Lecture 7

System Control

Consider a simple mechanical system consisting of a damped spring under an external force F(t)

$m\,x'' + \mathrm{bx}' + \mathrm{kx} = F\,(\mathrm{t})$

m= 1kg (mass)

b = 10N s/m (friction)

k = 20N/m (spring constant)

Giving:

$x'' + 10x' + 20x = F\,(t)$

The equation above is a differential equation for x(t) in the time domain. The control goal is to supply a time dependent external force F(t) in order to stear the system x(t) to a desirable goal state.

In the following discussion we target the goal state

x(t) = 1

PID Feedback Control

A common control strategy is to generate a control signal that is a function of the error between the current and the goal state. A general form of the PID control is

$$F\,(t) = k_P e(t) + k_I \int e(\tau\ )\ d\tau + k_d \frac{d}{\mathrm{dt}} e(t)$$

with control gains

Kp: control directly proportional to the error

KI: proportional to the accumulated history of the error

Kd: proportional to the derivative, anticipated direction of the error

The engineering task is to choose a set of approrpiate control gains that lead to a control with desirable features, such as stability, time to reach the goal state, minimal overcorrection etc.

From the previous week we studies the following rules how functions of time translate into the Laplace s-domain

Impulse, delta function

$\delta(t) \rightarrow 1$

Constant or step function:

$$\theta(t) \to \frac{1}{s}$$

Linear time:

$$t \to \frac{1}{s^2}$$

Integral:

$$\int x(t) \to \frac{x(s)}{s}$$

Function:

$$x(t) \to x(s)$$

First derivative

$x(t) \to s\,x(s)$ + initial conditions

Second derivative

$\frac{d^2}{\mathrm{dt}^2} x(t) \to s^2\,x(s)$ + intial conditions

Translated into the s domain, the equation of the system becomes

$$(s^2 + 10s + 20)x(s) = f(s)$$

```
syms x(t) t s
laplace(diff(diff(x,t),t)+diff(x,t)+x(t),t,s)
```

```
ans =
```

$$s\,\mathrm{laplace}(x(t), t, s) - x(0) - s\,x(0) + s^2\,\mathrm{laplace}(x(t), t, s) - \left(\left(\frac{\partial}{\partial t}\,x(t)\right)\Big|_{t=0}\right) + \mathrm{laplace}(x(t), t, s)$$

Step Response

What force should one apply for the system to reach a steady state of x(t)=1?

If we require that after the goal is reached that the first and second derivatives of x(t) are zero, the system equation reads

$$20\,x(t) = F(t)$$

thus

$$F(t) = 20 \quad \text{for} \;\; x(t) = 1$$

Assuming we switch on the external force at t=0, the system becomes

$$x'' + 10x' + 20 = 20\,\theta(t)$$

or in the s-domain

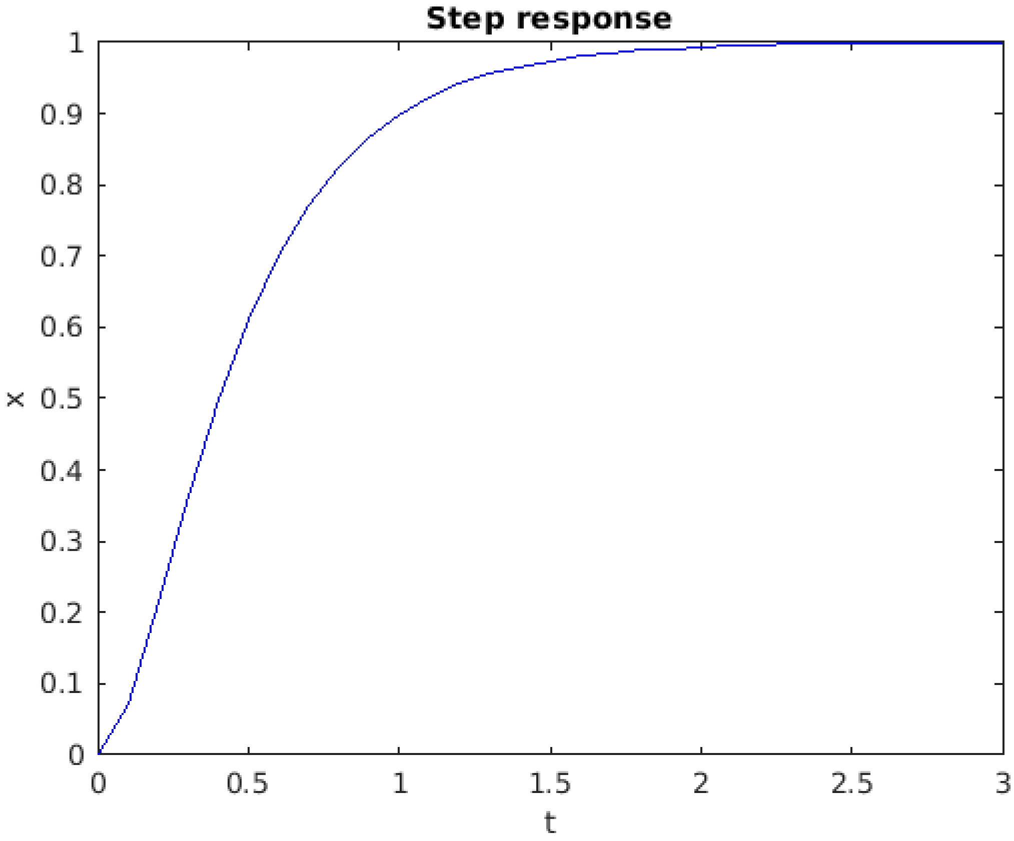
$$(s^2 + 10s + 20)x(s) = \frac{20}{s}$$

Taking the inverse laplace transfrom and plotting x(t) shows that the system reaches the goal x(t)=1 after around 3sec.

```
syms s t
f(t)=ilaplace(20/(s*(s^2+10*s+20)))
```

f(t) = $1 - e^{-5\,t}\,(\cosh(\sqrt{5}\;t) + \sqrt{5}\;\sinh(\sqrt{5}\;t))$

```
plot(0:.1:3,f(0:.1:3),'b-')
title("Step response")
xlabel("t")
ylabel("x")
```



Proportional Control

In the previous example the external control force did not actually take the state of the system into account. This worked well because we had a good understanding about the magnitude of the required force to reach the goal state and there was no noise or other disturbance in the system. However in many other situations this strategy is not sufficient. It is like stepping on the gas pedal or turning the stearing wheel without looking at the road ahead.

In the following we consider a control pthat is proportional to the error between the current state of the system and the goal. In this case we have

$$x'' + 10x' + 20x = k_p(1 - x)$$

Assuming a positive Kp=300, the control force is positive as long x(t) is below 1 and becomes negative in case x(t) overshoots and becomes larger than 1.

$$x'' + 10x' + (\mathrm{kp} + 20)x = \mathrm{kp}$$

or in the s-domain

$$(s^2 + 10s + (\mathrm{kp} + 20))x(s) = \frac{\mathrm{kp}}{s}$$

```
syms s t
kp=300
```

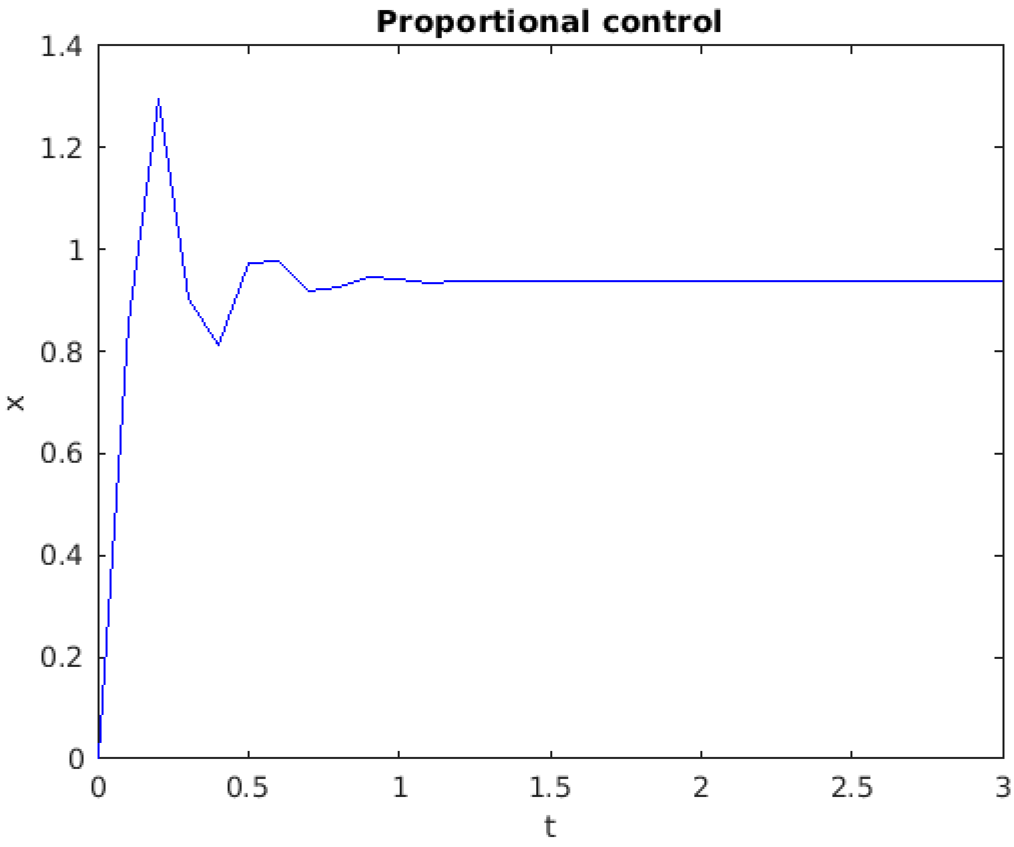
```
kp = 300
```

```
f(t)=ilaplace(kp/(s*(s^2+10*s+kp+20)))
```

```
f(t) =
```

$$\frac{15}{16} - \frac{15\,e^{-5\,t}\left(\cos(\sqrt{295}\;t) + \frac{\sqrt{295}\;\sin(\sqrt{295}\;t)}{59}\right)}{16}$$

```
plot(0:.1:3,f(0:.1:3),'b-')
title("Proportional control")
xlabel("t")
ylabel("x")
```



We can see that the system first overshoots and then settles below the desired goal state.

General PID Control

Lets consider the general case of a control system with proportional, integral and derivative control.

The system plus control equation is

$$x'' + 10x' + 20x = k_p(\theta(t) - x) + k_I \int (\theta(t) - x)\mathrm{dx} + k_d \frac{d}{\mathrm{dt}}(\theta(t) - x)$$

The step function 1 in the last derivative gives a delta function that transforms to 1, while under the integral it integrates to t and transforms to 1/s^2

$$(s^2 + 10s + 20)x(s) = \frac{k_p}{s} - k_p x(s) + \frac{k_I}{s^2} - \frac{k_I}{s} x(s) + k_d - \mathrm{sk}_d x(s)$$

Solving for x(s) gives

$$x(s) = \frac{s^2 k_d + \mathrm{sk}_p + k_I}{s\left(s^3 + (10 + k_d)s^2 + (20 + k_p)s + k_I\right)}$$

Below we carry out the symbolic calculation within Matlab. First we set up definitions for variables, the system state function x(t), the error e(t) and its derivatives and integral

```
syms x(t) t s d ki kp kd Y

x1(t) = diff(x(t),t);
x2(t) = diff(x1(t),t);
e(t) = heaviside(t)-x(t);
e1(t) = diff(e(t),t);
ei(t)=int(e(d),0,t);
```

The laplace transform of the system

```
lhs=laplace(x2(t)+10*x1(t)+20*x(t),t,s);
lhs0 =  subs(lhs,[x(0) x1(0) x2(0) laplace(x(t),t,s)],[0 0 0 Y])
```

lhs0 = $Y\, s^2 + 10\, Y\, s + 20\, Y$

The laplace transform of the control

```
rhs=laplace(kp*e(t)+ki*ei(t)+kd*e1(t),t,s);
rhs0 = subs(rhs,[x(0) x1(0) laplace(x(t),t,s)],[0 0 Y])
```

rhs0 =

$$-\mathrm{kd}\,(Y\, s - 1) - \mathrm{kp}\,\left(Y - \frac{1}{s}\right) - \mathrm{ki}\,\left(\frac{Y}{s} - \frac{1}{s^2}\right)$$

Solving for the system state Y=x(s) under the influence of the control

```
eq = lhs0==rhs0;
eqY = solve(eq,Y)
```

```
eqY =
```

$$\frac{\mathrm{kd}+\frac{\mathrm{ki}}{s^2}+\frac{\mathrm{kp}}{s}}{\mathrm{kp}+10\,s+\mathrm{kd}\,s+\frac{\mathrm{ki}}{s}+s^2+20}$$

Substituing values for the control gains

```
eqY0 = subs(eqY,[kp ki kd],[350 300 50])
```

```
eqY0 =
```

$$\frac{\frac{350}{s}+\frac{300}{s^2}+50}{60\,s+\frac{300}{s}+s^2+370}$$

Transforming the system response back into the time domain

```
syms s t
f(t)=ilaplace(eqY0)
```

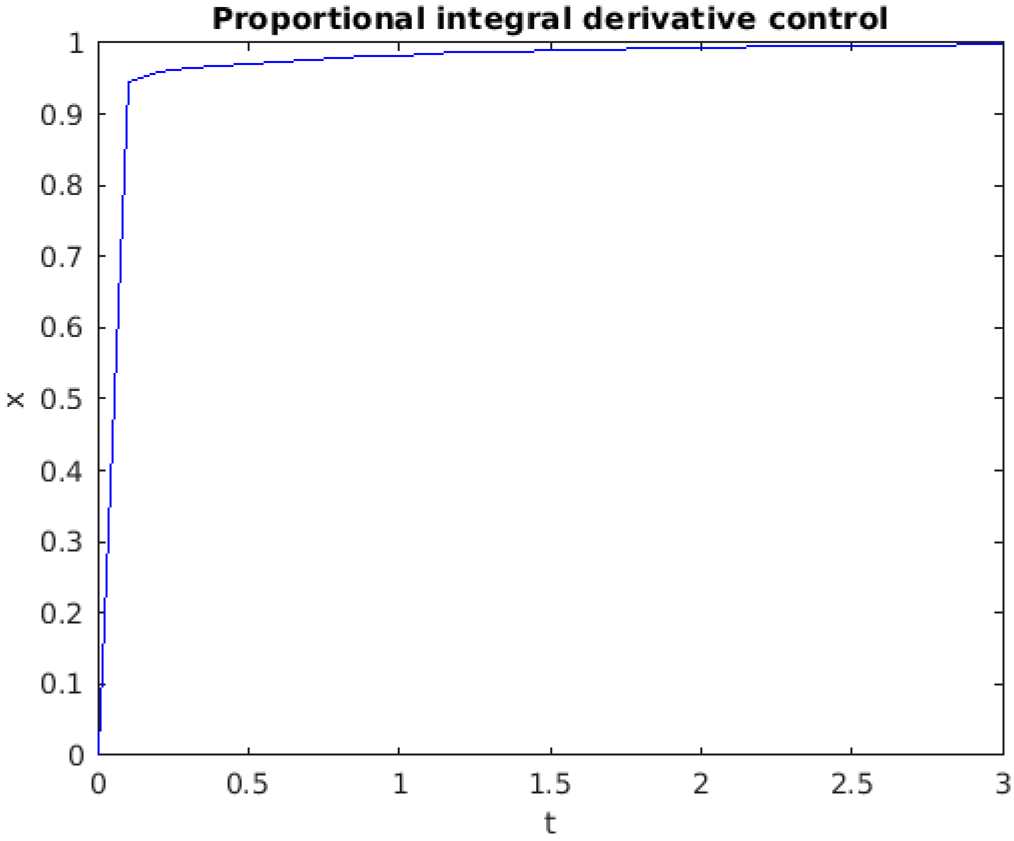
```
f(t) =
```

$$1-20\,\Bigg(\sum_{k=1}^{3}\frac{\mathrm{e}^{t\,\sigma_1}}{3\,{\sigma_1}^2+120\,\sigma_1+370}\Bigg)-10\,\Bigg(\sum_{k=1}^{3}\frac{\mathrm{e}^{\sigma_1\,t}\,\sigma_1}{120\,\sigma_1+3\,{\sigma_1}^2+370}\Bigg)-\Bigg(\sum_{k=1}^{3}\frac{\mathrm{e}^{t\,\sigma_1}\,{\sigma_1}^2}{3\,{\sigma_1}^2+120\,\sigma_1+370}$$

where

$$\sigma_1=\mathrm{root}\big({s_5}^3+60\,{s_5}^2+370\,s_5+300,s_5,k\big)$$

```
plot(0:.1:3,f(0:.1:3),'b-')
title("Proportional integral derivative control")
xlabel("t")
ylabel("x")
```



Automatic Controller Tuning

Matlab provides functionality to automatically tune the control gains according to design objectives. To invoce this function one first needs to specify the "plant" function, the left hand side of the system in s-space (without the control terms):

```
s = tf('s');
P = 1/(s^2+10*s+20)
```

```
P =

         1
  ---------------
  s^2 + 10 s + 20

Continuous-time transfer function.
```

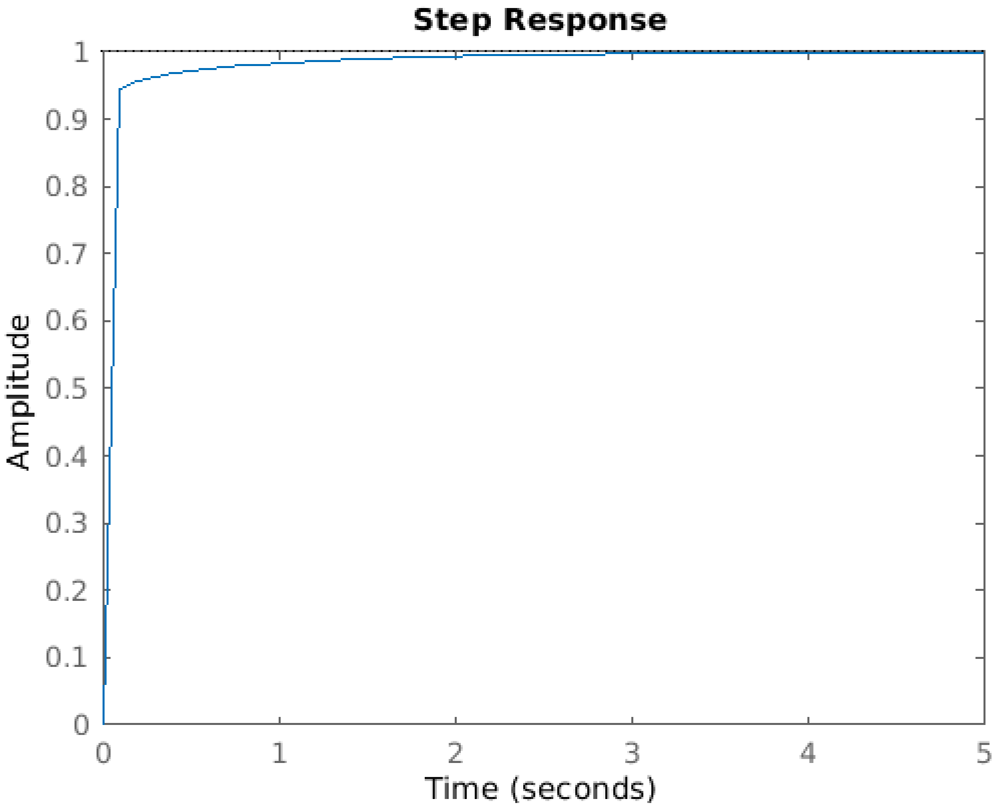
```
C= pid(350,300,50)
```

```
C =

             1
  Kp + Ki * --- + Kd * s
             s

  with Kp = 350, Ki = 300, Kd = 50

Continuous-time PID controller in parallel form.
```

```
T = feedback(P*C,1)
```

```
T =
 
     50 s^2 + 350 s + 300
  --------------------------
  s^3 + 60 s^2 + 370 s + 300
 
Continuous-time transfer function.
```

```
step(T,0:.1:5)
```



```
ans = 1
```

Now start the interactive pidTuner

```
%pidTuner(P)
```