

## Állapotteres modellen alapuló szabályozótervezés

A klasszikus irányítási algoritmusok zöme a folyamatok ki-bemeneti modellje (átviteli függvények) alapján határozzák meg az irányítási feladatot megoldó szabályozót. Ezek a szabályozók (a kaszkád szabályozó kivételével) csak a folyamat kimenetét veszik figyelembe a beavatkozó jel meghatározásánál. Az állapotteres modellekkel komplexebb viselkedésű rendszerek is leírhatók. Ezen modelleken alapuló szabályozók a folyamat belső állapotai is felhasználják a beavatkozó jel meghatározásánál. Így komplexebb irányítási feladatok, mint például nem polinomiális alapjel követése nulla állandósult állapotbeli hibával vagy nemlinearitást is tartalmazó rendszer irányítása, is megoldhatóak.

Az állapotteres modell  $n$  darab egymással csatolt elsőfokú differenciálegyenlettel írja le a folyamat viselkedését. Mindegyik differenciálegyenlet az egyik állapot változását adja meg az összes állapot és bemenetek függvényében:

$$\begin{aligned}\dot{\underline{x}} &= A\underline{x} + B\underline{u} \\ \underline{y} &= C\underline{x} + D\underline{u}\end{aligned}\tag{9.1}$$

$n$  az állapotok ( $\underline{x}$ ),  $m$  a bemenetek ( $\underline{u}$ ),  $p$  a kimenetek ( $\underline{y}$ ) száma. Az  $A$ ,  $B$ ,  $C$ ,  $D$  mátrixok a folyamat paramétereit tartalmazzák, definiálva a folyamat dinamikus viselkedését. Az  $A$  mátrixot állapotmátrixnak is nevezzük.

Mintavételes állapotteres modell a rendszer  $k+1$ -ik mintavételbeli állapotát határozza meg a  $k$ -ik mintavételbeli állapot és a  $k$ -ik mintavételbeli bemenet függvényében. A kimeneti egyenletben a  $k$ -ik mintavételbeli kimenetet a  $k$ -ik mintavételbeli állapot és  $k$ -ik mintavételbeli bemenet alapján írható fel. Lineáris rendszerek esetén az állapotteres modell:

$$\begin{cases} \underline{x}_{k+1} = \Phi \underline{x}_k + \Gamma \underline{u}_k \\ \underline{y}_k = C \underline{x}_k + D \underline{u}_k \end{cases}\tag{9.2}$$

A  $\Phi$ ,  $\Gamma$ ,  $C$ ,  $D$  mátrixok a folytonos rendszer paramétereitől valamint a mintavételi periódustól függenek.

### ***1.1. Egy bemenetű egy kimenetű folyamatok állapotteres modelljének irányíthatósági formája***

Az irányíthatósági formában felírt állapotteres modellt az irányítási algoritmus tervezésénél alkalmazzuk. A standard irányíthatósági alakban felírt modellre könnyű meghatározni a folyamat irányítását megvalósító szabályozót.

Az irányítási alak meghatározásához induljunk ki a rendszer átviteli függvényéből. Feltételezzük, hogy a rendszer relatív fokszáma  $l$ :

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_{n-1}s^{n-1} + \dots + b_1s + b_0}{s^n + a_{n-1}s^{n-1} + \dots + a_1s + a_0} \quad (9.3)$$

Vezessük be a  $\zeta(s)$  segédváltozót az alábbi formában:

$$\xi(s) = \frac{1}{A(s)}U(s) \quad (9.4)$$

Tehát áttérve időtartományba:

$$\begin{aligned} A(s) \cdot \xi(s) &= U(s) \\ \xi^{(n)}(t) + a_{n-1}\xi^{(n-1)}(t) + \dots + a_1\xi'(t) + a_0\xi(t) &= u(t) \end{aligned} \quad (9.5)$$

Ugyancsak a  $\zeta(s)$  definíciója alapján kapjuk:

$$\begin{aligned} Y(s) &= B(s) \cdot \zeta(s) \\ y(t) &= b_{n-1}\xi^{(n-1)}(t) + \dots + b_1\xi'(t) + b_0\xi(t) \end{aligned} \quad (9.6)$$

Mivel a rendszer fokszáma  $n$ , az állapotvektor dimenziója is  $n$ . Definiáljuk a rendszer állapotainak deriváltjait az alábbi formában:

$$\begin{aligned} \dot{x}_1(t) &= \xi^{(n)}(t) = -a_{n-1}\xi^{(n-1)}(t) - \dots - a_1\xi'(t) - a_0\xi(t) + u(t) \\ \dot{x}_2(t) &= \xi^{(n-1)}(t) = x_1 \\ &\vdots \\ \dot{x}_n(t) &= \xi^{(1)}(t) = x_{n-1} \end{aligned} \quad (9.7)$$

Ennek alapján kapjuk a rendszer állapotterezs modelljét irányíthatósági alakban:

$$\begin{aligned} \dot{\underline{x}} &= \begin{pmatrix} -a_{n-1} & -a_{n-2} & \dots & -a_1 & -a_0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix} \underline{x} + \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} u \\ y &= (b_{n-1} \quad b_{n-2} \quad \dots \quad b_1 \quad b_0) \underline{x} \end{aligned} \quad (9.8)$$

## 1.2. Pólusáthelyezés állapotvisszacsatolással

Pólusáthelyezéssel a szabályozott rendszernek előírt tranziens viselkedést lehet biztosítani. Állapotterezs modelleknél az  $A$  mátrix sajátértékei definiálják a rendszer tranziens viselkedését. A szabályozási feladat tranziens viselkedésének előírásaként a szabályozott rendszer sajátértékeit írjuk elő. Ehhez első lépésként a kívánt tranziens

viselkedést biztosító karakterisztikus polinomot szükséges meghatározni, amelynek gyökei lesznek az előírt sajátértékek.

A pólusát helyezés feladata: a beavatkozó jelet (a rendszer  $u$  bemenete) határozzuk meg úgy az állapotok függvényében, hogy a visszacsatolt zárt rendszer sajátértékei az előírtak legyenek.

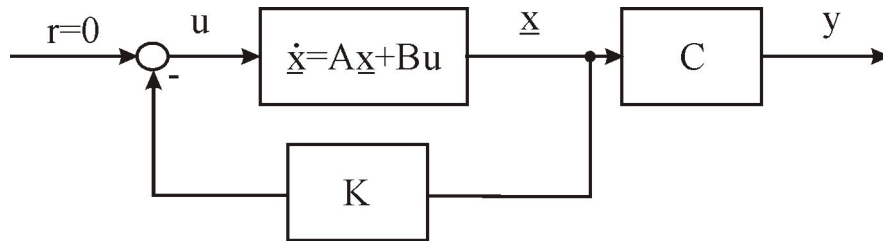
Egy bemenetű egy kimenetű rendszerek esetén a beavatkozó jelet az alábbi módon számíthatjuk:

$$\begin{aligned}
 u &= -K\underline{x} \\
 K &= (k_{n-1} \quad k_{n-2} \quad \dots \quad k_0) \quad \underline{x} = (x_1 \quad x_2 \quad \dots \quad x_n)^T \\
 u &= k_{n-1}x_1 + k_{n-2}x_2 + \dots + k_0x_n
 \end{aligned}
 \tag{9.9}$$

A  $K$  sorvektor konstans értékeket (erősítéseket) tartalmaz. A beavatkozó jel az állapotok lineáris kombinációja.

Az irányított rendszer az állapotvisszacsatolással:

$$\begin{aligned}
 \dot{\underline{x}} &= A\underline{x} + B u; \quad u = -K\underline{x} \\
 \dot{\underline{x}} &= (A - BK)\underline{x}
 \end{aligned}
 \tag{9.10}$$



0.1 Ábra: Állapotvisszacsatolás kialakítása

**9.1 Példa:** Írjuk fel harmadfokú szabályozási rendszer esetén a zárt rendszer  $A$  állapotmátrixát.

Legyen a rendszer állapotváltozását leíró modell:  $\dot{\underline{x}} = A\underline{x} + B u$

A beavatkozó jel számítása  $n=3$  -ra:

$$\underline{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \quad K = (k_2 \quad k_1 \quad k_0) \quad u = -(k_2 \quad k_1 \quad k_0) \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = -k_2x_1 - k_1x_2 - k_0x_3 \tag{9.11}$$

A zárt rendszer állapotmátrixa:

$$A - BK = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} - \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \begin{pmatrix} k_2 & k_1 & k_0 \end{pmatrix} = \begin{pmatrix} a_{11} - b_1 k_2 & a_{12} - b_1 k_1 & a_{13} - b_1 k_0 \\ a_{21} - b_2 k_2 & a_{22} - b_2 k_1 & a_{23} - b_2 k_0 \\ a_{31} - b_3 k_2 & a_{32} - b_3 k_1 & a_{33} - b_3 k_0 \end{pmatrix} \quad (9.12)$$

Látszik, hogy a zárt rendszer pólusai megváltoznak és függenek az állapotvisszacsatolás erősítésvektorától. A  $K$  vektor elemeinek helyes megválasztásával biztosíthatjuk, hogy a zárt rendszer sajátértékei az előírtak legyenek. Már a harmad fokú rendszer esetén is látszik, hogy általános esetben a feladatot megoldani nehéz.

### 1.2.1. Pólusáthelyezés feladata, ha a rendszer modellje irányíthatósági alakban van megadva

Feltételezzük, hogy az irányított folyamatot leíró modell a (9.8) irányíthatósági alakban van megadva.

**9.1 Lemma:** A (9.8) irányíthatósági alak  $A$  állapotmátrixában megjelenő  $a_0, a_1, \dots, a_{n-1}$  elemek egyenlőek a mátrix karakterisztikus polinomjának együtthatóival:

$$\det(\lambda I - A) = \lambda^n + a_{n-1}\lambda^{n-1} + a_{n-2}\lambda^{n-2} + \dots + a_1\lambda + a_0 \quad (9.13)$$

**9.2 Példa:** Igazoljuk a 9.1 Lemmát harmad fokú rendszer esetén.

Harmad fokú rendszer esetén az állapotmátrix:

$$A = \begin{pmatrix} -a_2 & -a_1 & -a_0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \quad (9.14)$$

A karakterisztikus polinom:

$$\begin{aligned} \det(\lambda I - A) &= \det \left( \begin{pmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{pmatrix} - \begin{pmatrix} -a_2 & -a_1 & -a_0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \right) = \det \begin{pmatrix} \lambda + a_2 & a_1 & a_0 \\ -1 & \lambda & 0 \\ 0 & -1 & \lambda \end{pmatrix} \\ &= (\lambda + a_2) \det \begin{pmatrix} \lambda & 0 \\ -1 & \lambda \end{pmatrix} + \det \begin{pmatrix} a_1 & a_0 \\ -1 & \lambda \end{pmatrix} = \lambda^2(\lambda + a_2) + a_1\lambda + a_0 = \lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 \end{aligned} \quad (9.15)$$

A mátrix sajátértékei (a karakterisztikus polinom gyökei) alapján a karakterisztikus polinom együtthatói egyértelműen meghatározhatóak. Legyenek az előírt, zárt rendszer karakterisztikus polinomjainak együtthatói:  $p_0, p_1, \dots, p_{n-1}$ . Ezzel a zárt rendszer állapotmátrixa irányíthatósági alakban:

$$\begin{pmatrix} -p_{n-1} & -p_{n-2} & \dots & -p_1 & -p_0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix} \quad (9.16)$$

Határozzuk meg úgy az állapotvisszacsatolás  $K = (k_{n-1} \ k_{n-2} \ \dots \ k_0)$  erősítésvektorát, hogy a zárt rendszer karakterisztikus polinomjának együtthatói az előírtak legyenek. Mivel a folyamat modellje irányíthatósági alakban van megadva, a visszacsatolt zárt rendszer:

$$A - BK = \begin{pmatrix} -a_{n-1} & -a_{n-2} & \dots & -a_1 & -a_0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \begin{pmatrix} k_{n-1} & k_{n-2} & \dots & k_0 \end{pmatrix} \quad (9.17)$$

$$\begin{pmatrix} -p_{n-1} & -p_{n-2} & \dots & -p_1 & -p_0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix} = \begin{pmatrix} -a_{n-1} - k_{n-1} & -a_{n-2} - k_{n-2} & \dots & -a_1 - k_1 & -a_0 - k_0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix}$$

Tehát ahhoz, hogy a zárt rendszer karakterisztikus polinomjának együtthatói az előírtak legyenek, az erősítésvektor elemei:

$$k_i = p_i - a_i$$

$$K = (p_{n-1} - a_{n-1} \ \dots \ p_1 - a_1 \ p_0 - a_0) \quad (9.18)$$

### 1.2.2. Az Ackermann formula

Általában a rendszerek állapotteres modellje nem irányíthatósági alakban van megadva. Keressük arra a választ, hogy hogyan kell megválasztani az állapotvisszacsatolás erősítésvektorát általános esetben úgy, hogy a zárt rendszer karakterisztikus polinomja az előírt legyen.

$$K = e_n^T \cdot M_C^{-1} \cdot \varphi_{Cref}(A) \quad (9.31)$$

A (9.31) összefüggés definiálja az állapotvisszacsatolást általános állapotteres modell esetén. Az összefüggés nem függ az irányíthatósági alaktól csak az általános rendszer állapotmátrixától ( $A$ ), irányíthatósági mátrixától, valamint az előírt karakterisztikus polinom együtthatóitól. Az  $e_n^T$  sorvektor kiemeli az  $M_C^{-1}$  irányíthatósági mátrix inverzének és a  $\varphi_{Cref}(A)$  mátrix összeszorzásából kapott mátrix utolsó sorát. Ez lesz a keresett erősítésvektor.

Ha a rendszer irányíthatósági mátrixa nem invertálható, tehát a rangja nem egyenlő a rendszer fokszámával, az erősítésvektor nem megvalósítható. Tehát ahhoz, hogy az irányítási feladatot megoldó erősítésvektort meg tudjuk határozni, a rendszer irányítható kell, hogy legyen.

Mivel a módszer algebrai, *mintavételes rendszerek* esetén is alkalmazható. Ebben az esetben a karakterisztikus polinom gyökeit az egységsugarú körön belül kell megválasztani:

$$K = e_n^T \cdot M_C^{-1} \cdot \varphi_{Cref}(\Phi) \quad (9.32)$$

A beavatkozó jelet a  $k$ -ik mintavételben a  $k$ -ik mintavételbeli állapot függvényében számítjuk:

$$u_k = -K \underline{x}_k \quad (9.33)$$

Az így számított beavatkozó jellel a zárt rendszer:

$$\underline{x}_{k+1} = \Phi \underline{x}_k - \Gamma u_k \Rightarrow \underline{x}_{k+1} = (\Phi - \Gamma K) \underline{x}_k \quad (9.34)$$

A (9.31) és (9.32) összefüggéseket, amilyekkel a zárt rendszernek előírt tranzienseket tudunk biztosítani, Ackermann formulának nevezzük. A gyakori alkalmazás miatt a Matlab programozási környezet Control könyvtára külön függvényt biztosít az erősítésvektor meghatározására, ez az 'acker' függvény.

### ***1.3. Az állapotvisszacsatolás megvalósítása állapotmegfigyelővel***

Az állapotvisszacsatolás feltételezi, hogy a rendszer összes állapota (a teljes  $\underline{x}$  állapotvektor) ismert, mérhető. Általában azonban csak a folyamat kimeneteit mérjük. Ezért fontos, hogy a kimenet, bemenet és a rendszer ismert modellje alapján meg tudjuk határozni, meg tudjuk becsülni az ismeretlen állapotokat. A beavatkozó jel kiszámításához pedig a becsült állapotát használjuk.

Jelölje  $\hat{\underline{x}}$  a becsült állapotok vektorát,  $\tilde{\underline{x}} = \underline{x} - \hat{\underline{x}}$  az állapotbecslési hibát. Az állapotmegfigyelő feladata, a becsült állapotok meghatározása minden időpillanatban úgy, hogy az állapotbecslési hiba nullába konvergáljon.

Keressük az állapotmegfigyelőt dinamikus rendszer formájában. Feltételezzük, hogy az becsült állapotokat egy olyan dinamikus rendszer generálja, amelynek a bemenetei az irányított folyamat  $u$  bemenete és  $y$  kimenete:

$$\frac{d\hat{\underline{x}}}{dt} = F \hat{\underline{x}} + Gy + Hu \quad (9.35)$$

Keressük úgy az állapotmegfigyelő paramétereit tartalmazó  $F$ ,  $G$ ,  $H$  mátrixokat úgy, hogy az állapotbecslési hiba nullába konvergáljon. Határozzuk meg az állapotbecslési hiba dinamikáját, feltételezve, hogy a folyamatunk dinamikáját a (9.1) összefüggés írja le  $D=0$  feltétel mellett:

$$\frac{d\tilde{x}}{dt} = \frac{dx}{dt} - \frac{d\hat{x}}{dt} = Ax + Bu - F\hat{x} - G \cdot C\underline{x} - Hu \quad (9.36)$$

$H = B$  választással kapjuk:

$$\frac{d\tilde{x}}{dt} = (A - G \cdot C) \cdot \underline{x} - F\hat{x} \quad (9.37)$$

$F=A-GC$  választással kapjuk:

$$\frac{d\tilde{x}}{dt} = F\hat{x} \quad (9.38)$$

Tehát már csak az  $y$ -t szorzó  $G$  oszlopvektort kell meghatározni. Vegyük észre, hogy ha a (9.38) autonóm rendszernek az összes sajátértékének valós része negatív, akkor a becslési hiba dinamikáját leíró rendszer stabil, tehát bármely kezdőállapotból a hibavektor összes eleme nullába konvergál. Tehát úgy kell megválasszuk a  $G$  oszlopvektort, hogy stabilá tegye az  $F=A-GC$  mátrixot.

Kihasználva, hogy egy mátrix sajátértéke egyenlő a transzponáltjának sajátértékeivel az  $F$  mátrix karakterisztikus polinomja:

$$\varphi(F) = \det(\lambda I - F) = \det(\lambda I - (A - G \cdot C)) = \det(\lambda I - (A^T - C^T \cdot G^T)) \quad (9.39)$$

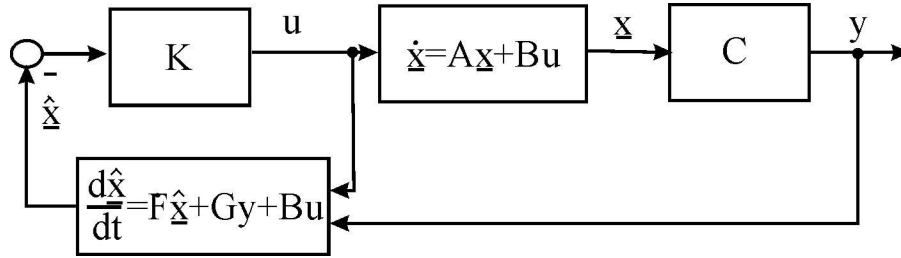
A feladat visszavezethető az állapotvisszacsatolással megvalósított pólusát helyezés problémájával: keressük úgy a  $G^T$  vektort, hogy az  $A^T, C^T$ -al definiált lineáris dinamikus rendszert stabilizálja. Vagyis a  $G^T$  erősítésvektorral megvalósított állapotvisszacsatolással az  $A^T - C^T G^T$  zárt rendszer stabil legyen, előírt tranzienssel. Ehhez alkalmazni lehet a (9.31) Ackermann formulát. Mivel  $A, B$  helyett  $A^T, C^T$ -ra kell alkalmazni a (9.31) képletet, az Ackermann formulában az irányíthatósági mátrix helyett a megfigyelhetőségi mátrixot kell alkalmazni. Ugyanakkor előírunk egy karakterisztikus polinomot ( $\varphi_{Cref}$ ) az  $F$ -nek. A megfigyelő  $G$  vektora:

$$G^T = e_n^T \cdot M_O^{-1} \cdot \varphi_{Cref}(A^T) \quad (9.40)$$

Ahhoz, hogy a  $G$  vektort meg tudjuk határozni a megfigyelhetőségi mátrix invertálható kell legyen, a rendszer megfigyelhető kell legyen.

Az így meghatározott állapotmegfigyelő generálja a becsült állapotokat, aminek függvényében számítjuk ki a beavatkozó jelet (lásd 9.2 Ábra):

$$\begin{cases} \frac{d\hat{x}}{dt} = F\hat{x} + Gy + Bu \\ u = K\hat{x} \end{cases} \quad (9.41)$$



0.2 Ábra: Állapotviszacsatolás állapotmegfigyelővel

Mintavételes esetben keressük az állapotmegfigyelőt az alábbi formában:

$$\hat{\underline{x}}_i = F \cdot \hat{\underline{x}}_{i-1} + G \cdot y_i + H \cdot u_{i-1} \quad (9.42)$$

Az állapotbecslési hiba változása:

$$\begin{aligned} \tilde{\underline{x}}_i &= \underline{x}_i - \hat{\underline{x}}_i = \Phi \cdot \underline{x}_{i-1} + \Gamma \cdot u_{i-1} - G \cdot C \cdot \underline{x}_i - F \cdot \hat{\underline{x}}_{i-1} - H \cdot u_{i-1} = \\ &= \Phi \cdot \underline{x}_{i-1} + \Gamma \cdot u_{i-1} - G \cdot C \cdot (\Phi \cdot \underline{x}_{i-1} + \Gamma \cdot u_{i-1}) \underline{x}_i - F \cdot \hat{\underline{x}}_{i-1} - H \cdot u_{i-1} = \\ &= F \cdot (\underline{x}_{i-1} - \hat{\underline{x}}_{i-1}) + (\Phi - G \cdot C \cdot \Phi - F) \underline{x}_{i-1} + (\Gamma - G \cdot C \cdot \Gamma - H) u_{i-1} \end{aligned} \quad (9.43)$$

Válasszuk az állapotmegfigyelő  $F$  és  $H$  mátrixait:

$$\begin{aligned} F &= \Phi - G \cdot C \cdot \Phi \\ H &= \Gamma - G \cdot C \cdot \Gamma \end{aligned} \quad (9.44)$$

Ezzel a választással az állapotbecslési hiba változása:

$$\tilde{\underline{x}}_i = F \cdot \tilde{\underline{x}}_{i-1} \quad (9.45)$$

A  $G$  vektort úgy kell meghatározni, hogy az  $F$  mátrix biztosítsa a (9.45) mintavételes rendszer stabilitását, vagyis azt, hogy  $F$  összes sajátértékének abszolút értéke kisebb legyen, mint 1. Ugyanúgy, mint folytonos esetben, írjuk fel  $F$  karakterisztikus polinomját:

$$\varphi(F) = \det(\lambda I - F) = \det(\lambda I - (\Phi - G \cdot C \cdot \Phi)) = \det(\lambda I - (\Phi^T - \Phi^T C^T \cdot G^T)) \quad (9.46)$$

Mintavételes esetben az Ackermann formulát a  $\Phi^T$ ,  $\Phi^T C^T$  mátrixokra kell alkalmazni. A karakterisztikus polinom előírásánál azt is figyelembe kell venni, hogy az állapotmegfigyelő válasza általában gyorsabb kell legyen, mint a zárt rendszer dinamikája, tehát a választott sajátértékek stabil és gyors választ kell biztosítsanak az állapotmegfigyelőnek.

Az állapotmegfigyelőkkel a szakirodalomban állapotbecslő néven is találkozhatunk.

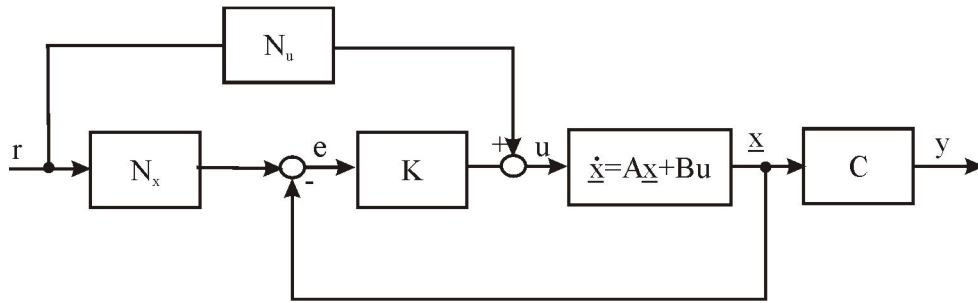


Az állapot megfigyelőket tervezhetünk más megközelítésben is. Elterjedtek tervezési módszerek még a minimális fokszerű Luneberger állapotmegfigyelő vagy a mérési zajokat is figyelembe vevő Kálmán szűrő.

### 1.4. Alapjel miatti korrekció

A pólusát helyezését megvalósító állapotvisszacsatolás csak az előírt tranzienseket biztosítja, nem biztosítja a nulla állandósult állapotbeli hibát. Ennek biztosítására előreccatoló ágat kell, hogy elhelyezzünk a szabályozóba. Keressük úgy az előreccatoló ágat, hogy egységugrás alapjelre ( $r$ ) állandósult állapotban garantálja a nulla állandósult állapotbeli hibát:  $y_\infty = r$ .  $y_\infty$  a kimenetet jelöli állandósult állapotban.

Helyezzük el a szabályozóba az előreccatolást, ahogy a 9.3 Ábrán látszik.



0.3 Ábra: Állapotvisszacsatolás alapjel miatti korrekcióval

Keressük úgy az  $N_x$ ,  $N_u$  mátrixokat, hogy

$$\begin{aligned} \underline{x}_\infty &= N_x \cdot r \\ u_\infty &= N_u \cdot r \end{aligned} \quad (9.47)$$

A (9.47) összefüggésből és hogy  $e_\infty = N_x r - x_\infty = 0$  következik, hogy az  $y_\infty = r$  feltételhez szükséges, hogy:

$$N_x \cdot r = \underline{x}_\infty \Rightarrow y_\infty = C \cdot N_x \cdot r \Rightarrow C \cdot N_x = I \quad (9.48)$$

Állandósult állapotban  $\dot{x}_\infty = 0$  és  $e_\infty = 0$ , tehát:

$$\begin{aligned} N_u \cdot r = u_\infty \Rightarrow A \underline{x}_\infty + B u_\infty &= A \cdot N_x \cdot r + B \cdot N_u \cdot r = (A \cdot N_x + B \cdot N_u) r = 0 \\ A \cdot N_x + B \cdot N_u &= 0 \end{aligned} \quad (9.49)$$

Az  $N_x$ ,  $N_u$  mátrixok az alábbi összefüggés alapján következnek:

$$\begin{cases} C \cdot N_x = I \\ A \cdot N_x + B \cdot N_u = \underline{0} \end{cases} \quad (9.50)$$

A (9.50) összefüggést mátrixegyenlet alakban felírva, kapjuk a megoldást:

$$\begin{pmatrix} A & B \\ C & 0 \end{pmatrix} \cdot \begin{pmatrix} N_x \\ N_u \end{pmatrix} = \begin{pmatrix} 0 \\ I \end{pmatrix} \Rightarrow \begin{pmatrix} N_x \\ N_u \end{pmatrix} = \begin{pmatrix} A & B \\ C & 0 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 0 \\ I \end{pmatrix} \quad (9.51)$$

Mintavételes esetben az  $N_x$ ,  $N_u$ -t ugyanolyan konfigurációban helyezzük el a szabályozóban, mint folytonos esetben. A meghatározásukhoz a követelményeket ugyancsak a (9.47) összefüggés adja. Mintavételes szabályozási rendszereknél  $\underline{x}_{\infty i+1} = \underline{x}_{\infty i}$ , tehát:

$$\begin{aligned} y_\infty &= C \cdot N_x \cdot r \Rightarrow C \cdot N_x = I \\ \underline{x}_{\infty i+1} = \underline{x}_{\infty i} &= \Phi \underline{x}_{\infty i} + \Gamma u_{\infty i} \Rightarrow (\Phi N_x + N_u \Gamma) r = N_x \cdot r \Rightarrow (\Phi - I) N_x + \Gamma N_u = 0 \end{aligned} \quad (9.52)$$

Ennek alapján mintavételes esetben az előreccsatolás súlyzómatrixait az alábbi összefüggés adja:

$$\begin{pmatrix} \Phi - I & \Gamma \\ C & 0 \end{pmatrix} \cdot \begin{pmatrix} N_x \\ N_u \end{pmatrix} = \begin{pmatrix} 0 \\ I \end{pmatrix} \Rightarrow \begin{pmatrix} N_x \\ N_u \end{pmatrix} = \begin{pmatrix} \Phi - I & \Gamma \\ C & 0 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 0 \\ I \end{pmatrix} \quad (9.53)$$

Az alapjel miatti korrekcióval kibővített állapotviszacsatolás az előírt tranziensek mellett konstans alapjelre a nulla állandósult állapotbeli hibát is garantálja.

### 1.5. Terhelésbecslés

Számos irányítástechnikai alkalmazásnál bemenetre ható zajra (terhelésre) számíthatunk, amely állandósult állapotbeli hibához vezethet. Az  $d$  bemeneti zaj általában additívan módosítja a bementet, így a rendszer bemenete  $u$  helyett  $u+d$  lesz. A  $d$  értéke általában ismeretlen.

Ennek kiküszöbölésére, kompenzálására a szabályozót módosítani kell. Állapotteres szabályozóknál megoldás lehet az állapotmegfigyelő módosítása úgy, hogy a bemenetre ható zajt is megbecsülje majd az így kapott becült zajértékkel kompenzáljuk a bemeneti zaj hatását. Feltételezzük, hogy a bemenetre ható  $d$  zaj konstans, vagy csak nagyon lassan változik:  $\dot{d} \cong 0$ . Ha az ismeretlen bemeneti zajt a rendszer új állapotának tekintjük

( $x_d = d$ ), az állapotok változását leíró modell módosul:

$$\begin{cases} \dot{\underline{x}} = A \underline{x} + B(u + x_d) \\ \dot{x}_d = 0 \end{cases} \quad (9.54)$$

A rendszer teljes állapotteres modellje a bementi zajjal bővített állapotvektorral:

$$\begin{cases} \begin{pmatrix} \dot{\underline{x}} \\ \dot{\underline{x}}_d \end{pmatrix} = \underbrace{\begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix}}_{\tilde{A}} \cdot \underbrace{\begin{pmatrix} \underline{x} \\ \underline{x}_d \end{pmatrix}}_{\tilde{\underline{x}}} + \underbrace{\begin{pmatrix} B \\ 0 \end{pmatrix}}_{\tilde{B}} u \\ y = \underbrace{(C \quad 0)}_{\tilde{C}} \cdot \begin{pmatrix} \underline{x} \\ \underline{x}_d \end{pmatrix} \end{cases} \quad (9.55)$$

A (9.55) összefüggésben bevezetett jelölésekkel a módosított állapotterezes modell:

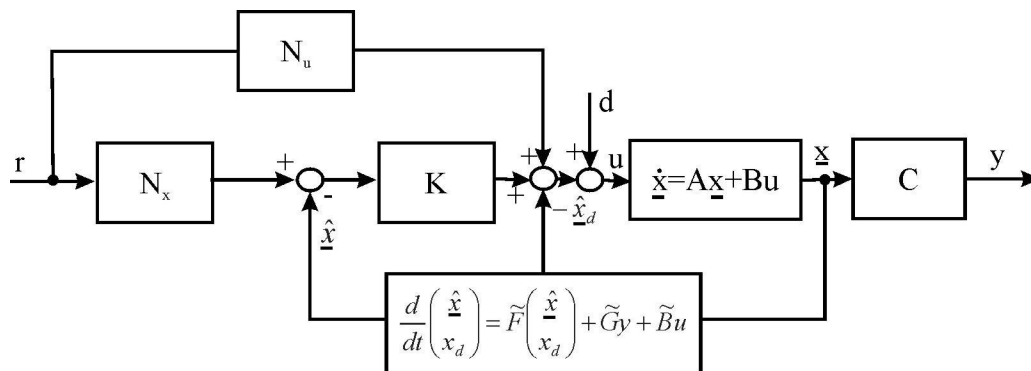
$$\begin{cases} \dot{\tilde{\underline{x}}} = \tilde{A} \cdot \tilde{\underline{x}} + \tilde{B} \cdot u \\ y = \tilde{C} \cdot \tilde{\underline{x}} \end{cases} \quad (9.56)$$

Amennyiben a (9.56) rendszerre tervezünk állapotmegfigyelőt, az ismeretlen bemeneti zajt is meg fogja becsülni. Az állapotvisszacsatolást ugyanakkor az eredeti, nem bővített rendszerre tervezzük. Az irányítás alatt a szabályozó által kiszámított beavatkozó jelből ( $u_C$ ) kivonjuk a becsült zajértéket. Amennyiben az állapotmegfigyelőnk jól működik ( $\hat{x}_d \equiv d$ ), a folyamat bemenete  $u = u_C - \hat{x}_d + d \equiv u_C$ . Tehát az új állapottal kompenzálhatjuk az additív bemeneti zaj hatását, a beavatkozó jel nem torzul.

*Mintavételes esetben* ugyancsak az állapotmegfigyelőt kell kibővíteni, az eljárás ugyanaz, mint folytonos esetben. A kibővített folytonos modellt kell mintavételes alakra hozni. Ehhez például a Zero Order Hold módszert alkalmazhatjuk. Használható a Matlab 'c2dm' függvénye 'zoh' paraméterezéssel.

$$\tilde{A} = \begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix} \quad \tilde{B} = \begin{pmatrix} B \\ 0 \end{pmatrix} \quad \tilde{C} = (C \quad 0) \quad \xrightarrow{\text{"zoh"}} (\tilde{\Phi}, \tilde{\Gamma}, \tilde{C}) \quad (9.57)$$

A kibővített mintavételes modellre tervezzük meg az állapotbecslést.



**0.4 Ábra: Állapotvisszacsatolás megvalósítása állapotmegfigyelővel, terhelésbecslővel, alapjel miatti korrekcióval**

## 1.6. Állapotteres szabályozók bővítése integráló taggal

Közismert, hogy a szabályozóban elhelyezett integrátor tag javítja a szabályozó kör alapjel követési tulajdonságát, ugyanakkor a bemeneti zajokat is képes kompenzálni. Amennyiben az alapjel konstans, az integrátor garantálja a nulla állandósult állapotbeli hibát, konstans bemeneti zaj hatását pedig képes teljesen kompenzálni.

Bővítsük ki úgy a szabályozónkat, hogy a visszacsatolásban jelenjen meg a kimenet integrálja ( $x_I$ ).

$$x_I = \int y dt \Rightarrow \dot{x}_I = y = C \cdot \underline{x} \quad (9.58)$$

A szabályozó kibővítéséhez a kimenet integrálját helyezzük el úgy az állapotteres modellben, mint egy új állapotot:

$$\underbrace{\begin{pmatrix} \dot{x} \\ \dot{x}_I \end{pmatrix}}_{\dot{\tilde{x}}} = \underbrace{\begin{pmatrix} A & 0 \\ C & 0 \end{pmatrix}}_{\tilde{A}} \cdot \underbrace{\begin{pmatrix} x \\ x_I \end{pmatrix}}_{\tilde{x}} + \underbrace{\begin{pmatrix} B \\ 0 \end{pmatrix}}_{\tilde{B}} u \quad (9.59)$$

Tervezzük meg az állapotvisszacsatolást a  $\tilde{\dot{x}} = \tilde{A} \cdot \tilde{x} + \tilde{B} \cdot u$  bővített rendszerre. Az így kapott állapotvisszacsatolással a beavatkozó jel tartalmazza a kimenet integrálját is:

$$u = -\tilde{K} \cdot \tilde{x} = -(K \quad K_I) \begin{pmatrix} x \\ x_I \end{pmatrix} = -(K \quad K_I) \left( \int y dt \right) \quad (9.60)$$

A kibővített állapotvisszacsatolás tartalmazza az integrátoros szabályozások előnyeit (jó alapjelkövetés, zajelnyomás). Amennyiben a szabályozóban állapotbecslést is szükséges alkalmazni, azt nem a bővített, hanem az eredeti rendszerre kell megtervezni.

Mintavételes rendszereknél ugyanígy járunk el: az állapotok vektorát bővítjük a kimenet integráljának mintavételes megközelítésével, majd a kibővített állapotvektort csatoljuk vissza. Az integrátor mintavételes megközelítéséhez alkalmazhatunk téglalapszabályt:

$$x_I = \int y dt \quad x_{I_{i+1}} = x_{I_i} + T \cdot y_i = x_{I_i} + T \cdot C \cdot \underline{x}_i \quad (9.61)$$

$T$  a mintavételi periódust jelöli.

A kibővített mintavételes rendszer, amelyre az állapotvisszacsatolást tervezzük:

$$\begin{cases} \begin{pmatrix} \underline{x}_{i+1} \\ x_{I_{i+1}} \end{pmatrix} = \begin{pmatrix} \Phi & 0 \\ T \cdot C & 0 \end{pmatrix} \cdot \begin{pmatrix} \underline{x}_i \\ x_{I_i} \end{pmatrix} + \begin{pmatrix} \Gamma \\ 0 \end{pmatrix} u \\ y_i = (C \quad 0) \cdot \begin{pmatrix} \underline{x}_i \\ x_{I_i} \end{pmatrix} \end{cases} \quad (9.62)$$

A kibővített rendszer állapotvisszacsatolásának tervezéséhez az Ackermann formulát alkalmazhatjuk.

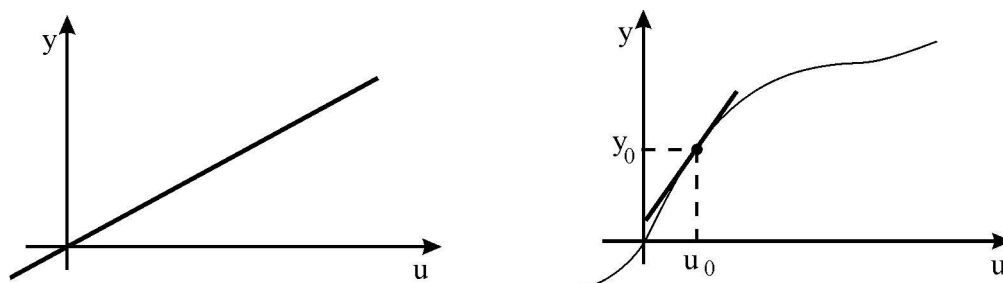
## *Lineáris irányítástechnikai módszerek kiterjesztése nemlineáris rendszerekre*

Amennyiben a nemlineáris rendszer állapotai az állapottér egy szűk tartományában tartózkodnak, a folyamatot leíró nemlineáris modell sok esetben megközelíthető lineáris modellel. Ebben a fejezetben megkeressük a folyamat nemlineáris modelljét megközelítő lineáris modellt, amikor az állapotok az állapottér egy pontja környékén mozognak (*munkapont*). Az így kapott lineáris modellel tervezünk szabályozót, amely alkalmazható a nemlineáris rendszer irányítására, amennyiben a rendszer állapotai a munkapont környékén mozognak.

### 10.1.1 Munkapont körüli linearizálás

Legyen a (10.1) modellel leírt nemlineáris rendszer. Amennyiben a rendszer stabil akkor egy  $\underline{u}_0$  konstans bemenetre a rendszer állapota a tranziensek lecsengése után  $\underline{x}_0$  lesz. Az  $\underline{x}_0$  állapotnak és  $\underline{u}_0$  bemenetnek megfelel az  $\underline{y}_0$  kimenet. Az  $\underline{u}_0$ ,  $\underline{x}_0$ ,  $\underline{y}_0$ , vektorok definiálják a rendszer munkapontját. Például ha egy egyenáramú motor bemenete konstans feszültség, akkor a motor állandósult állapotban konstans szögsebességgel forog és a rotoron átfolyó áram szintén konstans lesz. A konstans bemeneti feszültség és a neki megfelelő áram és szögsebesség definiálják a rendszer munkapontját.

A 10.5 ábrán egy stabil lineáris rendszer és egy stabil nemlineáris rendszer statikus átvitele látható egy bemenet egy kimenet esetén. Lineáris rendszer esetén az átvitelt egy (általában a zérón áthaladó) egyenes írja le, vagyis állandósult állapotban bármely stabil lineáris rendszer egy erősítő, az erősítése az egyenes meredeksége. Nemlineáris esetben az átvitelt valamilyen görbe írja le. Ha egy munkapontban húzunk a görbéhez egy érintőt, akkor ezzel az érintővel a munkapont körül a görbe jól approximálható. Minél közelebb vagyunk a munkaponthoz, annál jobb az approximáció.



**0.5 Ábra: Stabil lineáris rendszer átvitele. Stabil nemlineáris rendszer átvitele és linearizálása munkapont körül**

Az egyenessel történő approximációt Taylor sorba fejtéssel valósíthatjuk meg. A módszert először egy bemenetű egy kimenetű statikus nemlinearitásra mutatjuk be, majd kiterjesztjük többváltozós nemlineáris dinamikus rendszerekre. Ahhoz, hogy a Taylor sorba fejtést alkalmazni lehessen, a nemlineáris függvény deriválható kell legyen.

Legyen az  $y=f(u)$  nemlineáris statikus átvitel. Linearizáljuk az  $u_0, y_0$  munkapont körül. Az  $f$  függvény Taylor sorba fejtése:

$$y = f(u_0) + \frac{\partial f}{\partial u} \Big|_{u=u_0} (u - u_0) + \underbrace{\frac{\partial^2 f}{\partial u^2} \Big|_{u=u_0} (u - u_0)^2 + \frac{\partial^3 f}{\partial u^3} \Big|_{u=u_0} (u - u_0)^3 + \dots}_{\Delta^2(u)} \quad (10.19)$$

A másodfokú és magasabb kitevőjű tagokat ( $\Delta^2$ ) elhanyagoljuk. Az így kapott  $y = a \cdot u + b$  egyenes formája és paraméterei:

$$y = \underbrace{\frac{\partial f(u_0)}{\partial u}}_a \cdot u + \underbrace{f(u_0) - \frac{\partial f(u_0)}{\partial u} \cdot u_0}_b \quad (10.20)$$

**10.4 Példa:** Fejtsük Taylor sorba  $u_0=0$  pont körül az  $y = f(u) = \frac{1}{1 + e^{-u}}$  nemlinearitást.

Az  $f$  deriváltja:  $\frac{\partial f(u)}{\partial u} = \frac{e^{-u}}{(1 + e^{-u})^2}$ , a függvény értéke az  $u_0=0$  pontban:  $\frac{\partial f(0)}{\partial u} = \frac{1}{4}$

Alkalmazva a (10.20) összefüggést, kapjuk:  $y = f(0) + \frac{\partial f}{\partial u} \Big|_{u=0} (u - 0) = \frac{1}{4}u + \frac{1}{2}$

Terjesszük ki a módszert a (10.1) modellel leírt  $m$  bemenetű,  $n$  állapotú,  $p$  kimenetű nemlineáris dinamikus rendszerre. A linearizáláshoz szükséges, hogy az  $f$  és  $h$  nemlinearitások deriválhatóak legyenek. A rendszer (10.1) modellje kifejtve:

$$\begin{cases} \dot{x}_1 = f_1(x_1, x_2, \dots, x_n, u_1, u_2, \dots, u_m) \\ \dot{x}_2 = f_2(x_1, x_2, \dots, x_n, u_1, u_2, \dots, u_m) \\ \vdots \\ \dot{x}_n = f_n(x_1, x_2, \dots, x_n, u_1, u_2, \dots, u_m) \\ y_1 = h_1(x_1, x_2, \dots, x_n, u_1, u_2, \dots, u_m) \\ y_2 = h_2(x_1, x_2, \dots, x_n, u_1, u_2, \dots, u_m) \\ \vdots \\ y_p = h_p(x_1, x_2, \dots, x_n, u_1, u_2, \dots, u_m) \end{cases} \quad (10.21)$$

Legyen a munkapont, ami körül linearizálni szeretnénk:

$$\begin{aligned} \underline{y}_0 &= (y_{10}, \dots, y_{p0}) \\ \underline{x}_0 &= (x_{10}, \dots, x_{n0}) \\ \underline{u}_0 &= (u_{10}, \dots, u_{m0}) \end{aligned} \quad (10.22)$$

A rendszer linearizálásához a Taylor sorbefejtés többváltozós függvényekre kidolgozott formáját alkalmazhatjuk, elhanyagolva a másod- és magasabb fokú komponenseket:

$$\begin{aligned}
 f_1(\underline{x}, \underline{u}) &= f_1(\underline{x}_0, \underline{u}_0) + \frac{\partial f_1(\underline{x}_0, \underline{u}_0)}{\partial x_1} (x_1 - x_{10}) + \dots + \frac{\partial f_1(\underline{x}_0, \underline{u}_0)}{\partial x_n} (x_n - x_{n0}) + \\
 &\quad + \frac{\partial f_1(\underline{x}_0, \underline{u}_0)}{\partial u_1} (u_1 - u_{10}) + \dots + \frac{\partial f_1(\underline{x}_0, \underline{u}_0)}{\partial u_m} (u_m - u_{m0}) \\
 &\vdots
 \end{aligned} \tag{10.23}$$

$$\begin{aligned}
 f_n(\underline{x}, \underline{u}) &= f_n(\underline{x}_0, \underline{u}_0) + \frac{\partial f_n(\underline{x}_0, \underline{u}_0)}{\partial x_1} (x_1 - x_{10}) + \dots + \frac{\partial f_n(\underline{x}_0, \underline{u}_0)}{\partial x_n} (x_n - x_{n0}) + \\
 &\quad + \frac{\partial f_n(\underline{x}_0, \underline{u}_0)}{\partial u_1} (u_1 - u_{10}) + \dots + \frac{\partial f_n(\underline{x}_0, \underline{u}_0)}{\partial u_m} (u_m - u_{m0})
 \end{aligned}$$

$$\begin{aligned}
 h_1(\underline{x}, \underline{u}) &= h_1(\underline{x}_0, \underline{u}_0) + \frac{\partial h_1(\underline{x}_0, \underline{u}_0)}{\partial x_1} (x_1 - x_{10}) + \dots + \frac{\partial h_1(\underline{x}_0, \underline{u}_0)}{\partial x_n} (x_n - x_{n0}) + \\
 &\quad + \frac{\partial h_1(\underline{x}_0, \underline{u}_0)}{\partial u_1} (u_1 - u_{10}) + \dots + \frac{\partial h_1(\underline{x}_0, \underline{u}_0)}{\partial u_m} (u_m - u_{m0}) \\
 &\vdots
 \end{aligned} \tag{10.24}$$

$$\begin{aligned}
 h_p(\underline{x}, \underline{u}) &= h_p(\underline{x}_0, \underline{u}_0) + \frac{\partial h_p(\underline{x}_0, \underline{u}_0)}{\partial x_1} (x_1 - x_{10}) + \dots + \frac{\partial h_p(\underline{x}_0, \underline{u}_0)}{\partial x_n} (x_n - x_{n0}) + \\
 &\quad + \frac{\partial h_p(\underline{x}_0, \underline{u}_0)}{\partial u_1} (u_1 - u_{10}) + \dots + \frac{\partial h_p(\underline{x}_0, \underline{u}_0)}{\partial u_m} (u_m - u_{m0})
 \end{aligned}$$

Mátrixos formában felírva a linearizált rendszermodellt:

$$\begin{aligned}
 \begin{pmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{pmatrix} &= \begin{pmatrix} f_1(\underline{x}, \underline{u}) \\ \vdots \\ f_n(\underline{x}, \underline{u}) \end{pmatrix} = \begin{pmatrix} f_1(\underline{x}_0, \underline{u}_0) \\ \vdots \\ f_n(\underline{x}_0, \underline{u}_0) \end{pmatrix} + \begin{pmatrix} \frac{\partial f_1(\underline{x}_0, \underline{u}_0)}{\partial x_1} & \dots & \frac{\partial f_1(\underline{x}_0, \underline{u}_0)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(\underline{x}_0, \underline{u}_0)}{\partial x_1} & \dots & \frac{\partial f_n(\underline{x}_0, \underline{u}_0)}{\partial x_n} \end{pmatrix} \begin{pmatrix} x_1 - x_{10} \\ \vdots \\ x_n - x_{n0} \end{pmatrix} \\
 &\quad + \begin{pmatrix} \frac{\partial f_1(\underline{x}_0, \underline{u}_0)}{\partial u_1} & \dots & \frac{\partial f_1(\underline{x}_0, \underline{u}_0)}{\partial u_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n(\underline{x}_0, \underline{u}_0)}{\partial u_1} & \dots & \frac{\partial f_n(\underline{x}_0, \underline{u}_0)}{\partial u_m} \end{pmatrix} \begin{pmatrix} u_1 - u_{10} \\ \vdots \\ u_m - u_{m0} \end{pmatrix}
 \end{aligned} \tag{10.25}$$

$$\begin{pmatrix} y_1 \\ \vdots \\ \vdots \\ \vdots \\ y_p \end{pmatrix} = \begin{pmatrix} h_1(\underline{x}, \underline{u}) \\ \vdots \\ \vdots \\ \vdots \\ h_p(\underline{x}, \underline{u}) \end{pmatrix} = \begin{pmatrix} h_1(\underline{x}_0, \underline{u}_0) \\ \vdots \\ \vdots \\ \vdots \\ h_p(\underline{x}_0, \underline{u}_0) \end{pmatrix} + \begin{pmatrix} \frac{\partial h_1(\underline{x}_0, \underline{u}_0)}{\partial x_1} & \cdots & \frac{\partial h_1(\underline{x}_0, \underline{u}_0)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p(\underline{x}_0, \underline{u}_0)}{\partial x_1} & \cdots & \frac{\partial h_p(\underline{x}_0, \underline{u}_0)}{\partial x_n} \end{pmatrix} \begin{pmatrix} x_1 - x_{10} \\ \vdots \\ \vdots \\ \vdots \\ x_n - x_{n0} \end{pmatrix} \\
+ \begin{pmatrix} \frac{\partial h_1(\underline{x}_0, \underline{u}_0)}{\partial u_1} & \cdots & \frac{\partial h_1(\underline{x}_0, \underline{u}_0)}{\partial u_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p(\underline{x}_0, \underline{u}_0)}{\partial u_1} & \cdots & \frac{\partial h_p(\underline{x}_0, \underline{u}_0)}{\partial u_m} \end{pmatrix} \begin{pmatrix} u_1 - u_{10} \\ \vdots \\ \vdots \\ \vdots \\ u_m - u_{m0} \end{pmatrix} \quad (10.26)$$

Vezessük be az alábbi jelöléseket:

$$A = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix} \Bigg|_{\substack{\underline{x}=\underline{x}_0 \\ \underline{u}=\underline{u}_0}} \quad B = \begin{pmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_m} \\ \frac{\partial f_n}{\partial u_1} & \cdots & \frac{\partial f_n}{\partial u_m} \end{pmatrix} \Bigg|_{\substack{\underline{x}=\underline{x}_0 \\ \underline{u}=\underline{u}_0}} \quad (10.27)$$

$$C = \begin{pmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial x_1} & \cdots & \frac{\partial h_p}{\partial x_n} \\ \frac{\partial h_p}{\partial x_1} & \cdots & \frac{\partial h_p}{\partial x_n} \end{pmatrix} \Bigg|_{\substack{\underline{x}=\underline{x}_0 \\ \underline{u}=\underline{u}_0}} \quad D = \begin{pmatrix} \frac{\partial h_1}{\partial u_1} & \cdots & \frac{\partial h_1}{\partial u_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial u_1} & \cdots & \frac{\partial h_p}{\partial u_m} \\ \frac{\partial h_p}{\partial u_1} & \cdots & \frac{\partial h_p}{\partial u_m} \end{pmatrix} \Bigg|_{\substack{\underline{x}=\underline{x}_0 \\ \underline{u}=\underline{u}_0}} \quad (10.28)$$

Az  $A$ ,  $B$ ,  $C$ ,  $D$  Jacobi mátrixok definiálják a linearizált rendszer dinamikáját. A linearizált dinamikus rendszer modelljének felírásához vezessük be az alábbi állapotot:

$$\tilde{\underline{x}} = \underline{x} - \underline{x}_0 \quad (10.29)$$

Mivel  $\underline{x}_0, \underline{u}_0$  konstansak, az  $\tilde{\underline{x}}$  állapot változása:

$$\dot{\tilde{\underline{x}}} = \dot{\underline{x}} - \dot{\underline{x}}_0 = \dot{\underline{x}} - f(\underline{x}_0, \underline{u}_0) \quad (10.30)$$

Ugyanígy legyen az új kimenet és bemenet:

$$\tilde{\underline{u}} = \underline{u} - \underline{u}_0 \quad \tilde{\underline{y}} = \underline{y} - \underline{y}_0 \quad (10.31)$$

Felhasználva a (10.25)-(10.31) összefüggéseket, jelöléseket az alábbi modellt kapjuk:



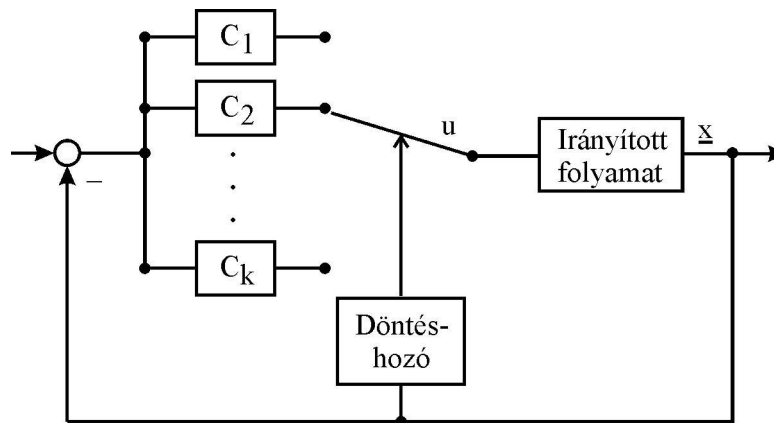
$$\begin{cases} \dot{\underline{\tilde{x}}} = A \cdot \underline{\tilde{x}} + B \cdot \underline{\tilde{u}} \\ \underline{\tilde{y}} = C \cdot \underline{\tilde{x}} + D \cdot \underline{\tilde{u}} \end{cases} \quad (10.32)$$

A fenti lineáris dinamikus rendszerre lineáris szabályozót tervezhetünk, amellyel az eredeti nemlineáris rendszert a munkapont közelében irányíthatjuk. Vegyük észre, hogy ha a (10.32) rendszert stabilizáljuk, a rendszer kimenete és állapotai a munkapontba fognak konvergálni:  $\underline{\tilde{x}} \rightarrow 0 \Rightarrow \underline{x} \rightarrow \underline{x}_0$ .

### 10.1.2 Erősítésprogramozás

A munkapont körüli linearizálás technikája a munkapontba történő szabályozáshoz jó tranziens, pontos szabályozást biztosít. A munkapontra jól működő szabályozó jó eséllyel helyes működést mutat a munkapont környezetében is. Így a nemlineáris rendszer irányításához a következő stratégiát alkalmazhatjuk: A rendszer állapotterét felosztjuk, és mindegyik altérben felvesszünk egy munkapontot. A rendszert mindegyik munkapontban linearizáljuk és minden munkapontra egy-egy helyi szabályozót tervezünk. Amikor a rendszer az állapotter megfelelő alterében tartózkodik, akkor az altérben található munkaponthoz tervezett szabályozót alkalmazzuk a rendszer irányítására.

Az irányítás tömbvázlata a 10.6 ábrán látható. A mért állapotok függvényében a döntéshozó blokk meghatározza, hogy az állapotter melyik tartományában van a rendszer és kiválasztja az ennek megfelelő szabályozót. Az így kapott szabályozó a teljes állapotterben képes irányítani a rendszert.



0.6 Ábra: Erősítésprogramozást alkalmazó szabályozási rendszer

Az erősítésprogramozás módszerének hátránya a szabályozó komplexitása. Amennyiben az állapotteret sok apró tartományra osztjuk, akkor sok helyi szabályozóval kell számoljunk. Ez implementálási problémákhoz vezethet. Ha a rendszernek  $n$  állapota van és az  $i$ -ik állapot mentén  $k_i$  részre osztjuk a az állapotteret, akkor a döntéshozó  $k = k_1 \cdot k_2 \cdot \dots \cdot k_n$  szabályozó közül kell válasszon.