7b. PID Controller Design

7b.1 Ziegler-Nichols Oscillation Method

In practice, the coefficients of a control system should be set so that the closed –loop system has an optimum response. Depending on the controller type, the critical gain, the integral time constant, and the derivative time constant can be determined based on experimental and computational methods. One of these method is the Ziegler-Nichols method.

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \qquad\qquad G_c(s) = K_p\left(1 + \frac{1}{T_i s} + T_d s\right)$$

Ziegler-Nichols Design:

1.Use step input for the reference.

2.Start controlling the system with a constant gain, K.

3.Increase K so that the closed-loop response has a continuous oscillation response.

4. In case of the continous oscillation response, choose the control gain and measure the period of the response. Then , design the control system.

$$\text{P control}: K_p = 0.5K_c$$

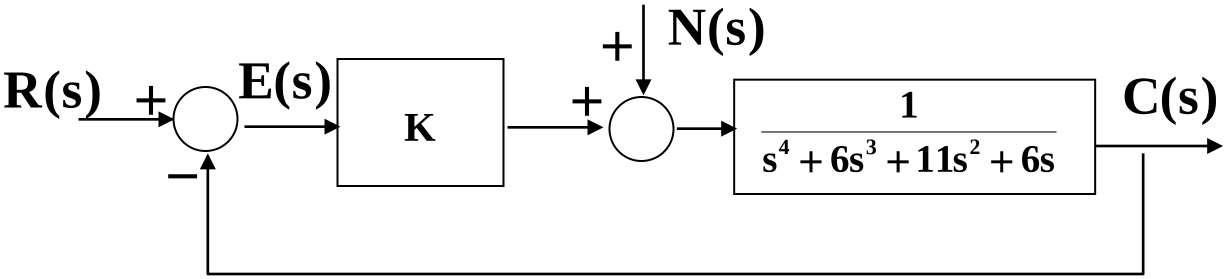
$$\text{PI control}: K_p = 0.45K_c, T_i = 0.83T_c$$

$$\text{PID control}: K_p = 0.6K_c, T_i = 0.5T_c, T_d = 0.125T_c$$

K_c: Critical gain.

T_c: Oscillation period.

Example 7b.1: **Consider Example 7.3. Design the control system (P, PI, PID.) by the Ziegler-Nichols method.**



$$H(s)=\frac{K}{s(s^3+6s^2+11s+6)+K}$$

D(s)= $s^4+6s^3+11s^2+6s+K=0$

Critical gain: $K_c=10$ **(Determine by the Routh-Hurwitz method)**

$$s^4+6s^3+11s^2+6s+10=0$$

MATLAB: >> a=[1,6,11,6,10];roots(a)

$$s_{1,2}=-3\pm 1i, s_{3,4}=\pm 1i$$

$\omega_c T_c = 2\pi$ $\quad$ $\omega_c = 1\ \mathrm{rad/s}$ $\quad$ $T_c = 6.2832\ \mathrm{s}$

$$G(s) = \frac{1}{s(s^3 + 6s^2 + 11s + 6)} \qquad K_c = 10 \qquad T_c = 6.2832\ s$$

Pcontrol: $K_P = (0.5)(10) = 5$

$$G_c(s) = 5$$

$$H(s) = \frac{5}{s(s^3 + 6s^2 + 11s + 6) + 5}$$

PI control: $K_P = (0.45)(10) = 4.5$ $\quad T_i = (0.83)(6.2832) = 5.2150$

$$G_c(s) = 4.5\left(1 + \frac{1}{5.2150s}\right)$$

$$H(s) = \frac{4.5s + 0.863}{s^5 + 6s^4 + 11s^3 + 6s^2 + 4.5s + 0.863}$$

PID control: $K_P = (0.6)(10) = 6$ $\quad T_i = (0.5)(6.2832) = 3.1416$

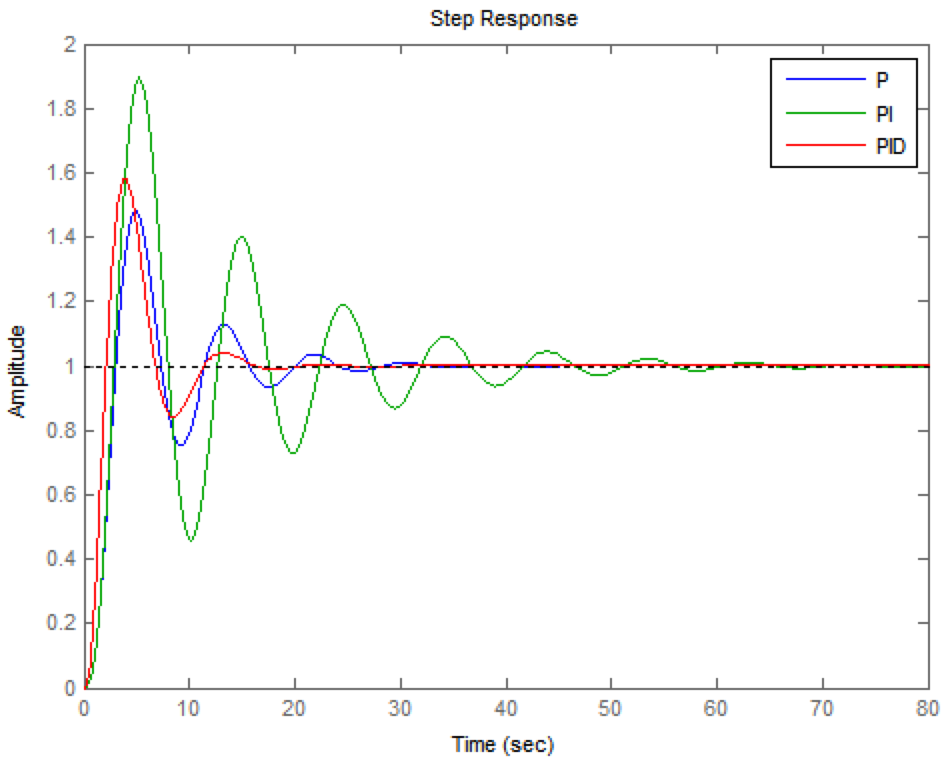
$$T_d = (0.125)(6.2832) = 0.7854$$

$$G_c(s) = 6\left(1 + \frac{1}{3.1416s} + 0.7854s\right)$$

$$H(s) = \frac{4.712s^2 + 6s + 1.91}{s^5 + 6s^4 + 11s^3 + 10.712s^2 + 6s + 1.91}$$

Study and compare the closed-loop responses of P, PI and PID control.

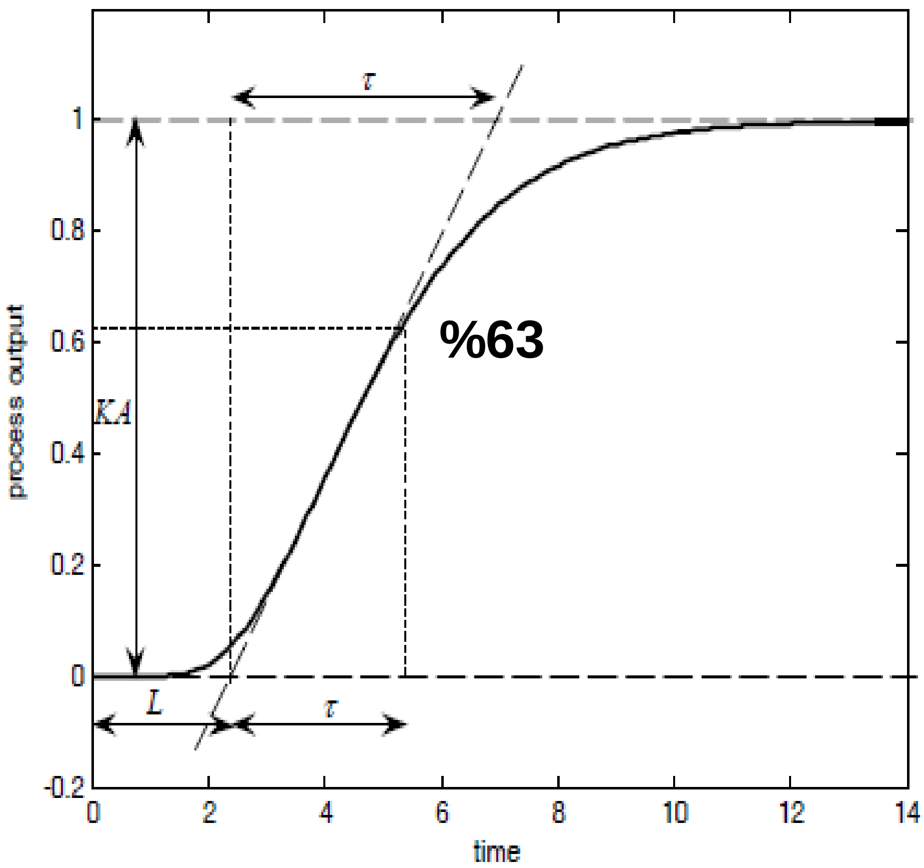
```
clc;clear
k=5;
%P control
nh1=k;dh1=[1 6 11 6 k];
h1=tf(nh1,dh1);step(h1);hold on
%PI control
nh2=[4.5 0.863];dh2=[1 6 11 6 4.5 0.863];
h2=tf(nh2,dh2);step(h2)
%PID control
nh3=[4.712 6 1.91];dh3=[1 6 11 10.712 6 1.91];
h3=tf(nh3,dh3);step(h3)
```



7b.2 Tangent Method

A response of system plant can be obtained with a open loop response, and using following procedure; Obtain the open loop step response. Draw a tangent at the inflection point.

$$\mathbf{K} = \frac{\text{Steady-state value of output}(y_{ss})}{\text{Amplitude of input }(r(t))} \qquad \mathbf{K} = \frac{\mathbf{y}_\infty - \mathbf{y}_0}{\mathbf{u}_\infty - \mathbf{u}_0} \qquad a = \frac{K * L}{\tau}$$



$$G_c(s) = K_p\left(1 + \frac{1}{T_i s} + T_d s\right)$$

Several PID tuning methods have been proposed. Some of them are given in below

- Ziegler–Nichols method
- Chien–Hrones–Reswick method
- Cohen–Coon method
- Refined Ziegler–Nichols method
- The Wang–Juang–Chan method
- Optimum method

Chien-Hrones-Reswick (CHR) method

CHR method emphasizes the set-point regulation or disturbance rejection. CHR method uses the parameters τ, L and a from the plant response.

Table 1. Set-point regulation

Controller type	With 0% overshoot			With 20% overshoot		
	Kp	Ti	Td	Kp	Ti	Td
P	0.3/a	-	-	0.7/a	-	-
PI	0.35/a	$1.2*\tau$	-	0.6/a	τ	-
PID	0.6/a	τ	0.5*L	0.95/a	$1.4*\tau$	0.47*L

Table 2. Disturbance rejection

Controller type	With 0% overshoot			With 20% overshoot		
	Kp	Ti	Td	Kp	Ti	Td
P	0.3/a	-	-	0.7/a	-	-
PI	0.6/a	$4*\tau$	-	0.7/a	$2.3*\tau$	-
PID	0.95/a	$2.4*\tau$	0.42*L	1.2/a	$2*\tau$	0.42*L

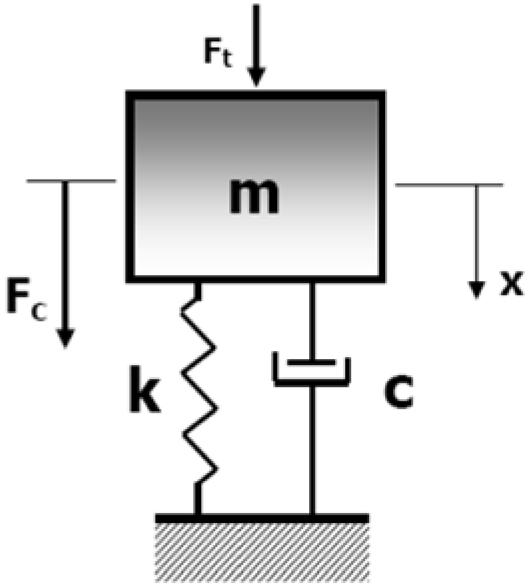
Ziegler-Nichols method

Z-N method uses the parameters τ, L and a from the plant response.

Controller type	Z-N method		
	Kp	Ti	Td
P	1/a	-	-
PI	0.9/a	3*L	-
PID	1.2/a	2*L	0.5*L

Example 7b.2:

Design the PID control system with the tangent methods.



m=2
c=2
k=100

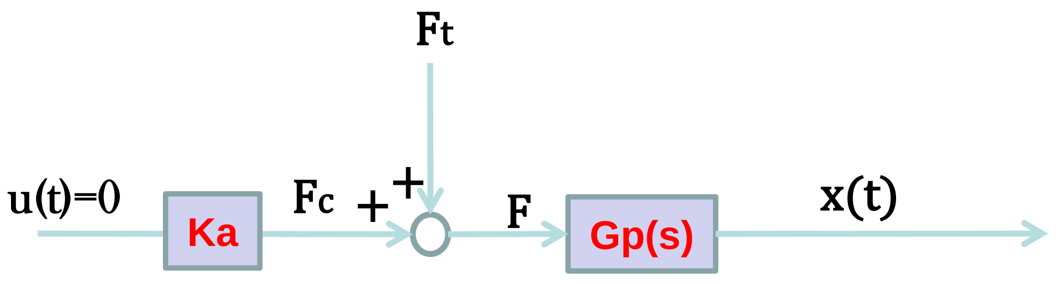
F_c=Control force
F_t= Disturbance force

$$m\ddot{x}(t)+c\dot{x}(t)+kx(t)=F(t) \qquad F(t)=F_c+F_t$$

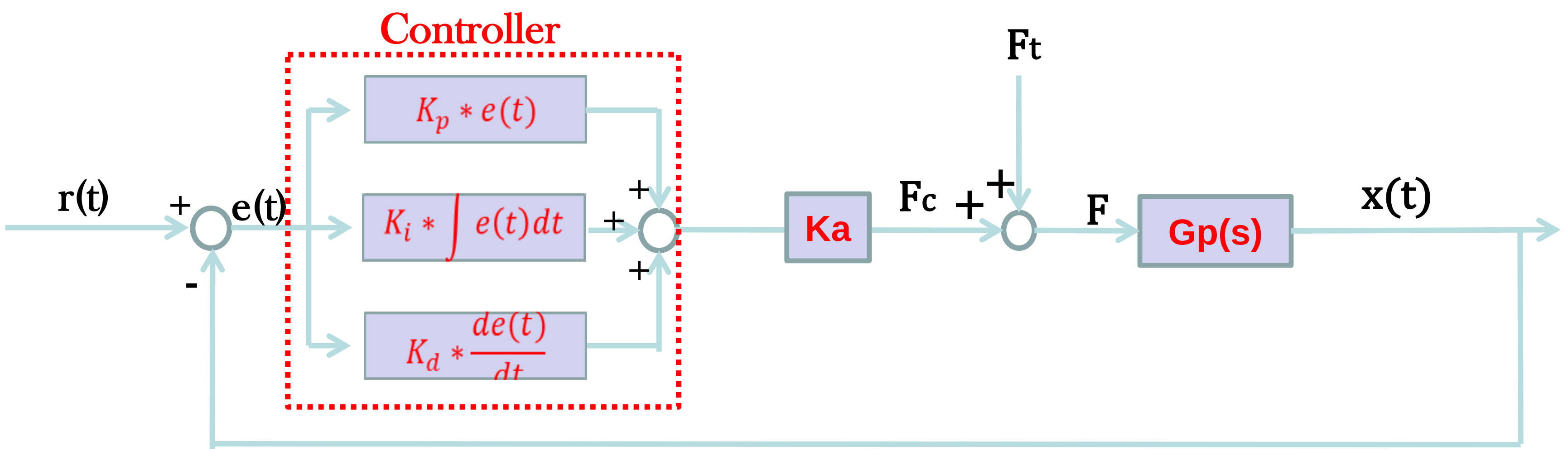
$$G_p(s)=\frac{1}{ms^2+cs+k}=\frac{1}{2s^2+2s+100}$$

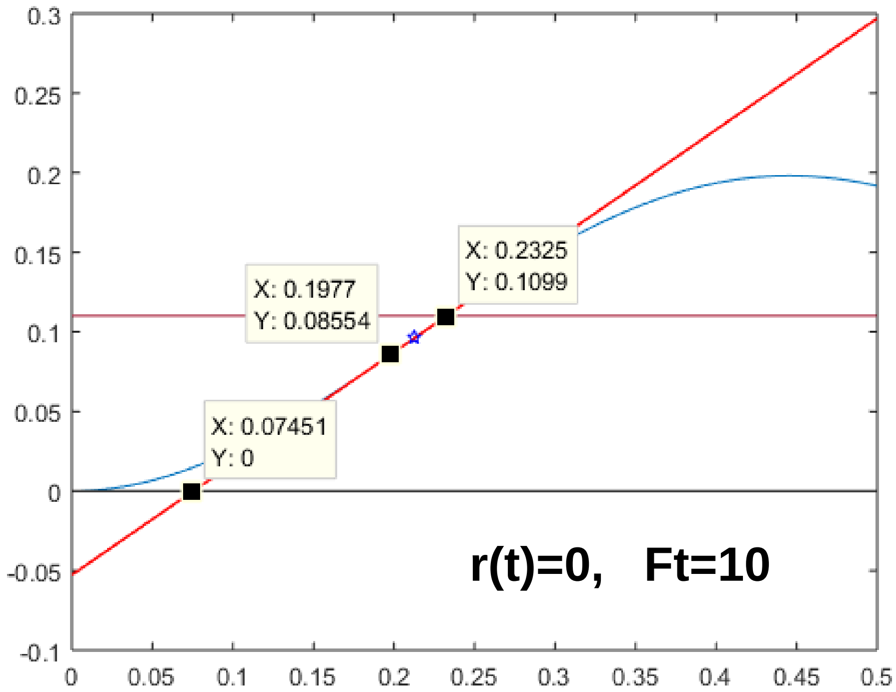
Open-loop System:

$F_c = u(t)K_a$ $K_a = 10$ m/N



Closed-loop System:





K=dcgain(Gp(s)), K=0.01

L=0.07451 and τ +L=0.2325

τ = 0.2325-0.07451 = 0.157

$$a = \frac{K*L}{\tau} = \frac{1x10^{-2}x0.07451}{0.157} = 4.76x10^{-3}$$

Z-N method:

$$K_p = \frac{1.2}{4.71x10^{-3}} = 254.45$$

$$T_i = 2xL = 2x0.07451 = 0.149$$

$$T_d = 0.5xL = 0.5x0.07451 = 0.0372$$

$$K_p = 254.45$$

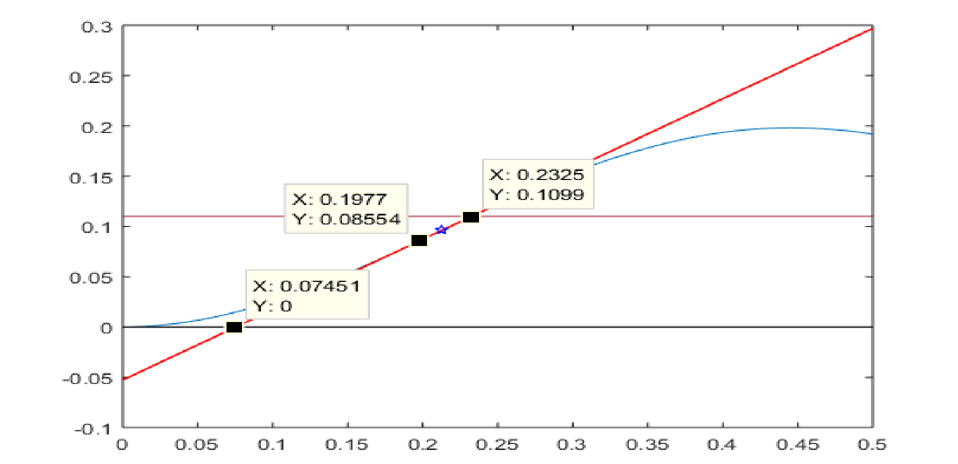
$$K_i = \frac{K_p}{T_i} = \frac{254.45}{0.149} = 1707.5$$

$$K_d = K_p x T_d = 3522.1x0.0372 = 9.479$$

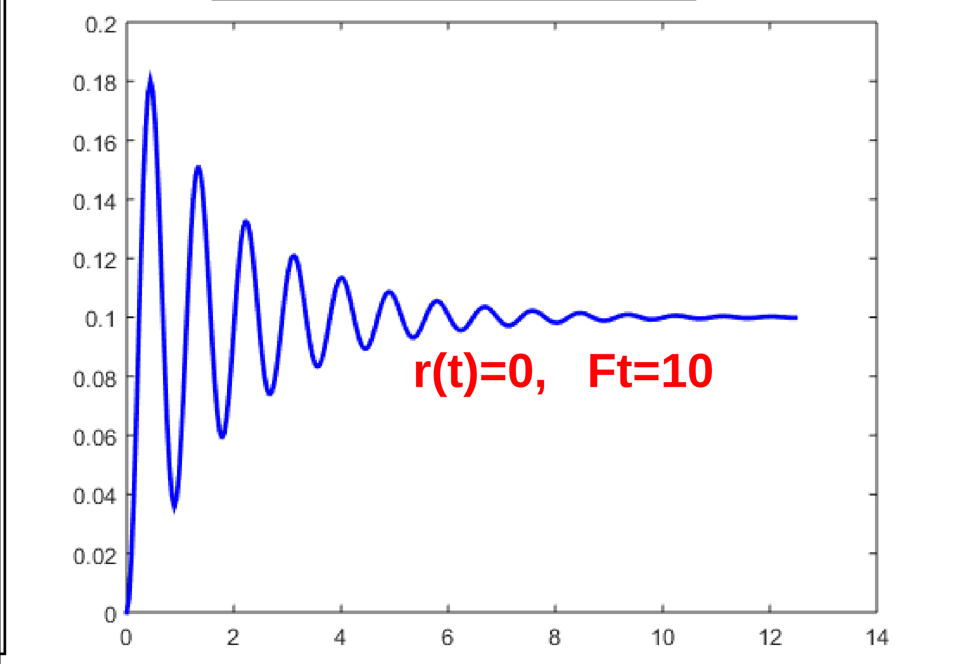
$$G_c(s) = 254.45(1 + \frac{1}{0.149s} + 0.0372s)$$

Matlab code for PID tuning algorithm:

```
clc,clear all,close all
x0=0;fd=10;tdelay=0;  % for open loop response
r0=0.1;                          % reference input
m=2;c=2;k=100;Ka=10;
np=1;dp=[m c k];gp=tf(np,dp,'inputdelay',tdelay);
p=roots(dp);p0=p(1);cksi_v2;
dt=0.0444;ts=12.5;t=0:dt:ts+tdelay;
xo=step(fd*gp,t); plot(t,xo,'b','Linewidth',2), return
ns=floor(tdelay/dt)+1;
xc=xo(ns:end);tcut=t(ns:end)-tdelay;
 % Open-loop response must be positive sign
plot(tcut,xc,'b'),xoss=0.1;
hold on;plot(t,xoss*ones(length(t)));
t1=0.1865;t2=0.2353;
y1=0.07073;y2=0.1016;
y=y1+((t-t1)*(y2-y1))/(t2-t1);plot(t,y,'r')
plot(t,zeros(length(t)),'k')
L=0.07548;topL=0.2353;
Tao=topL-L;
K=dcgain(gp);
a=K*L/Tao;
disp(['K"  "L"  "Tao" "a']);
disp([K L Tao a]);close,
```



Open Loop Response:



$$G_p(s) = \frac{1}{2s^2 + 2s + 100}$$

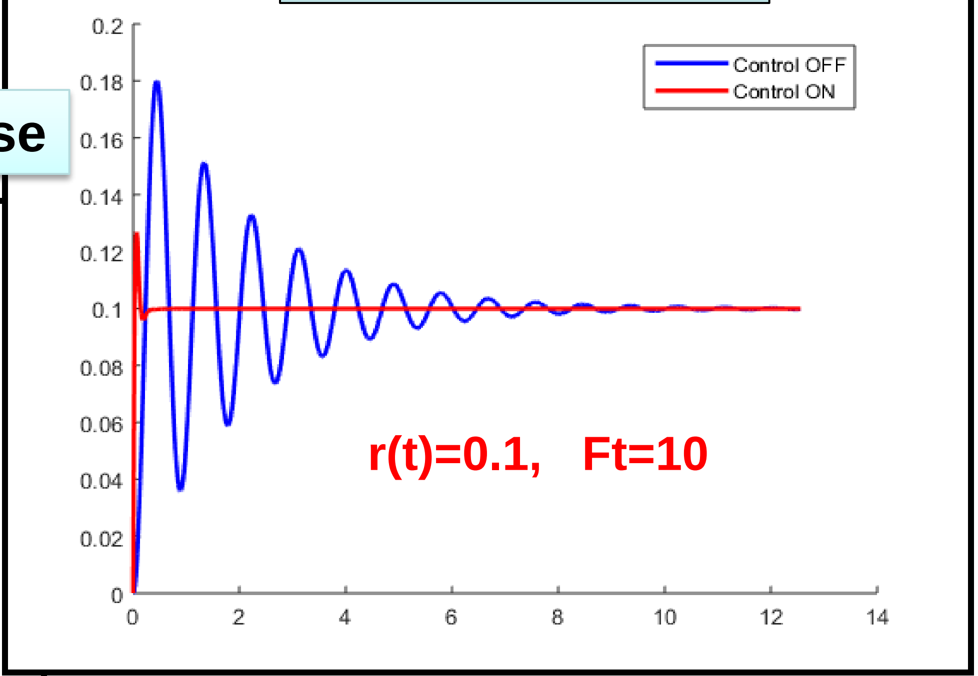
K=0.001, L=0.07451, τ = 0.156

$$G_c(s) = 254.45\left(1 + \frac{1}{0.149s} + 0.0372s\right)$$

Matlab code for closed-loop response

```
kp=1.2/a;ti=2*L;td=0.5*L;ki=kp/ti;kd=td*kp;
disp('PID gains');
disp(vpa([kp ki kd],4)),%close
 nc=[kd kp ki];dc=[1 0];gc=tf(nc,dc);
nh=conv(Ka*np,nc);dh=polyadd(conv(dc,dp),nh);
h=tf(nh,dh)
nhn=conv(np,dc);hn=tf(nhn,dh,'inputdelay',tdelay)
p=roots(dh);p0=p(1);cksi_v2;
p0=p(3);cksi_v2;
dt=0.00361;ts=12.56;t=0:dt:ts;
xo=step(fd*gp,t);
hold on,plot(t,xo,'b','Linewidth',2),
xr=step(r0*h,t);xn=step(fd*hn,t);
xclsys=xr+xn;
hold on;plot(t,xclsys,'r','Linewidth',2)
legend('Control OFF','Control ON')
stepinfo(xclsys(ns:end),t(ns:end))
```

Close Loop Response:




```
    RiseTime: 2.3661e-02
SettlingTime: 2.4768e-01
 SettlingMin: 9.5948e-02
 SettlingMax: 1.2677e-01
   Overshoot: 2.6773e+01
  Undershoot: 0
        Peak: 1.2677e-01
    PeakTime: 6.4980e-02
```

CHR method :

K=0.01,   L=0.07451, τ = 0.157

With 0% overshoot for disturbance rejection:

$$G_c(s) = 201.44(1 + \frac{1}{0.379s} + 0.0312s)$$

With 20% overshoot for disturbance rejection:

$$G_c(s) = 254.45(1 + \frac{1}{0.315s} + 0.0312s)$$