

Дополнительное задание по лабораторной работе №4

Вы синтезировали контур скорости в п.4 с учетом того, что контур тока был настроен на линейный оптимум. Настройте, контур тока на биномиальный оптимум и перенастройте контур скорости с учетом этого. Пришлите коэффициенты ПД-регулятора скорости. Постройте график переходного процесса

Загрузка данных

```
load('data.mat')
```

Синтез регулятора

Расчет ПИ регулятора тока

```
syms Tt s L R
Wia_spml = 1/(L*s+R);
wol = 1/(3*Tt*s*(Tt*s+1))
```

wol =

$$\frac{1}{3 Tt s (Tt s + 1)}$$

```
wreg = wol/Wia_spml;
disp(collect(wreg, s))
```

$$\frac{L s + R}{(3 Tt^2) s^2 + (3 Tt) s}$$

```
syms Kpa Kia
Kpa = L/3/Tt;
Kia = R/3/Tt;
wreg_t = (Kpa+Kia/s)/(Tt*s+1);
disp(collect(wreg_t, s))
```

$$\frac{L s + R}{(3 Tt^2) s^2 + (3 Tt) s}$$

```
Tt = 0.005;
```

```
cr1.Kpa = ob.L/3/Tt
```

```
cr1 = struct with fields:
  Kpa: 1.5098
  Kia: 721.7834
```

```
cr1.Kia = ob.R/3/Tt;
kdw = 180/pi;
Tmdl = 0.1;
```

```
syms Tu Tt s Ce J
```

коэффициент датчика скорости

```
syms kdw
Wcr1 = 1/(3*Tt*s+1)
```

$$Wcr1 = \frac{1}{3 T_t s + 1}$$

```
Wob2 = Ce*kdw/(J*s)
```

$$Wob2 = \frac{C_e k_{dw}}{J s}$$

передаточная функция раз. системы, настроенный на биномиальный оптиум

```
disp(simplify(collect(Wcr1*Wob2,s)))
```

$$\frac{C_e k_{dw}}{J s (3 T_t s + 1)}$$

```
wol = 1/(3*Tu*s*(Tu*s+1));
Wreg = wol/(Wcr1*Wob2);
disp(simplify(collect(Wreg, s)))
```

$$\frac{J (3 T_t s + 1)}{3 C_e T_u k_{dw} (T_u s + 1)}$$

Проверка формул коэффициентов

```
syms kdw J Kg Tu Kda Kpa Tt s Tur Ki Kp
Kg = J/(3*Ce*Tu*kdw)
```

$$K_g = \frac{J}{3 C_e T_u k_{dw}}$$

Аналоговый ПДД

```
Kda = Kg*3*Tt
```

$$K_{da} = \frac{J T_t}{C_e T_u k_{dw}}$$

```
Kpa = Kg*1
```

$$K_{pa} = \frac{J}{3 C_e T_u k_{dw}}$$

```
simplify((Kda*s+Kpa)/(Tu*s+1))
```

```
ans =
```

$$\frac{J (3 T_t s + 1)}{3 C_e T_u k_{dw} (T_u s + 1)}$$

$$\frac{J (3 T_t s + 1)}{3 C_e T_u k_{dw} (T_u s + 1)}$$

Моделирование

```
cr1.Tt = 0.0001;  
  
cr1.Kpa = ob.L/3/cr1.Tt;  
cr1.Kia = ob.R/3/cr1.Tt;  
  
spl.Tu = cr1.Tt;  
kdw = 180/pi;  
Tmd1 = 1;  
To = 0.1*cr1.Tt;  
spl.Tur = 0.5*To;  
spl.Tu = spl.Tur;  
  
spl.Kg = ob.J/(3*ob.Ce*spl.Tu*kdw)
```

```
spl = struct with fields:  
    Tu: 5.0000e-06  
    Tur: 5.0000e-06  
    Kg: 5.3495e+04  
    Kda: 32.0969  
    Kpa: 2.1398e+04  
    Kd2dg: 1.2520e+06  
    Kp2dg: 2.1398e+04
```

Аналоговый ПД

```
spl.Kda = 3*spl.Kg*cr1.Tt;%аналоговая система  
spl.Kpa = spl.Kg*1;
```

Цифровой ПД

```
spl.Kd2dg = 3*spl.Kg*1/(exp(To/cr1.Tt)-1);%решение задачи на переходный процесс в апериодическом  
% данные коэффициенты позволяют компенсировать Д-звено  
spl.Kp2dg = spl.Kg*1;
```

Построение графиков переходных процессов

```
warning off  
Tmd1=0.1;  
simNew = sim('lab4_2_1.slx', 'ReturnWorkspaceOutputs','on')
```

```
simNew =
```

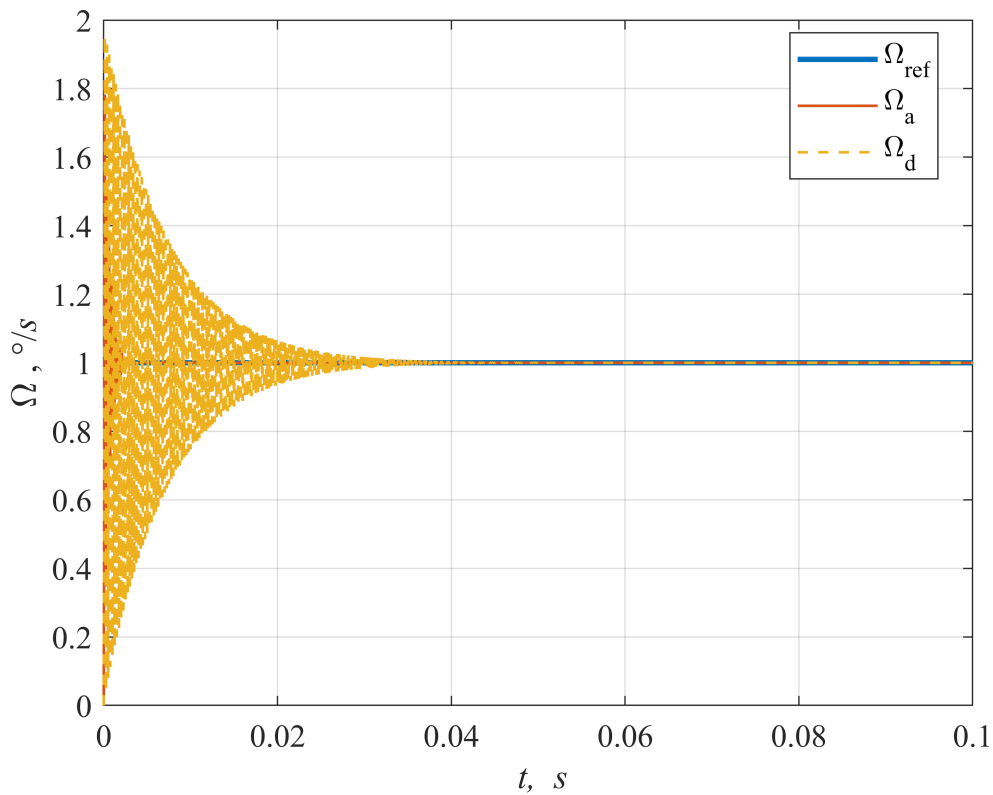
Simulink.SimulationOutput:

tout: [1000001x1 double]
w: [1000001x5 double]

SimulationMetadata: [1x1 Simulink.SimulationMetadata]
ErrorMessage: [0x0 char]

```
t = simNew.w(:,1);  
wref = simNew.w(:,2);  
wa = simNew.w(:,3);  
wd = simNew.w(:,4);
```

```
h=figure;  
set(h,'DefaultAxesFontSize', 12, 'DefaultAxesFontName','Times New Roman');  
  
plot(t,wref,'Linewidth', 2)  
hold on  
grid on  
plot(t,wa,'Linewidth', 1, 'LineStyle','-')  
plot(t,wd,'Linewidth', 1, 'LineStyle','--')  
xlabel('\it t,\rm \it s')  
ylabel('\Omega \rm, \circ/\its')  
legend('\Omega_{ref}', '\Omega_a', '\Omega_d','Location','best')
```



С запаздыванием

```
spl.Tur = 0.5*To;  
spl.Tz = 1.5*To;  
  
spl.Tu = spl.Tur+spl.Tz;  
  
spl.Kg = ob.J/(3*ob.Ce*spl.Tu*kdw)
```

```
spl = struct with fields:  
    Tu: 2.0000e-05  
    Tur: 5.0000e-06  
    Kg: 1.3374e+04  
    Kda: 4.0121  
    Kpa: 1.3374e+04  
    Kd2dg: 3.8148e+05  
    Kp2dg: 1.3374e+04  
    Tz: 1.5000e-05
```

Аналоговый ПД

```
spl.Kda = 3*spl.Kg*cr1.Tt;%аналоговая система  
spl.Kpa = spl.Kg*1;
```

Цифровой ПД

```
spl.Kd2dg = 3*spl.Kg*1/(exp(To/cr1.Tt)-1);%решение задачи на переходный процесс в апериодическом  
% данные коэффициенты позволяют компенсировать Д-звено  
spl.Kp2dg = spl.Kg*1;  
warning off  
Tmdl=0.005;  
simNew = sim('lab4_2_1_1.slx', 'ReturnWorkspaceOutputs','on')
```

```
simNew =  
    Simulink.SimulationOutput:  
  
        tout: [50001x1 double]  
         w: [50001x5 double]  
  
    SimulationMetadata: [1x1 Simulink.SimulationMetadata]  
    ErrorMessage: [0x0 char]
```

```
t = simNew.w(:,1);  
wref = simNew.w(:,2);  
wa = simNew.w(:,3);  
wd = simNew.w(:,4);
```

```
h=figure;  
set(h,'DefaultAxesFontSize', 12, 'DefaultAxesFontName','Times New Roman');  
  
plot(t,wref,'Linewidth', 2)  
hold on  
grid on
```

```

plot(t,wa,'Linewidth', 1, 'LineStyle','-')
plot(t,wd,'Linewidth', 1, 'LineStyle','--')
xlabel('\it t,\rm \it s')
ylabel('\Omega \rm, \circ/\it s')
legend('\Omega_{ref}', '\Omega_a', '\Omega_d','Location','best')

```

