ROBOTIRÁNYÍTÁS

12. előadás
Alapjel miatti korrekció, állapotmegfigyelő, LQ szabályozás

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

Physiological Controls Group

Élettani Szabályozások Csoport
Óbudai Egyetem

Tartalom

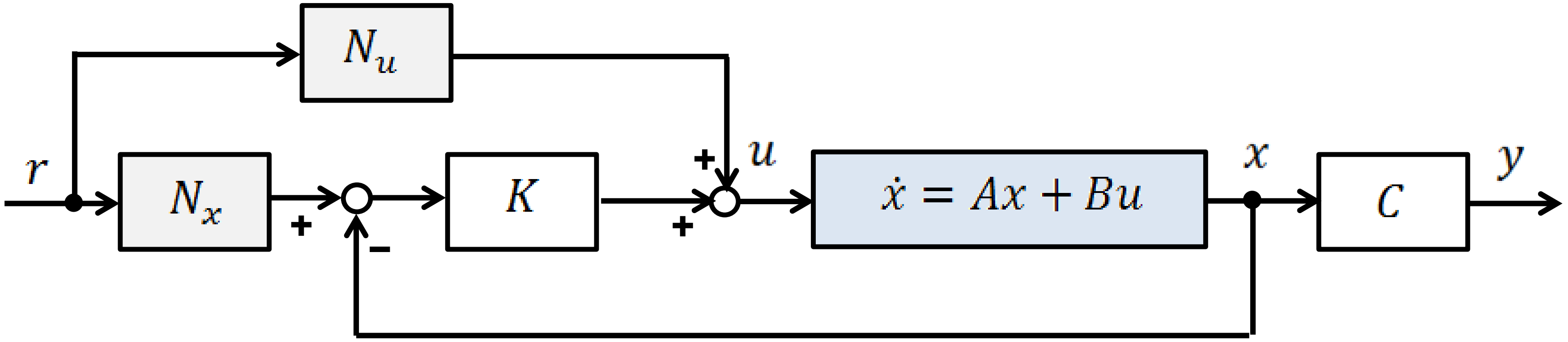
1. Alapjel miatti korrekció

1.1. Alapjel miatti korrekció megvalósítása

1.2. Alapjel miatti korrekció – példa

Alapjel miatti korrekció megvalósítása

- ha a szabályozás célja nem csak az, hogy stabilizáljuk a rendszert
- hanem az is, hogy egy nem nulla referencia jelet kövessen,
- a blokk diagramot ki kell terjeszteni olyan speciális blokkokkal, amelyek a referencia jel követését lehetővé teszik (N_x és N_u)
- feltesszük, hogy a szakasz D mátrixa nulla, így $y = Cx$
- állapotvisszacsatolás nem nulla alapjel esetén

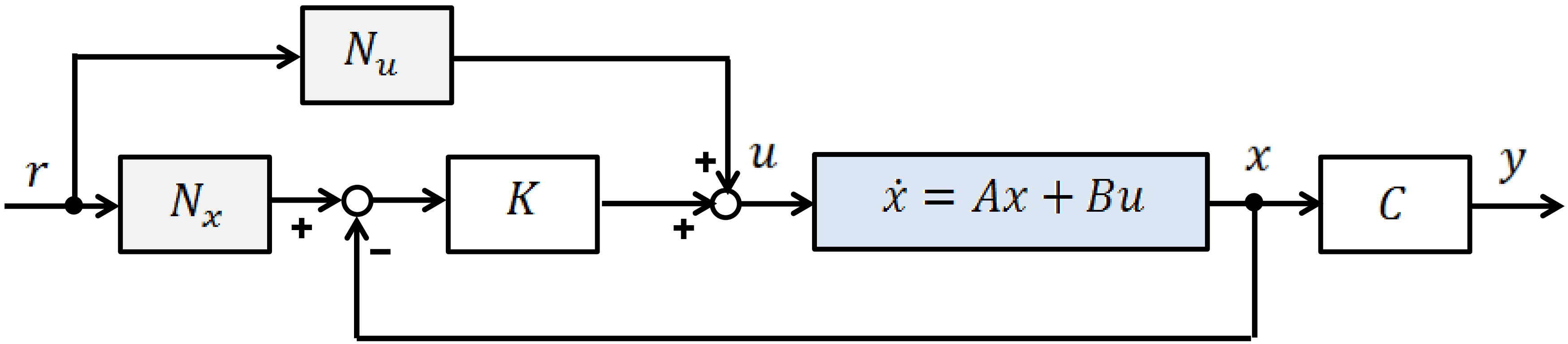


Alapjel miatti korrekció megvalósítása

• amikor az N_x és N_u vektorokat számítjuk, akkor feltesszük, hogy egységugrás bemenet esetén

✓ a kivonó tag után a jel értéke 0, ①

✓ és a kimenet értéke 1 állandósult állapotban ②



① $N_x r - x_\infty = 0 \quad \Rightarrow \quad x_\infty = N_x r$

② $y_\infty = r \qquad y_\infty = Cx_\infty = CN_x r = r \quad \Rightarrow \quad CN_x = I$

$$u_\infty = N_u r \qquad \dot{x}_\infty = Ax_\infty + Bu_\infty = 0 \Rightarrow (AN_x + BN_u)r = 0 \Rightarrow AN_x + BN_u = 0$$

Alapjel miatti korrekció megvalósítása

- a követelmények az alábbi egyenletekben összegezhetőek:

$$AN_x + BN_u = 0$$
$$CN_x = I$$

- mátrix alakban az egyenletek:

$$\begin{bmatrix} A & B \\ C & 0 \end{bmatrix} \begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} A & B \\ C & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Alapjel miatti korrekció – példa

- az előző előadás „Ackermann-formula – Matlab példa” mintarendszerét használva, a szakasz

```
A=[-0.3929 0.1786 0.1429; 0.5 0 0; 0 0.25 0];
B=[1;0;0];
C=[0 0 -0.7143];
D=0;
```

- N_x és N_u számítható

```
N=inv([A B; C D])*[0;0;0;1];
Nx=N(1:3)
Nx =
 0
 0
-1.4000
Nu = N(4)
Nu =
 0.2000
```

2. Kimeneti visszacsatolás állapotmegfigyelő használatával

2.1. Állapotmegfigyelő megvalósítása

2.2. Állapotmegfigyelő – példa

Állapotmegfigyelő megvalósítása

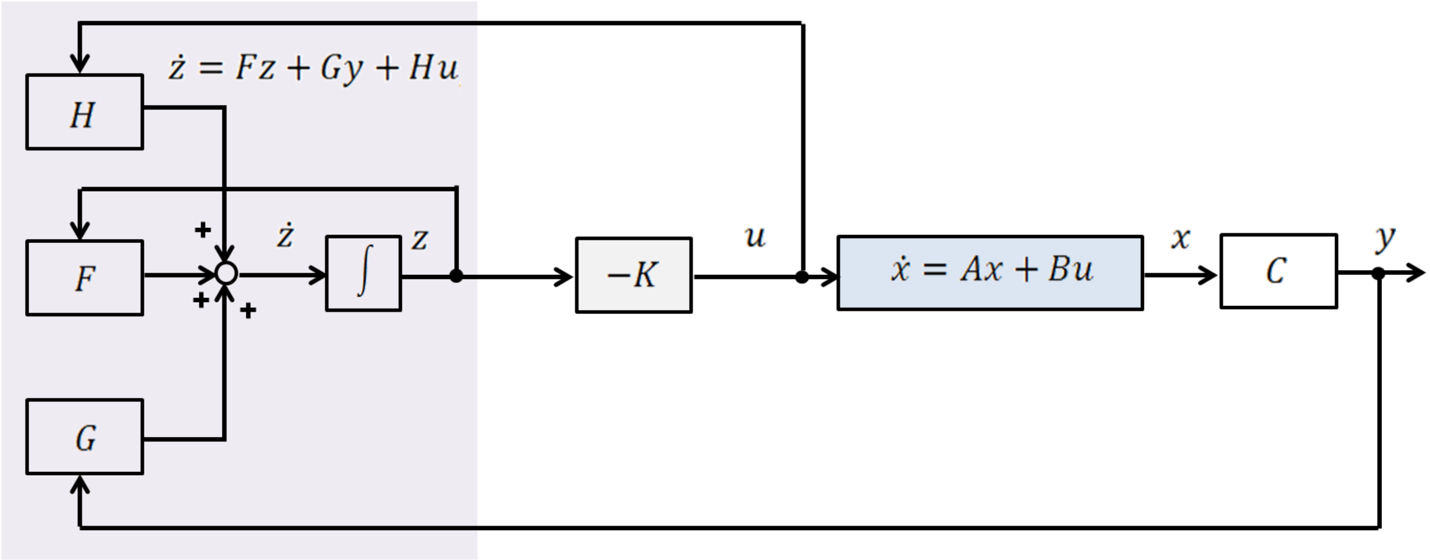
- egy erős hátránya az állapotvisszacsatolásnak, hogy az összes állapotváltozó értéke ismert kell, hogy legyen a visszacsatolás meghatározásához
- valós rendszerek esetén ez *extrém drága* (vagy *lehetetlen*)
- hasznos lehet egy olyan tagot használni, amely
  - ✓ meg tudja becsülni az állapotváltozók értékeit adott bemenetek és kimenetek mellett
  - ✓ biztosítani tudja, hogy megfelelően pontos modell esetén a becsült értékek a valósakhoz tartanak ➡

Állapotmegfigyelő

- egy dinamikus rendszer, melynek állapotai a becsült állapotváltozók
- az állapotmegfigyelő állapotegyenlete (a szabályozó által megvalósítva):

$$\dot{z} = Fz + Gy + Hu$$

Állapotmegfigyelő megvalósítása



• legyen a becslés hibája $\tilde{x} = x - z$, akkor a $H = B$ választással élve a hiba tranziense az alábbi differenciál-egyenlettel írható le:

$$\dot{\tilde{x}} = F\tilde{x} = (A - GC)\tilde{x}$$

ahol az F mátrix sajátértékei tetszőlegesen megválaszthatók egy megfelelő G választással élve, ha a (A,C) pár megfigyelhető

• az F mátrix sajátértékeinek elhelyezése algebrailag hasonló a pólusáthelyezés problémájához ➡ Ackermann formula használható

• ebben az esetben az A és C mátrix transzponáltjait kell behelyettesíteni

Állapotmegfigyelő – példa

- folytatva a korábbi példát, ahol a zárt kör K erősítése az alábbi volt:

```
K = acker(A,B,[-2.5000 -1.2500 -0.7143])
```

```
Gt = acker(A',C',[-5.0000 -2.5000 -1.4286])
Gt =
-120.9577 -111.7226 -11.9500

G = Gt'
G =
-120.9577
-111.7226
-11.9500
```

- szeretnénk olyan tranzienst előírni az állapotmegfigyelőnek, amely gyorsabb, mint a zárt kör tranziense

→ így az állapotmegfigyelő sajátértékeit a bal félsíkon távolabb kell elhelyezni, mint a zárt körét

Állapotmegfigyelő – példa

- miután kiszámítottuk az állapotmegfigyelő G mátrixának az értékét az Ackermann képlettel, az F mátrix is meghatározható:

```
F=A-G*C
F =
 -0.3929     0.1786  -86.2555
  0.5000          0  -79.8018
       0     0.2500   -8.5357
```

- végezetül pedig az állapotmegfigyelő H mátrixát úgy választjuk meg, hogy a szabályozandó rendszer B mátrixával legyen egyenlő:

```
H=B
H =
     1
     0
     0
```

3. LQ optimális szabályozás

Az LQ szabályozási probléma

- a rendszer egy LTI állapotteres (state space) modell

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

- a szabályozás célja: állapotvisszacsatolás tervezése a rendszer stabilizálására

$$u = -Kx$$

  ✓ K megtervezése egy kompromisszum a bementi és kimeneti előírások teljesítésére

- az optimális szabályozás megvalósításához költségfüggvényt definiálunk

$$J(x,u) = \frac{1}{2}\int_0^{\infty} [x^T(t)Qx(t) + u^T(t)Ru(t)]dt$$

  ✓ Q egy n x n szimmetrikus pozitív definit mátrix $Q^T = Q, Q > 0$
  ✓ R egy m x m szimmetrikus pozitív definit mátrix $R^T = R, R > 0$

➡ azt az u szabályozó jelet keressük, ami minimalizálja a J költségfüggvényt

Az LQ szabályozási probléma megoldása

- a folytonos idejű lineáris rendszer állapottérben leírva

$$\dot{x} = Ax + Bu$$

- a költségfüggvény az alábbi

$$J(x,u) = \frac{1}{2}\int_0^{\infty} [x^T(t)Qx(t) + u^T(t)Ru(t)]dt$$

- a visszacsatolás, amely a költségfüggvényt minimalizálja:

$$u = -Kx$$

  - ✓ ahol K a következő: $K = R^{-1}B^TP$

- P a folytonos idejű Control Algebraic Riccati Equation (CARE) megoldásából származik

$$PA + A^TP - PBR^{-1}B^TP + Q = 0$$

Az LQ szabályozási probléma megoldása Matlabbal

lqr

Linear-Quadratic Regulator (LQR) design

```
[K,S,e] = lqr(SYS,Q,R,N)
```

- az optimális K mátrixot számítja
- egy folytonos idejű rendszerre az u = −Kx szabályozó jel minimalizálja az alábbi költségfüggvényt $J(u) = \int_0^\infty (x^T Q x + u^T R u + 2x^T N u) dt$
- a szabályozandó szakasz $\dot{x} = Ax + Bu.$
- a visszacsatoló K mellett az lqr még a Ricatti egyenlet S megoldását is megadja $A^T S + SA - (SB + N) R^{-1} (B^T S + N^T) + Q = 0$
- és a zárt rendszer pólusait (e) = eig(A-B*K)
- K mátrix az S megoldásból számítható $K = R^{-1}(B^T S + N^T)$
- amikor N nincs megadva, akkor N = 0.

Az LQ szabályozás tulajdonságai

Statikus visszacsatolás

- az LQ módszer egy statikus K visszacsatoló mátrixot ad meg, ami nem egy dinamikus rendszer

⇒ a zárt rendszer fokszáma ugyanaz marad, ami a szakaszé volt

Robusztusság

- az LQ végtelen amplitúdótartalékot biztosít

$$Gm_{LQR} = \infty$$

- az LQ a fázis tartalékot is garantálja

$$\varphi_t > 60°$$

A Q és R súlyozó mátrix tipikus megválasztása

$$Q = \begin{bmatrix} q_1 & & & \\ & q_2 & & \\ & & \ddots & \\ & & & q_n \end{bmatrix} \qquad R = \rho \begin{bmatrix} r_1 & & & \\ & r_2 & & \\ & & \ddots & \\ & & & r_m \end{bmatrix}$$

$$q_i = \frac{1}{t_{s,i}(x_{i,max})^2} \qquad r_i = \frac{1}{(u_{i,max})^2} \qquad \rho > 0$$

$t_{s,i}$ az előírt beállási ideje az x_i állapotváltozónak

$x_{i,max}$ az $|x_i|$ megszorítása (limitje)

$u_{i,max}$ az $|u_i|$ megszorítása (limitje)

Q az állapotváltozókra hat, a rendszer minőségi jellemzőit súlyozza

R a szabályozó jelre hat, a bemenetet bünteti

Ez egy lehetséges megválasztás, de nem minden esetben így választjuk meg a súlyozó mátrixokat. A megválasztás az elvégzendő feladat specifikációjától függ!

LQ optimális szabályozás – példa

- szakasz

$$W(s) = \frac{A}{(1+sT_1)(1+sT_2)(1+sT_3)}$$

$A = 5$
$T_1 = 7$
$T_2 = 4$
$T_3 = 2$

```
% szakasz
Aplant=5; T1=7; T2=4; T3=2;
num=Aplant;
den=conv(conv([T1 1],[T2 1]),[T3 1]);
sys_tf=tf(num,den)
```

- state space modell

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

```
% áttérés ss-be
sys_ss=ss(sys_tf)
A=sys_ss.a;
B=sys_ss.b;
C=sys_ss.c;
D=sys_ss.d;
```

LQ optimális szabályozás – példa

Tervezzünk LQ szabályozót a $W(s)$ szakaszhoz. A súlyozó mátrixokat az alábbiak szerint válasszuk meg:

$$Q = \begin{bmatrix} q_1 & & & \\ & q_2 & & \\ & & \ddots & \\ & & & q_n \end{bmatrix} \qquad R = \rho \begin{bmatrix} r_1 & & & \\ & r_2 & & \\ & & \ddots & \\ & & & r_m \end{bmatrix}$$

$$q_i = \frac{1}{t_{s,i}(x_{i,max})^2} \qquad r_i = \frac{1}{(u_{i,max})^2}$$

$$x_{1,max} = 0.8$$
$$x_{2,max} = 0.4$$
$$x_{3,max} = 0.1$$
$$t_{s,1} = t_{s,2} = t_{s,3} = 12$$
$$u_{max} = 0.6$$

Hasonlítsuk össze az eredményeket különböző ρ értékekre.

$$\rho_1 = 1 \qquad \rho_2 = 0.1 \qquad \rho_3 = 0.01$$

LQ optimális szabályozás – példa

A súlyozó mátrixok számítása:

$$x_{1,max} = 0.8$$
$$x_{2,max} = 0.4$$
$$x_{3,max} = 0.1$$
$$t_{s,1} = t_{s,2} = t_{s,3} = 12$$
$$u_{max} = 0.6$$
$$\rho_1 = 1$$

$$q_1 = \frac{1}{t_{s,1}(x_{1,max})^2} = \frac{1}{12 \cdot 0.8^2} = 0.1302$$
$$q_2 = \frac{1}{t_{s,2}(x_{2,max})^2} = \frac{1}{12 \cdot 0.4^2} = 0.5208$$
$$q_3 = \frac{1}{t_{s,3}(x_{3,max})^2} = \frac{1}{12 \cdot 0.1^2} = 8.3333$$
$$Q = \begin{bmatrix} q_1 & & \\ & q_2 & \\ & & q_3 \end{bmatrix} = \begin{bmatrix} 0.1302 & & \\ & 0.5208 & \\ & & 8.3333 \end{bmatrix}$$
$$r = \frac{1}{(u_{max})^2} = \frac{1}{0.6^2} = 2.7778$$
$$R = \rho_1 \cdot r = 1 \cdot 2.7778 = 2.7778$$

LQ optimális szabályozás – példa

A súlyozó mátrixok számítása – Matlab kóddal:

```
% adott értékek
x_1max = 0.8;
x_2max = 0.4;
x_3max = 0.1;
t_s1 = 12;
t_s2 = 12;
t_s3 = 12;
u_max = 0.6;
```

```
% rho
rho = 1;
%rho = 0.1;
%rho = 0.01;
```

```
% Q elemeinek számítása
q1 = 1 / (t_s1 * x_1max^2);
q2 = 1 / (t_s2 * x_2max^2);
q3 = 1 / (t_s3 * x_3max^2);
```

```
% Q definiálása
Q = [q1 0 0
     0 q2 0
     0 0 q3]
```

LQ optimális szabályozás – példa

A súlyozó mátrixok és a szabályozó számítása – Matlab kóddal:

```
% R elemeinek számítása
r = 1 / (u_max^2); % egy bemenetünk van
```

```
% R definiálása
R  = rho * r
```

```
[K_LQ, P, eig_cl] = lqr(A, B, Q, R, 0);
% K_LQ a visszacsatoló mátrix
% P a Riccati egyenlet megoldása
% eig_cl tartalmazza a zárt kör sajátértékeit
((A-BK) sajátértékei)
```

LQ optimális szabályozás – példa

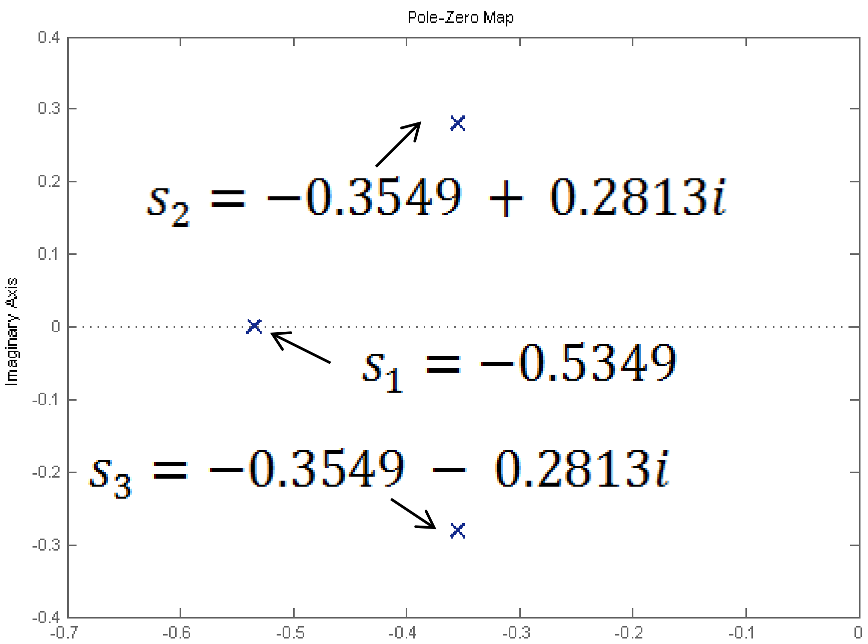
A szabályozó és a zárt kör

```
display('K mátrix:')
K_LQ
```

$$K_{LQ} = [0.3519 \quad 0.7054 \quad 1.4697]$$

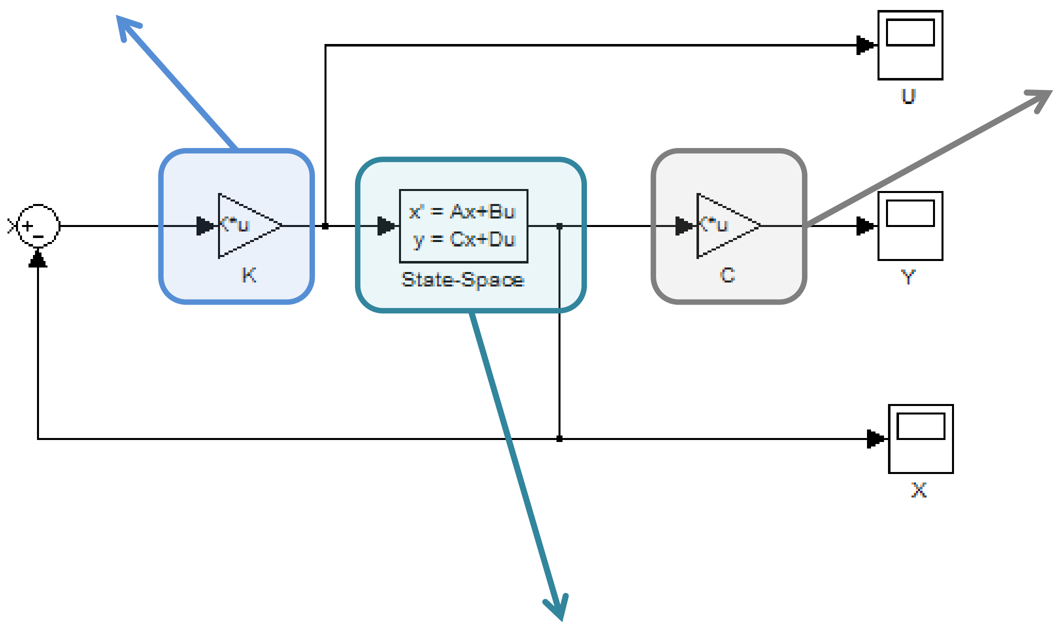
```
% sajátértékek numerikusan
display('A zárt kör
sajátértékei:')
eig_cl

% sajátértékek grafikusan
W_cl = ss(A-B*K_LQ, B, C, D);
pzmap(W_cl)
```



LQ optimális szabályozás – példa

A szabályozó a Matlabban `lqr` parancscsal számított K.

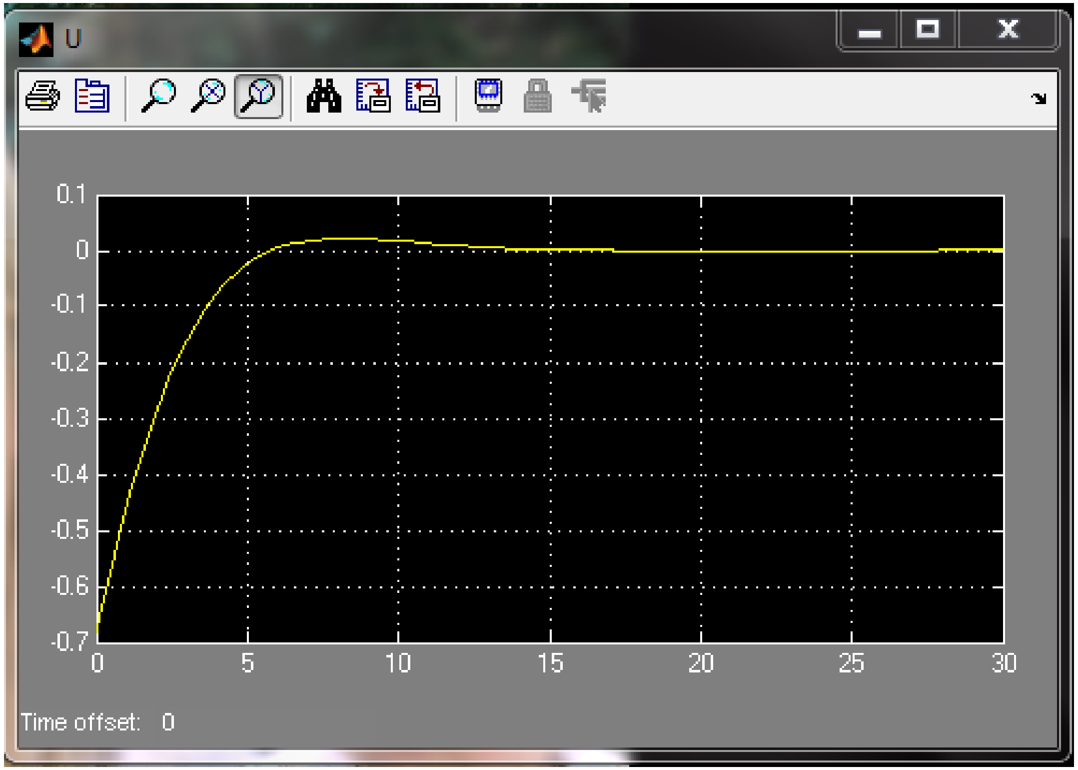


Az állapotokból ezután egy külön C mátrix-szorzással határozzuk meg a kimenetet. Ez a C mátrix eredetileg a rendszer része!

A C mátrix értékét a state-space blokkban egységmátrixra állítjuk, hogy a blokk kimenetén az állapotváltozókat kapjuk meg.

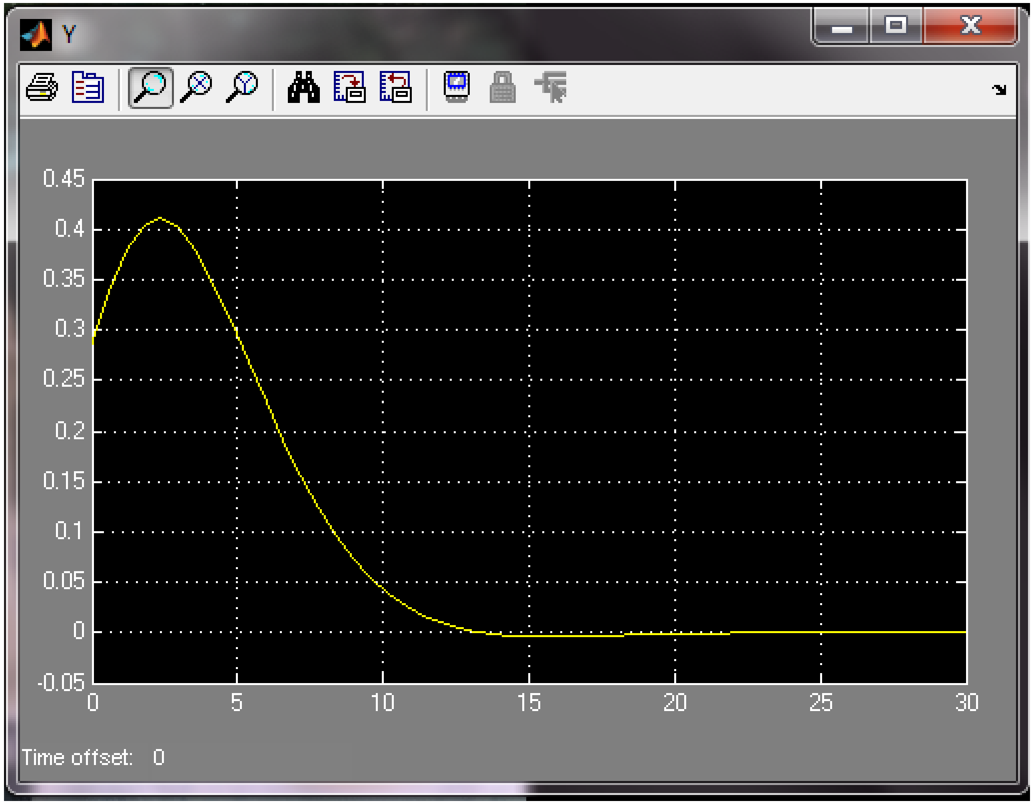
LQ optimális szabályozás – példa

A bemenet időfüggvénye



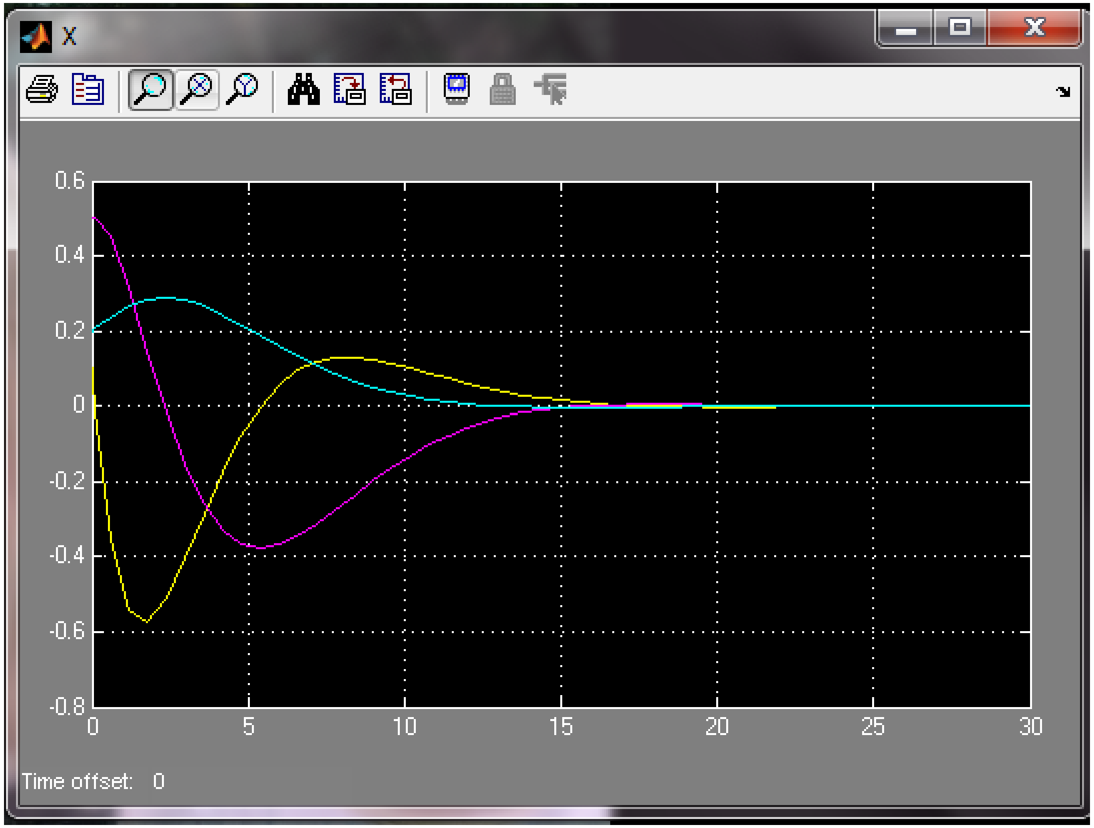
LQ optimális szabályozás – példa

A kimenet időfüggvénye



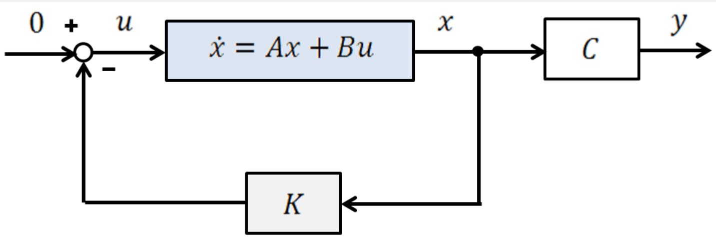
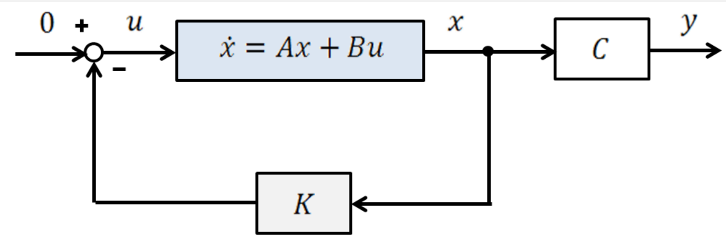
LQ optimális szabályozás – példa

Az állapotváltozók időfüggvénye



4. A pólusáthelyezés és az LQ szabályozás összehasonlítása

A pólusáthelyezés és az LQ szabályozás összehasonlítása

Pólusáthelyezés		LQ szabályozás
state space modell $\dot{x} = Ax + Bu$ $y = Cx$	szakasz	state space modell $\dot{x} = Ax + Bu$ $y = Cx$
	blokk diagram	
$u = -Kx$	szabályozó jel	$u = -Kx$
Ackermann-formula $K = [0 \quad 0 \quad \dots \quad 1] M_c^{-1} \varphi_c(A)$	megoldás	$K = R^{-1} B^T P$ Control Algebraic Riccati Equation $P \rightarrow PA + A^T P - PBR^{-1} B^T P + Q = 0$
a zárt rendszer pólusai s_i	megválasztandó paraméterek	súlyozó mátrixok Q, R

Videó

Pólusáthelyezés és LQ szabályozás összehasonlítása

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