frac{-}{(s+0.5)(s+1)(s+1.2)(s+1.3)} \right)$$

Works much better if you make the reference model 4th order as well

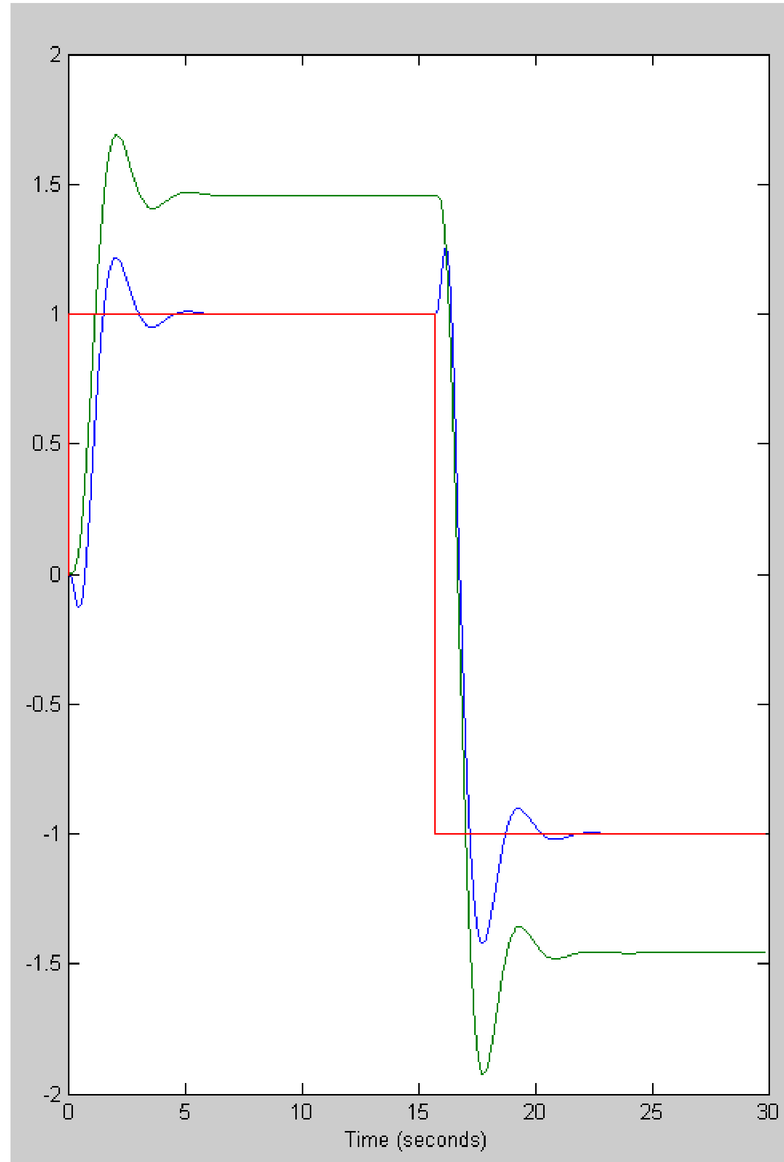


Response

$$G_m = \left(\frac{-}{(s+1+j2)(s+1-j2)(s+4)(s+5)} \right)$$

With LQG/LTR, just change the reference model and you get the desired response

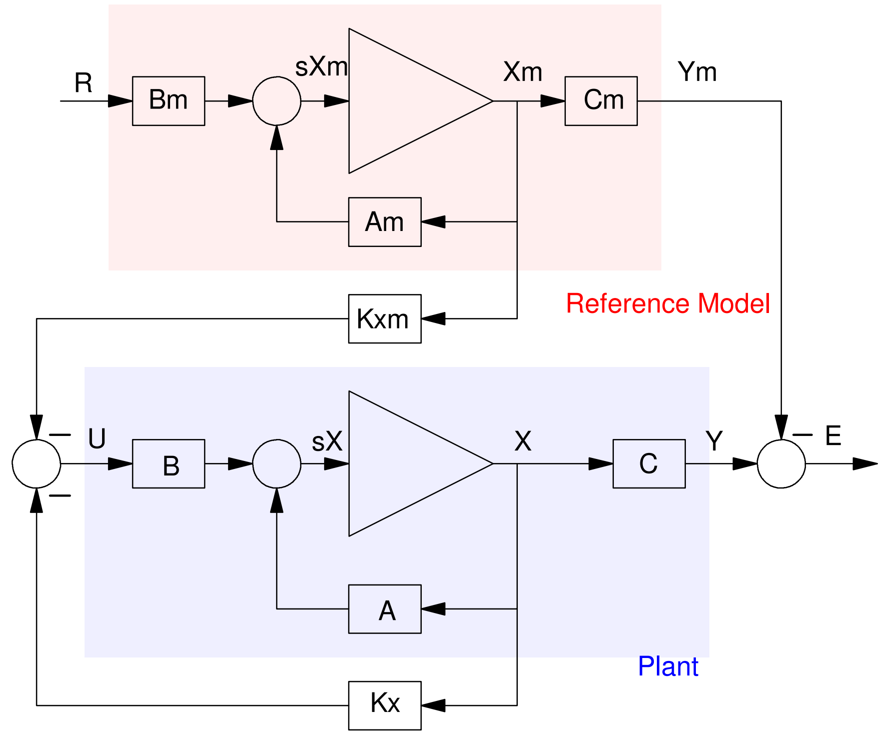
- Much easier to specify the desired response



Summary

LQG/LTR is another way to design feedback controllers

- The reference model defines how the system should behave
- The control law tries to make the plant behave like the reference model



The DC gain isn't 1.000

- Adding a gain (Kr) at the input allows you to make the DC gain 1.000

```
% Cart and Pendulum
% Lecture %31
% LQG/LTR

dt = 0.01;
t = 0;
%Plant
A = [0,0,1,0;0,0,0,1;0,-4.9,0,0;0,14.7,0,0];
B = [0;0;0.5;-0.5];
C = [1,0,0,0];

%Reference Model
Gm = zpk([],[-0.5],1);

DC = evalfr(Gm,0);
Gm = Gm / DC;
X = ss(Gm);
Am = X.a;
Bm = X.b;
Cm = X.c;
[n,m] = size(Am);

X = zeros(4,1);
Xm = zeros(n,1);

A6 = [A, zeros(4,n) ; zeros(n,4), Am];
B6 = [B; zeros(n,1)];
C6 = [C, -Cm];

Q = C6' * C6;
R = 1;

K6 = lqr(A6, B6, Q*1e4, 1);

Kx = K6(1:4);
Km = K6(5:4+n);
DC = -[C,0*Cm]*inv(A6-B6*K6)*[0*B;Bm];
Bm = Bm/DC;

n = 0;
y = [];
while(t < 29.9)
 Ref = sign(sin(0.2*t));
 U = -Km*Xm- Kx*X;

 dX = CartDynamics(X, U);
 dXm = Am*Xm + Bm*Ref;

 X = X + dX * dt;
 Xm = Xm + dXm * dt;

 t = t + dt;
 n = mod(n+1, 5);
 if(n == 0)
    CartDisplay(X, [Cm*Xm;0;0;0], Ref);
 end
 y = [y ; X(1), Cm*Xm, Ref];
end

hold off;
t = [1:length(y)]' * dt;
plot(t,y);
```