ROBOTIRÁNYÍTÁS

9. előadás

Soros kompenzátorok, PID szabályozó

Dr. habil. Kovács Levente
egyetemi docens

Dr. Haidegger Tamás
egyetemi adjunktus

EKIK Egyetemi Kutató és Innovációs Központ

K-MOOC

ÓBUDAI EGYETEM Bejczy Antal iRobottechnikai Központ

Élettani Szabályozások Csoport Óbudai Egyetem

Tartalom

1. Soros kompenzátorok

1.1. Soros kompenzálás elve

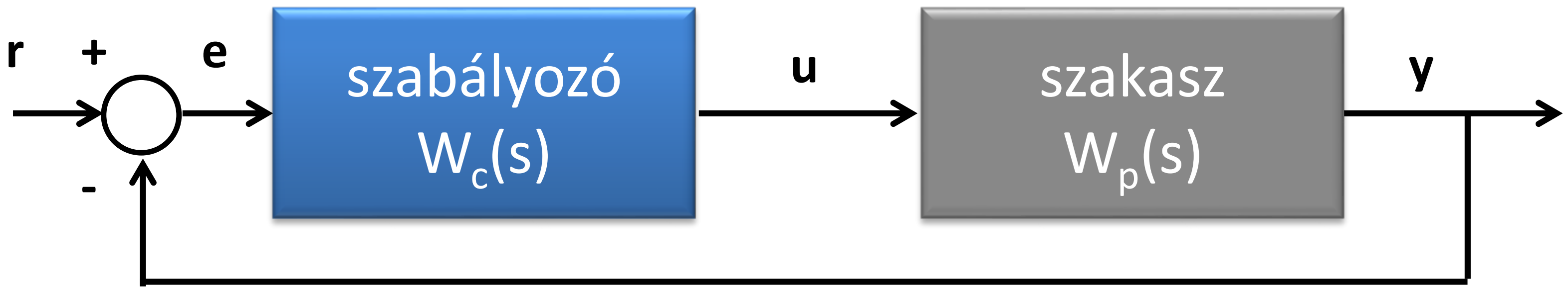
1.2. A P, I és a D tag

1.3. Soros kompenzátorok típusai

1.4. Ideális és közelítő PID szabályozó

1.5. Közelítő PID típusú szabályzó tulajdonságai

Soros kompenzálás elve



- szabályozó
  - ✓ a szakasszal sorba kapcsolt tag (szakasz előtt van)
  - ✓ bemenő jele a hibajel (*e*), kimenő jele a beavatkozó jel (*u*)
  - ✓ átviteli függvényét ($W_c(s)$) úgy tervezzük meg, hogy zárt körben a szakasz nem megfelelő tulajdonságai kikompenzálódjanak → szabályozó = kompenzátor
- szabályozási körrel szembeni elvárások
  1. stabil legyen
  2. egyéb minőségi tulajdonság:
     a) statikus pontosság
     b) zajelnyomás
     c) gyorsaság

csak a stabilitás rovására növelhetők bizonyos határon túl

A P, I és a D tag

P	arányos tag	a hiba aktuális értékét veszi figyelembe
I	integráló tag	a hiba korábbi értékeit veszi figyelembe
D	deriváló tag	a hiba alakulását veszi figyelembe

Soros kompenzátorok típusai

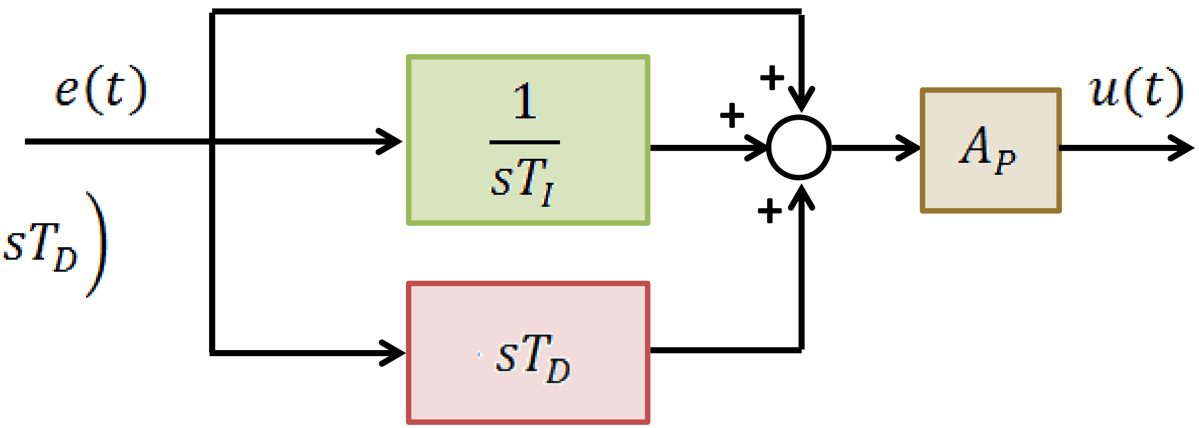
Jel.	Típus	Átviteli függvény	Paraméterek
P	arányos	$W_P(s) = A_P$	A_P erősítés
I	integráló	$W_I(s) = \dfrac{1}{sT_I}$	T_I integrálási idő
D_i	ideális deriváló	$W_{D_i}(s) = sT_D$	T_D deriválási idő
D	közelítő deriváló	$W_D(s) = \dfrac{sT_D}{1 + sT_C}$	T_D deriválási idő; T_C időállandó

fizikai rendszerrel nem valósítható meg

Ideális és közelítő PID szabályozó

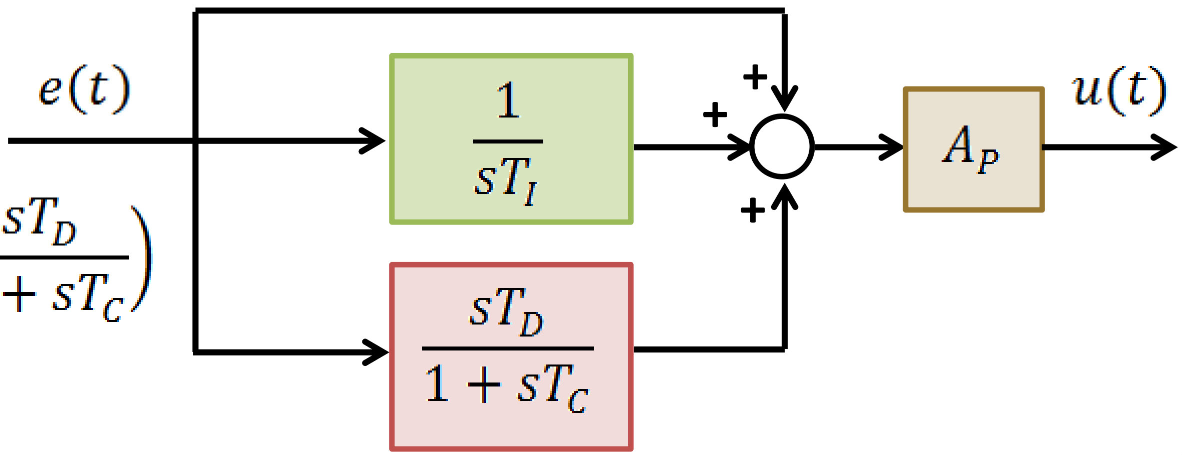
Ideális PID szabályozó

$$W_{PID}(s) = A_P \cdot \left(1 + \frac{1}{sT_I} + sT_D\right)$$



Közelítő PID szabályozó

$$W_{PID}(s) = A_P \left(1 + \frac{1}{sT_I} + \frac{sT_D}{1 + sT_C}\right)$$



Közelítő PID típusú szabályzó tulajdonságai

- átviteli függvény:

$$W_{PID}(s) = A_P\left(1 + \frac{1}{sT_I} + \frac{sT_D}{1+sT_C}\right) = \frac{A_P}{T_I}\cdot\frac{1+(T_I+T_C)s+T_I(T_D+T_C)s^2}{s(1+sT_C)}$$

- paraméterek:
  - ✓ erősítés: $A_P > 0$
  - ✓ időállandók: $T_I, T_D, T_C > 0$

- a szabályzó megváltoztatja
  - ✓ erősítést → $K = \frac{A_P}{T_I}\cdot K_{plant}$
  - ✓ eggyel növeli a nyitott kör átviteli függvényének típusszámát
  - ✓ a nyitott körben két új zérus jelenik meg
    → a zérusok komplex konjugált póruspárt alkothatnak megfelelő időállandók választása esetén
  - ✓ a nyitott körben két új pólus jelenik meg

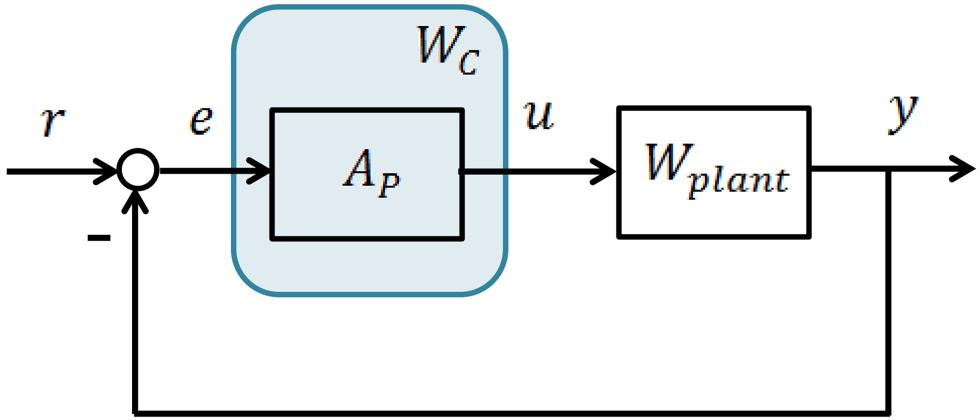
2. Soros kompenzátor tervezése előírt fázistartalékra

2.1. P szabályozó tervezése előírt fázistartalékra

2.2. Közelítő PID szabályozó tervezése előírt fázistartalékra

P szabályozó tervezése előírt fázistartalékra

Feladat



Tervezzünk P típusú szabályozót, mely W_C átviteli függvényű (az erősítése A_P) a W_{plant} átviteli függvényű szakaszhoz.
Az előírt fázistartalék legyen $\varphi_t = 45°$

$$W_{plant} = \frac{4}{(1+10s)(1+3s)(1+0.2s)}$$

P szabályozó tervezése előírt fázistartalékra

A tervezés koncepciója

1. Először tekintsünk egy P szabályozót $A_P = 1$ értékű erősítéssel és keressük meg azt a frekvenciát, ahol a felnyitott kör fázismenete

$$\varphi|_{\omega_o} = -180° + \varphi_t = -180° + 45° = -135°$$

2. A megkeresett frekvencia ω_o → ezt kell vágási frekvenciának választani

3. Az ω_o frekvencián a felnyitott kör erősítését állítsuk be 1-re

$$K_{open}|_{\omega_o} = K_P \cdot K_{plant} = 1 \leftrightarrow 0 \text{ dB}$$

4. Az ω_o frekvencián $A_P = 1$ erősítés mellett a felnyitott kör amplitúdó erősítése A_1

$$A_P \cdot A_1 = 1 \Rightarrow A_P = \frac{1}{A_1}$$

P szabályozó tervezése előírt fázistartalékra

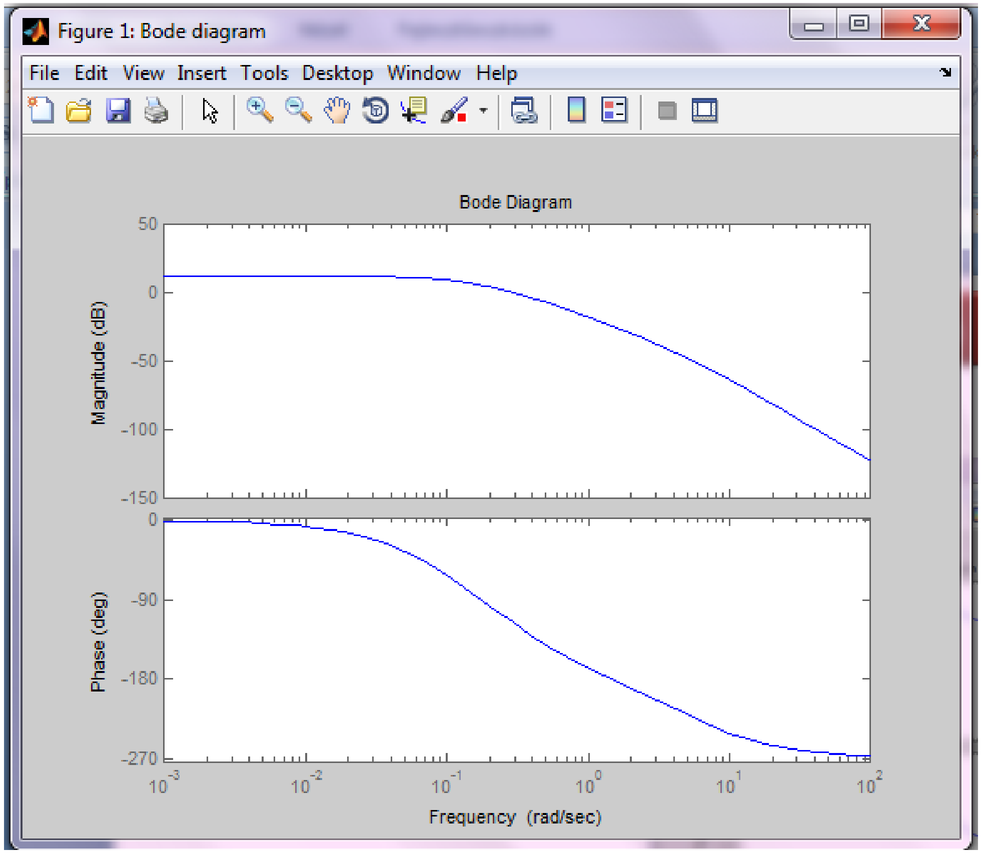
Megoldás Matlabban

```
% szakasz megadása
disp('A szabályozni kívánt szakasz:')
Wplant = zpk([], [-1/10 -1/3 -1/0.2], 4/(10*3*0.2))

% előírt fázistartalék
pm = 45;

% szabályozó K_P = 1 erősítéssel
num_Wc = 1;
den_Wc = 1;
Wc = tf(num_Wc, den_Wc);

% nyitott kör
W0 = series(Wc, Wplant);
```

P szabályozó tervezése előírt fázistartalékra

Megoldás Matlabban

```
%% 1. grafikus megoldás
figure('name','Bode diagram');
bode(W0);
```

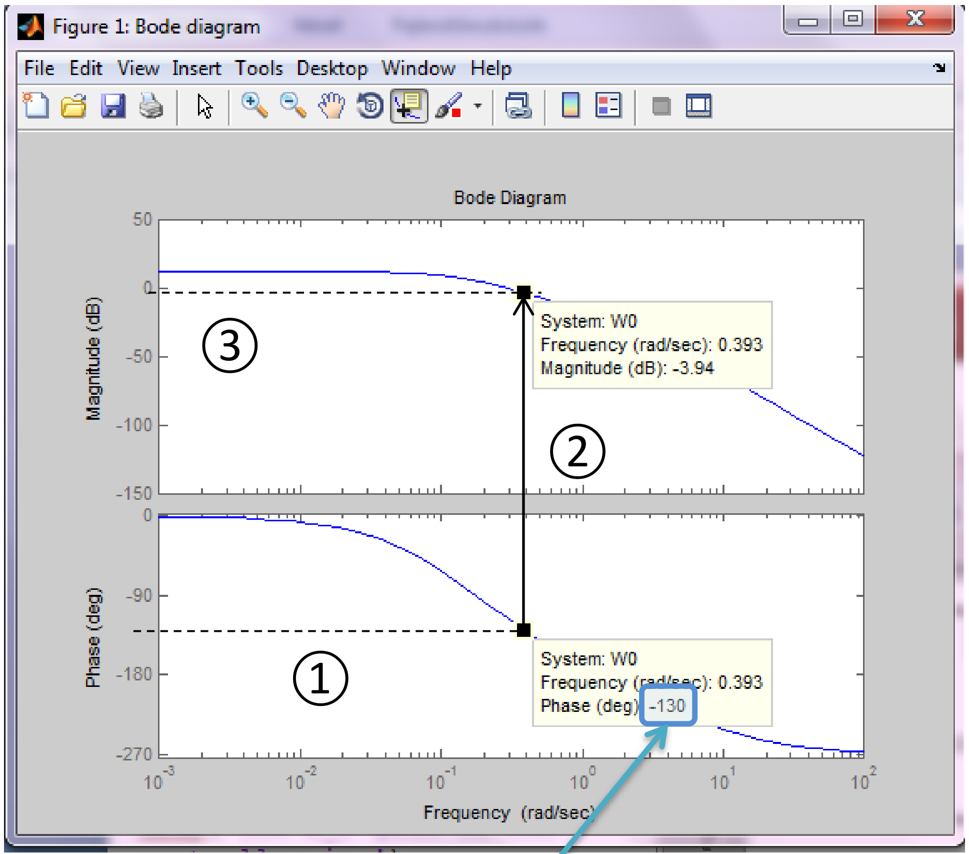


P szabályozó tervezése előírt fázistartalékra

Megoldás Matlabban

```
% keressük meg a -135 fokot
a fázismeneten
disp('A Bode diagram alapján
azt találtuk, hogy a vágási
frekvencia 0.393')
disp('A Bode diagram alapján
azt találtuk, hogy az
amplitúdó közelítőleg
-3.94 dB')
disp('A szabályozó A_P
erősítése ezek alapján:')
A_P_graphical = 10^(3.94/20)
% A_P [dB] = 20*log10(K_P)
```

```
A_P_graphical =
      1.5740
```

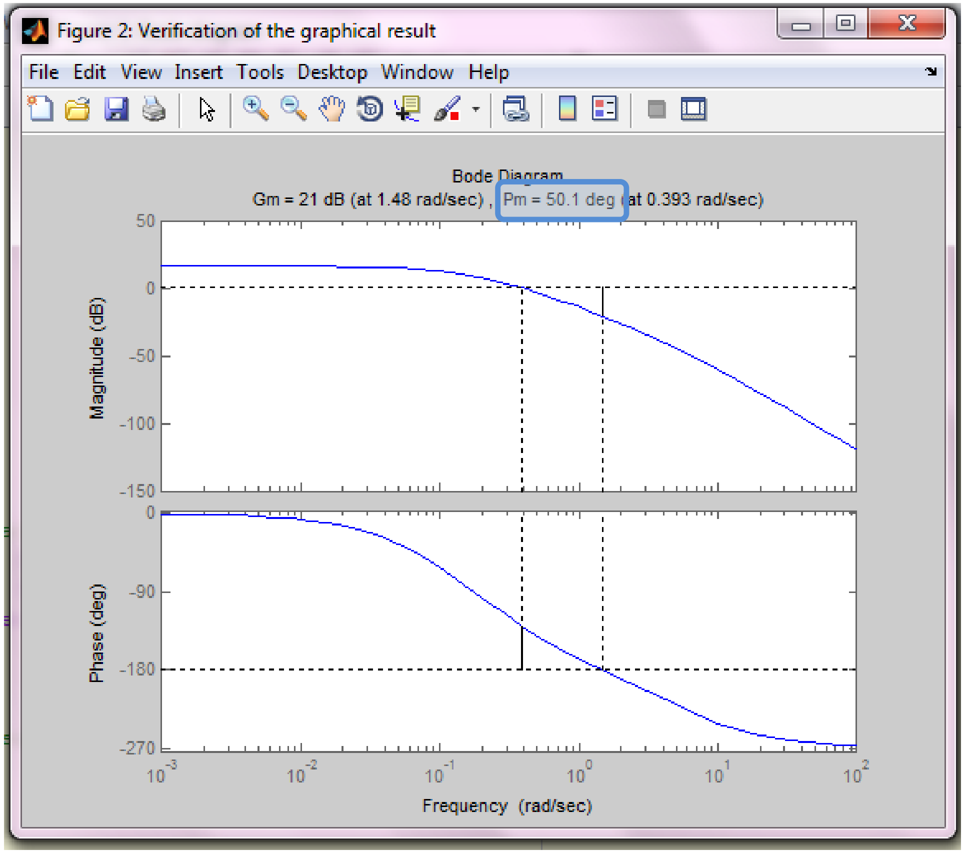


ebben az esetben nem találtuk meg a pontos -135°-ot

P szabályozó tervezése előírt fázistartalékra

Az eredmény ellenőrzése

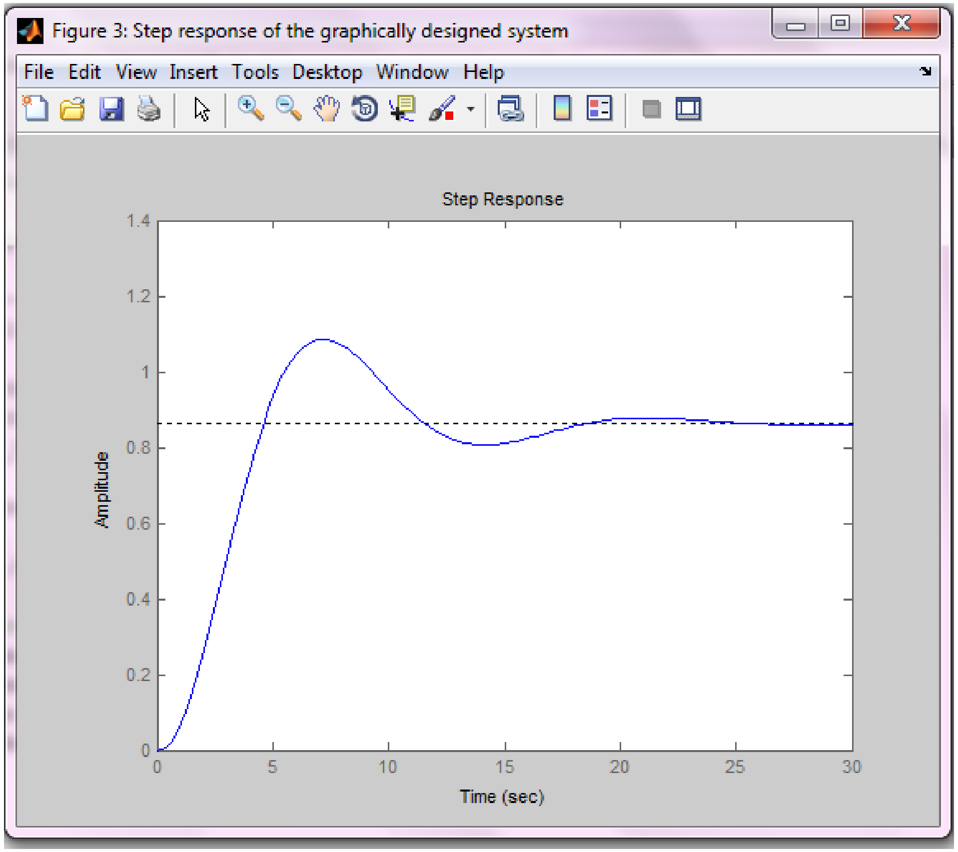
```
% a grafikus megoldás
ellenőrzése
num_Wc_v1 = A_P_graphical;
den_Wc_v1 = 1;
Wc_v1 = tf(num_Wc_v1,
den_Wc_v1);
W0_v1 = series(Wc_v1,
Wplant);
figure('name','Verification
of the graphical result');
margin(W0_v1)
```



P szabályozó tervezése előírt fázistartalékra

Az eredmény ellenőrzése

```
% a grafikusan tervezett
rendszer ugrásválasza
feedback_tf = tf(1,1); %
egységnyi erősítésű blokk
Wclosed_v1 = feedback(W0_v1,
feedback_tf);
figure('name','Step response
of the graphically designed
system');
step(Wclosed_v1)
```



P szabályozó tervezése előírt fázistartalékra

Megoldás Matlabban

```
%% 2. numerikus megoldás
[mag,phase,w]=bode(W0);

% a vágási frekvencia megkeresése
phi = abs((180+phase)-pm); % phi tartalmazza az egyes fázisértékek
esetén az előírt fázistartaléktól való távolságot
[min_value, phi_index] = min(phi); % min_value a nullához legközelebbi
                                    % phi_index az ehhez tartozó index

disp('A vágási körfrekvencia:')
w(phi_index)
0.3932
disp('A nyitott kör fázisa a vágási frekvencián:')
phase(phi_index)
-129.9407
disp('A nyitott kör amplitúdója a vágási frekvencián:')
mag(phi_index)
0.6355
```

P szabályozó tervezése előírt fázistartalékra

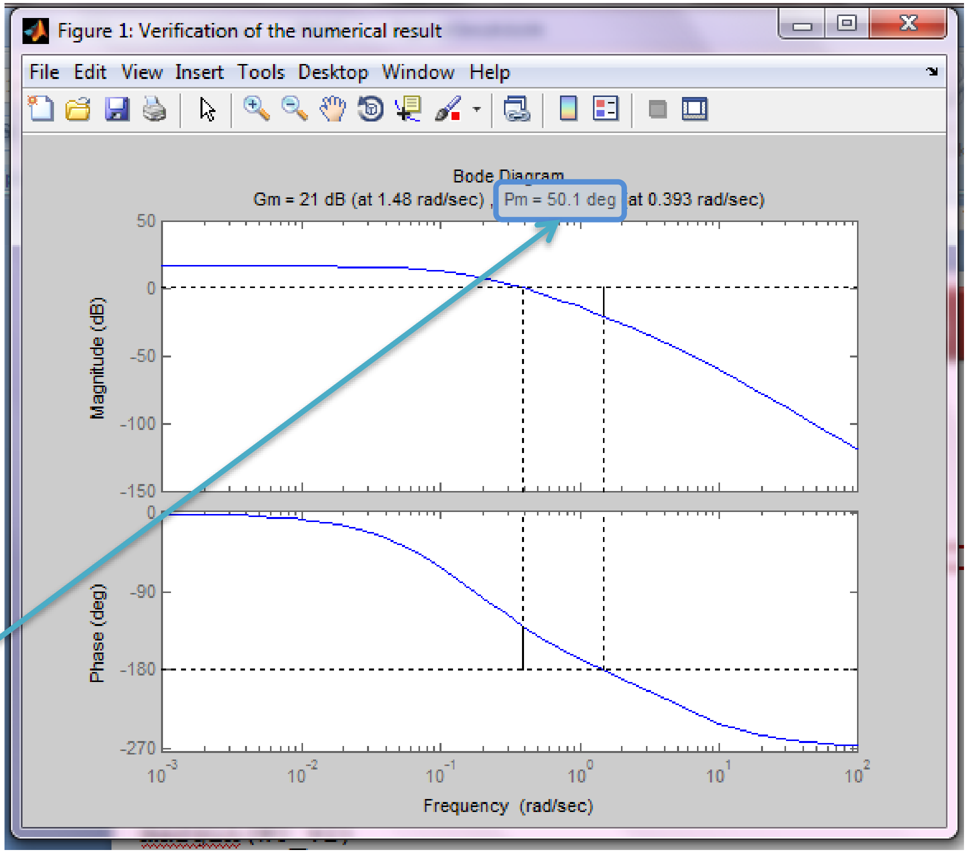
Megoldás Matlabban

```
% a szabályozó A_P erősítésének megkeresése
disp('A szabályozó A_P erősítése:')
A_P_numerical = 1/mag(phi_index)
```

```
A szabályozó A_P erősítése:

A_P_numerical =

    1.5735
```

Az eredmény ellenőrzése

```
% a numerikus megoldás
ellenőrzése
num_Wc_v2 = A_P_numerical;
den_Wc_v2 = 1;
Wc_v2 = tf(num_Wc_v2,
den_Wc_v2);
W0_v2 = series(Wc_v2,
Wplant);
figure('name','Verification
of the numerical result');
margin(W0_v2)
```

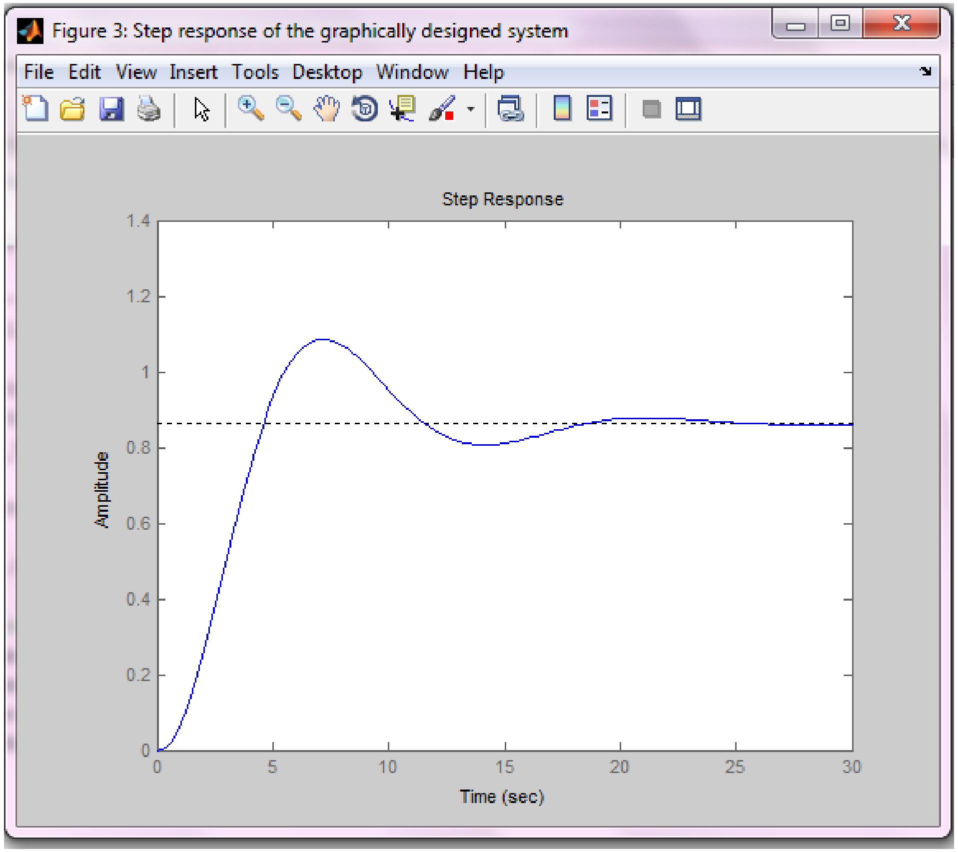


a numerikus hiba annak köszönhető, hogy a Matlab bode függvénye az amplitúdó- és fázisdiagramot diszkrét frekvencia-értékeken értékeli ki

P szabályozó tervezése előírt fázistartalékra

Az eredmény ellenőrzése

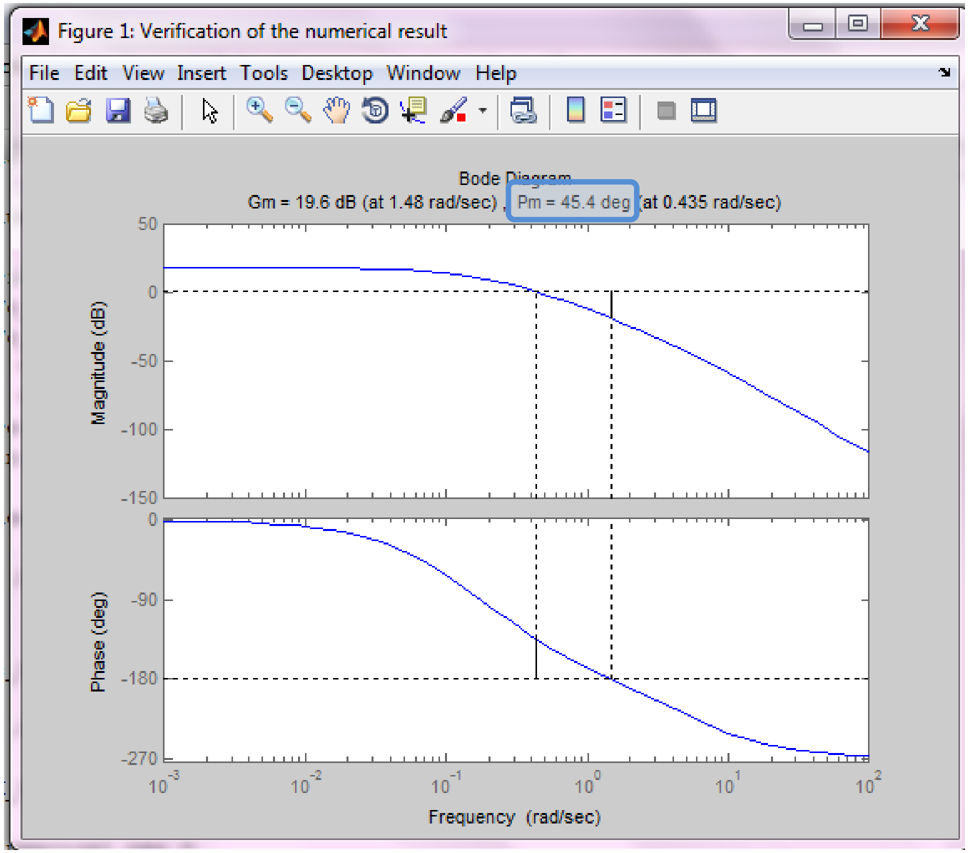
```
% a grafikusan tervezett
rendszer ugrásválasza
feedback_tf = tf(1,1); %
egységnyi erősítésű blokk
Wclosed_v2 = feedback(W0_v2,
feedback_tf);
figure('name','Step response
of the graphically designed
system');
step(Wclosed_v2)
```



P szabályozó tervezése előírt fázistartalékra

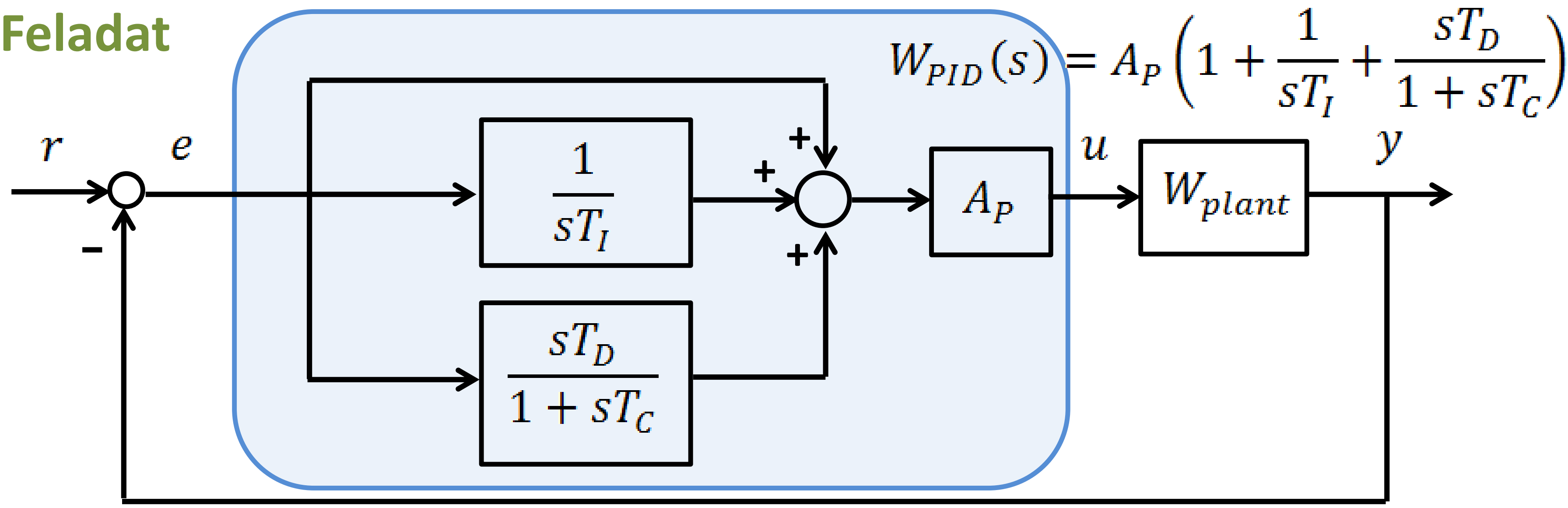
Az eredmény pontosítása

```
% a kapott pontatlan vágási
frekvencia: 0.3932

% adott frekvenciához
megkeressük a pontos amplitúdó
és fázis értékeket
[mag_new,phase_new] =
bode(W0,0.435)

% a szabályozó A_P erősítésének
megkeresése
disp('A szabályozó A_P
erősítése:')
A_P_numerical_new = 1/mag_new
```



Közelítő PID szabályozó tervezése előírt fázistartalékra

Feladat



Tervezzünk közelítő PID típusú szabályozót a W_{plant} átviteli függvényű szakaszhoz. Legyen az előírt fázistartalék $\varphi_t = 45°$, az állandósult állapotbeli hiba pedig $e_\infty = 0$.

$$W_{plant} = \frac{4}{(1+10s)(1+3s)(1+0.2s)}$$

Közelítő PID szabályozó tervezése előírt fázistartalékra

A tervezés koncepciója

- alap koncepció: pólus-zérus kiejtés

$$W_{PID}(s) = A_P\left(1 + \frac{1}{sT_I} + \frac{sT_D}{1+sT_C}\right) = \frac{A_P}{T_I}\cdot\frac{1+(T_I+T_C)s+T_I(T_D+T_C)s^2}{s(1+sT_C)}$$

$$= \frac{A_P}{T_I}\cdot\frac{(1+s\tau_1)(1+s\tau_2)}{s(1+sT_C)} \quad \Rightarrow \quad \begin{aligned}\tau_1+\tau_2 &= T_I+T_C\\ \tau_1\tau_2 &= T_I(T_D+T_C)\end{aligned}$$

1. Az $e_\infty = 0$ garantált az integrátor használata miatt
2. A szakasz két leglassabb pólusának kiejtése

$$T_C = 0.1T_D$$

$$\tau_1 = T_1$$

$$\tau_2 = T_2$$

A tervezés koncepciója

- az egyenletrendszer megoldása

$$\left.\begin{aligned} \tau_1 + \tau_2 &= T_I + T_C \\ \tau_1 \tau_2 &= T_I (T_D + T_C) \\ T_C &= 0.1 T_D \\ \tau_1 &= T_1 \\ \tau_2 &= T_2 \end{aligned}\right\}$$

$$0.11 T_D^{\ 2} - 1.1(\tau_1 + \tau_2) T_D + \tau_1 \tau_2 = 0$$

$$\rightarrow T_{D,1}, T_{D,2}$$

$$T_I = \tau_1 + \tau_2 - 0.1 T_D$$

megoldás: $T_{D,1}, T_{D,2} > 0$ és $T_I > T_D$

3. Az előírt fázistartalék eléréséhez

$$\frac{A_P}{T_I} \cdot A_1 = 1 \Rightarrow A_P = \frac{T_I}{A_1}$$

Közelítő PID szabályozó tervezése előírt fázistartalékra

Megoldás Matlabban

```
% szakasz megadása
A = 4; T1 = 10; T2 = 3; T3 = 0.2;
num_plant = A;
den_plant = conv(conv([T1 1],[T2 1]),[T3 1]);
W_plant = tf(num_plant,den_plant)

% előírt fázistartalék
pm = 45;

% a szakasz két leglassabb pólusának kiejtése
tau1 = T1;
tau2 = T2;
```

Közelítő PID szabályozó tervezése előírt fázistartalékra

Megoldás Matlabban

$$0.11{T_D}^2 - 1.1(\tau_1 + \tau_2)T_D + \tau_1\tau_2 = 0$$

```
% Td számítása
Td = min(roots([0.11 -1.1*(tau1+tau2) tau1*tau2]));
% roots: a polinomok által definiált egyenlet gyökeinek
megkeresése

if Td < 0
    Td = max(roots([0.11 -1.1*(tau1+tau2) tau1*tau2]));
end;

% Ti számítása
Ti = tau1 + tau2 - 0.1*Td;
```

$$T_I = \tau_1 + \tau_2 - 0.1T_D$$

Közelítő PID szabályozó tervezése előírt fázistartalékra

Megoldás Matlabban

$$W_{PID}(s) = \frac{A_P}{T_I} \cdot \frac{(1+s\tau_1)(1+s\tau_2)}{s(1+sT_C)}$$

```
% „előszabályozó" Ap = 1-el
num_p_pid = 1/Ti * conv([tau1 1],[tau2 1]);
den_p_pid = [0.1*Td 1 0]; % Tc = 0.1*Td
W_p_pid = tf(num_p_pid, den_p_pid);

% nyitott kör
W0_p_pid = series(W_p_pid, W_plant);
W0_p_pid = minreal(W0_p_pid); % minreal: egyszerűsítés az
átviteli függvényben (a kiejtés miatt)

% a vágási frekvencia megkeresése
[mag,phase,w] = bode(W0_p_pid);
phi = abs((180+phase)-pm);
[min_value, phi_index] = min(phi);
```

Közelítő PID szabályozó tervezése előírt fázistartalékra

Megoldás Matlabban

$$W_{PID}(s) = \frac{A_P}{T_I} \cdot \frac{(1 + s\tau_1)(1 + s\tau_2)}{s(1 + sT_C)}$$

```
disp('A vágási körfrekvencia:')
w(phi_index)

disp('A nyitott kör fázisa a vágási frekvencián:')
phase(phi_index)

disp('A nyitott kör amplitúdója a vágási frekvencián:')
mag(phi_index)

% Ap számítása
Ap = 1/mag(phi_index);
```

Közelítő PID szabályozó tervezése előírt fázistartalékra

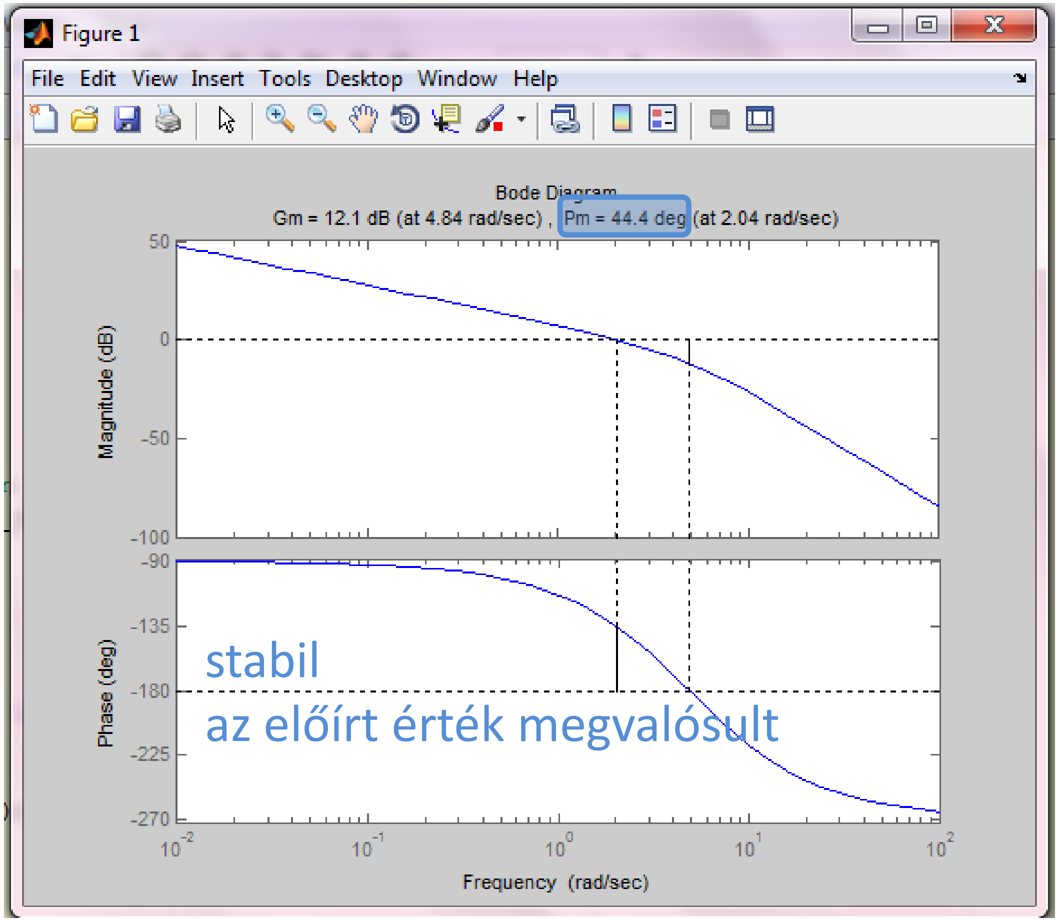
Megoldás Matlabban

$$W_{PID}(s) = \frac{A_P}{T_I} \cdot \frac{(1+s\tau_1)(1+s\tau_2)}{s(1+sT_C)}$$

```
% controller
disp('A PID szabályozó paraméterei:')
Ap
Ti
Td
Tc = 0.1*Td

num_pid = Ap/Ti * conv([tau1 1],[tau2 1]);
den_pid = [0.1*Td 1 0]; % Tc = 0.1*Td
W_pid = tf(num_pid, den_pid);

disp('PID szabályozó:')
W_pid
```

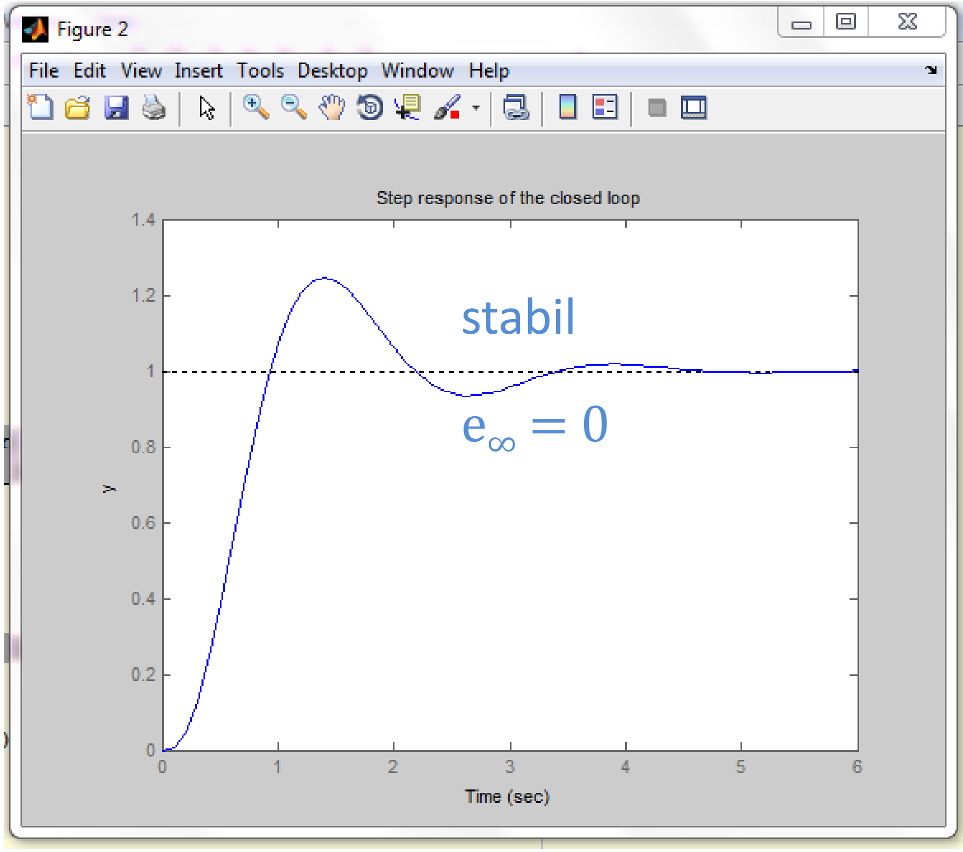
Az eredmény ellenőrzése

```
% nyitott kör
W0_pid = series(W_pid,
W_plant);

% Bode diagram
figure()
margin(W0_pid);
```



Az eredmény ellenőrzése

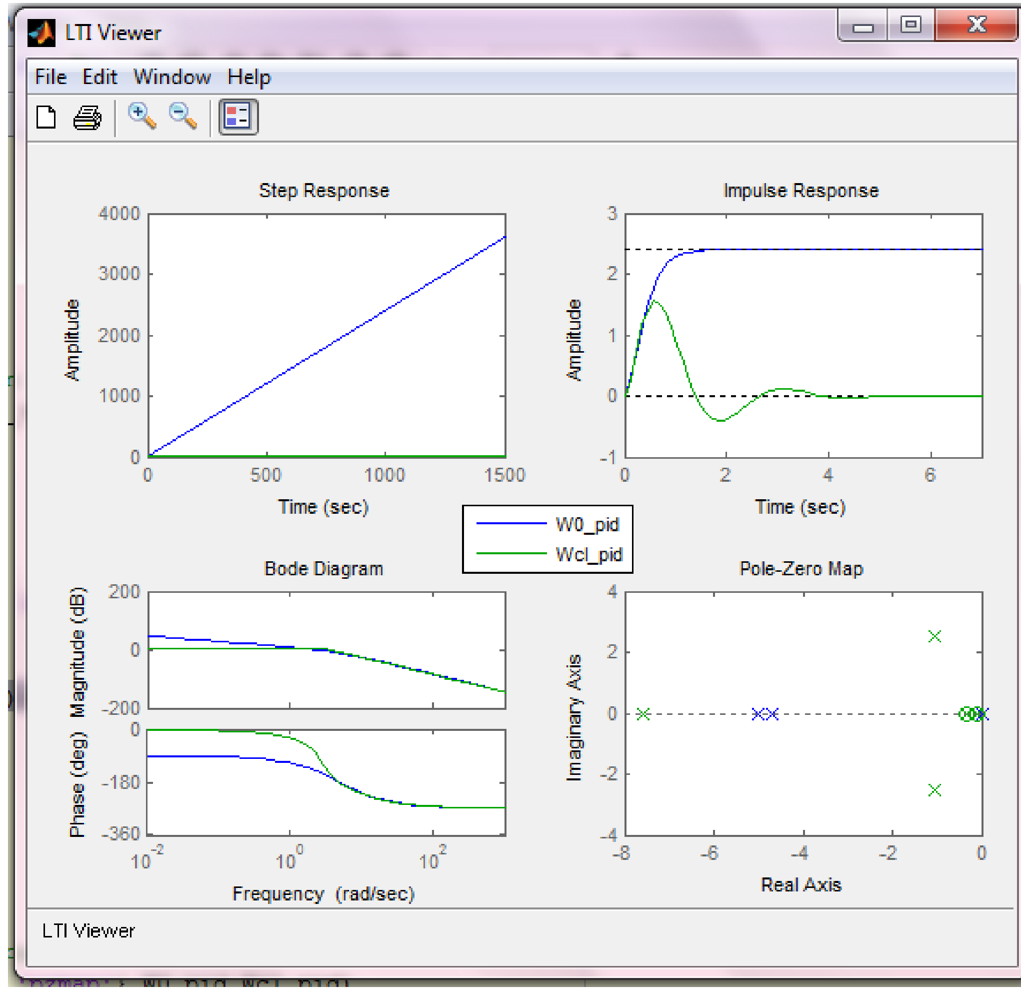
```
% zárt kör
feedback_tf = tf(1,1);
Wcl_pid =
feedback(W0_pid,feedback_tf);

% a zárt kör ugrásválasza
figure
step(Wcl_pid)
ylabel('y')
title('Step response of the
closed loop')
```



Közelítő PID szabályozó tervezése előírt fázistartalékra

Az eredmény ellenőrzése

```
% nyitott és zárt kör
összehasonlítása

ltiview({'step';
'impulse';'bode';
'pzmap'},W0_pid,Wcl_pid)
```



Videó

[Közelítő PID szabályozó mintavételes megvalósítása]

Köszönöm a figyelmet!

Dr. habil. Kovács Levente
egyetemi docens

kovacs.levente@nik.uni-obuda.hu

Dr. Haidegger Tamás
egyetemi adjunktus

haidegger@irob.uni-obuda.hu

EKIK Egyetemi Kutató és Innovációs Központ

K-MOOC

Óbudai Egyetem Bejczy Antal iRobottechnikai Központ

Physiological Controls Group

Élettani Szabályozások Csoport
Óbudai Egyetem