ML. A MATLAB néhány alkalmazása rendszeranalízisre

A MATLAB egy általános célú matematikai programcsomag, amely jól használható rendszeranalízisre is. A MATLAB egyes verziói némileg eltérőek. A következő rövid ismertetés keretében feltételezzük, hogy a „Control System Toolbox" szolgáltatásai rendelkezésre állnak. Más programcsomagok (például az OCTAVE) is nyújtnak hasonló szolgáltatásokat.

Feltételezzük, hogy az Olvasó ismeri a MATLAB használatának alapvető szabályait, habár a legtöbb rendszeranalízis-feladat megoldásához ezekre alig van szükség. Az alább megadott módon kapott eredményekből a MATLAB felhasználásával sok további, a rendszeranalízis szempontjából fontos következtetésre lehet jutni.

Az ismertetés során a beírandó MATLAB utasításokat **vastag betűkkel** *írjuk*, hogy a magyarázatoktól jól elkülönüljenek.

Az összefoglalás végén megadott néhány példát egyszerűnek választottuk, hogy az Olvasó az eredményt papír-ceruza módszerrel is ellenőrizhesse.

A továbbiakban a *rendszer* lineáris, invariáns, kauzális, egy-gerjesztésű, egy-válaszú (SISO) rendszert jelent. A diszkrét idejű (DI) rendszerre további megkötés nincs, a folytonos idejű (FI) rendszerről feltételezzük, hogy differenciális is.

A rendszer megadására, jellemzőinek meghatározására, adott gerjesztéshez tartozó válaszának számítására a MATLAB sok lehetőséget kínál. Az áttekinthetőség érdekében ezeknek csak egy részét ismertetjük. További lehetőségeket az Olvasó maga is értelmezhet vagy valamilyen **help** segítségével kaphat felvilágosítást.

A MATLAB egyes verziói szimbolikus műveletek elvégzésére és jelfolyam hálózatok analízisére is alkalmasak. A továbbiakban sem ezekre, sem sok más általánosítási lehetőségre (például sok-gerjesztésű, sok-válaszú rendszerek analízise) nem térünk ki. A lineáris rendszereket illetően a **help ltimodels** és más **help** felhasználásával kaphatunk felvilágosítást.

E függelék szakaszaira és pontjaira ML jelzéssel (például ML-1. vagy ML-1.2) hivatkozunk.

ML-1. A rendszer megadása

ML-1.1. Általános elvek

Lineáris, invariáns, kauzális DI és differenciális FI rendszer három módon adható meg úgy, hogy abból a rendszerre vonatkozó sok információ (például valamelyik rendszerjellemző függvény) vagy a másik két leírás előállítható legyen. Ezek az *állapotváltozós leírás* (**ss** „state space"), az *átviteli függvény* két polinom hányadosaként (**tr** „transfer function") vagy pólusaival és zérusaival (**zpk** „zero-pole-gain").

Célszerű a rendszerre egy vagy több betűből és számból álló *azonosítót* választani. A továbbiakban általában az **rsz** azonosítót használjuk vagy az **rd** illetve az **rc** azonosítót a DI illetve a FI jelleg hangsúlyozására. Egyes feladatok megoldhatók azonosító választása nélkül is, de ezt nem tárgyaljuk.

A rendszer impulzusválasza és átviteli karakterisztikája a három alak bármelyikéből meghatározható. Jóval nehezebbek a fordított feladatok. Ezekkel nem foglakozunk.

ML-1.2. Az állapotváltozós leírás

A DI és a FI rendszer *állapotváltozós leírásának* normálalakja

$$\mathbf{x}' = A\mathbf{x} + \mathbf{B}u, \quad y = \mathbf{C}^{\mathrm{T}}\mathbf{x} + Du.$$

Az állapotváltozós leírás MATLAB megadása (1. példa)

DI: **rd=ss(A,B,C,D,1)** FI: **rc=ss(A,B,C,D)**

Az **A,B,C,D** mátrixok beírhatók az **ss** utasításba, de megadhatjuk előre, és ekkor csak azonosítójukat kell beírni az utasításba.

A **ENTER** után megjelennek az állapotváltozós leírásban szereplő mátrixok könnyen azonosítható alakban.

ML-1.3. Az átviteli függvény

A DI és a FI rendszer racionális *átviteli függvénye* két polinom hányadosa

$$\mathrm{DI}: \quad H(z) = \frac{b_0 z^m + b_1 z^{m-1} + \cdots + b_{m-1} z + b_m}{a_0 z^n + a_1 z^{n-1} + \cdots + a_{n-1} z + a_n}, \quad a_0 \neq 0, m \leq n;$$

$$\mathrm{FI}: \quad H(s) = \frac{b_0 s^m + b_1 s^{m-1} + \cdots + b_{m-1} s + s_m}{a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n}, \quad a_0 \neq 0, m \leq n.$$

Szokásos az $a_0 = 1$ választás.

A két polinomot az együtthatókból alkotott **a** és **b** vektorokkal adjuk meg:

a = [a0 a1 ... an]; b = [b0 b1 ... bm];

A nulla értékű együtthatókat is be kell írni a megfelelő helyre. A **b** vektor tekinthető $n+1$ eleműnek is, amelynek első néhány eleme esetleg nulla.

Az átviteli függvény MATLAB megadása (2. példa)

DI: **rd=tf(b,a,1)**

FI: **rc=tf(b,a)**

Az **a,b** vektorok beírhatók a **tf** utasításba, de megadhatók előre, és ekkor csak azonosítójukat kell beírni az utasításba.

A **ENTER** után megjelenik az átviteli függvény képlete.

ML-1.4. Az átviteli függvény gyöktényezős alakja

A DI és a FI rendszer racionális *átviteli függvénye* gyöktényezős alakban

$$\text{DI}: \quad H(z)=K\frac{(z-z_1)(z-z_2)\cdots(z-z_m)}{(z-p_1)(z-p_2)\cdots(z-p_n)}, \quad m\le n;$$

$$\text{FI}: \quad H(s)=K\frac{(s-z_1)(s-z_2)\cdots(s-z_m)}{(s-p_1)(s-p_2)\cdots(s-p_n)}, \quad m\le n .$$

A p_i pólusokat és a z_i zérusokat egy-egy vektorba foglalva adjuk meg:

p=[p1 p2 ... pn]; z=[z1 z2 ... zm];

Kétszeres (többszörös) pólust vagy zérust kétszer (többször) kell megadni.

Az átviteli függvény MATLAB megadása (3. példa)

DI: **rd=zpk(z, p, K, 1)** FI: **rc=zpk(z, p, K)**

Az **z, p** vektorok és a **K** szám beírható a **zpk** utasításba, de megadhatók előre, és ekkor elegendő azonosítójukat beírni az utasításba.

A komplex pólus vagy zérus képzetes része (például $j4$) a következő módok bármelyikén megadható: **4j 4*j j*4 4i 4*i i*4.** A MATLAB a képzetes részt **4.0000i** alakban adja meg. Ha a képzetes rész nem szám, hanem egy jel (például **b**), akkor a * nem hagyható el.

A **ENTER** után megjelenik az átviteli függvény gyöktényezős alakjának képlete. A konjugált komplex párok szorzata egy valós együtthatós másodfokú kifejezést eredményez. Többszörös pólusok és zérusok a gyöktényező hatványaként jelentkezik.

ML-1.5. A rendszert jellemző adatok

Bármelyik utasítással (**ss, tf, zpk**) adtuk meg a például **rsz** azonosítójú rendszert, annak jellemző adatai (az állapotváltozós leírás mátrixai, az átviteli függvény számlálója és nevezője vagy zérusai és pólusai) a megfelelő

[A,B,C,D]=ssdata(rsz, 'v')
[b,a]= tfdata(rsz, 'v')
[z,p,K]= zpkdata(rsz, 'v')

utasítás és az **ENTER** hatására megjelennek. Az így adódó vektorok és mátrixok a továbbiakban definiált változókként kezelhetők.

Az átviteli függvény pólusainak **p** vektora és zérusainak **z** vektora a

p=pole(rsz), z=zero(rsz)

utasításokkal is meghatározhatók (4. példa). A komplex pólusok és zérusok algebrai alakban adódnak. Az abszolút értékek és a szögek (radiánban vagy fokban) az

rp=abs(p), fip=angle(p), fipfok= angle(p)*180/pi

utasításokkal kapjuk. Értelem szerinti a zérusok kezelése.

A rendszer átviteli függvényének pólus-zérus ábrája a

pzmap(rsz)

utasítással állítható elő.

A rendszer állapotmátrixának sajátértékei az **A** mátrix ismeretében az **eig(A)** vagy **la=eig(A)** utasítással adódnak. Utóbbi esetben **la** vektorként kezelhető.

ML-1.6. Rendszerlírás átalakítása

Az **ss, tf, zpk** utasítások bármelyikével megadott **rsz** azonosítójú rendszer egy másik leírásának képlete a

ss(rsz) tf(rsz) zpk(rsz)

utasítások egyikével **ENTER** hatására adódik. Ennek az alaknak is adhatunk azonosítót, például **rszl=ss(rsz).**

Mivel egy rendszer állapotváltozós leírása nem egyértelmű, ezért ismételt átalakításokkal rendszerint nem ugyanaz az állapotváltozós leírás adódik.

Mivel a rendszerre vonatkozó információk bármelyik alakból előállíthatók, ezért erre a szolgáltatásra ritkán van szükség.

A rendszerleírás *egyszerűsítése* egy olyan rendszer-leírást eredményez, amelyben nem szerepelnek a gerjesztés-válasz kapcsolatot nem befolyásoló állapotváltozók vagy megtörtént az átviteli függvény számlálójának és nevezőjének a közös gyöktényezővel történt egyszerűsítése. Ez a **minreal** („minimal realization") utasítással végezhető el. Ha **rsz** egy valamilyen módon **(ss, tf, zpk)** értelmezett rendszer, akkor

rm=minreal(rsz)

minimális számú állapotváltozót vagy minimális fokszámú átviteli függvényű, az előzővel megegyező gerjesztés-válasz kapcsolatú, **rm** azonosítójú rendszert értelmez. Az **ENTER** hatására annak az alaknak a képlete jelenik meg, amelyben **rsz** meg lett adva.

A minimalizált állapotváltozós leírás függhet attól, hogy **rsz** melyik leírásával lett megadva.

A kerekítési hibán belüli egyenlőség 10^{-8} nagyságrendű hibával értelmezett. Értelmezhető ennél nagyobb relatív hibahatár is.

A rendszerleírás egyszerűsítése csak akkor megengedett, ha ennek hatására nem tűnnek el instabilitást jelentő sajátértékek vagy pólusok. Ellenkező esetben célszerű az objektumot jobb modellel leírni.

ML-1.7. Diszkrét idejű szimuláció

Az **rc** azonosítójú FI rendszerhez a **c2d** („continuous to discrete") utasítás egy valamilyen elv szerint szimuláló DI rendszer rendel.

A $H_C(s)$ átviteli függvényhez T mintavételi periódusidővel a

$$H_D(z) = H_C(s), \quad s = \frac{2}{T}\frac{z-1}{z+1}$$

bilineáris transzformációval $H_D(z)$ átviteli függvényt rendelő **rd** azonosítójú rendszer a

rd=c2d(rc, T, 'tustin')

utasítással állítható elő. Ha **rc** állapotváltozós leírásával értelmezett, akkor az utasítás olyan DI állapotváltozós leírást eredményez, amelyhez a fenti átviteli függvényt tartozik (tehát nem a FI állapotváltozós leírást szimulálja valamilyen módszerrel).

Más szimulációs eljárásokat a **help c2d** ad meg.

ML-1.P. Példák

1. példa Állapotváltozós leírás megadása

A DI illetve egy FI rendszer állapotváltozós leírásának közös alakja

$$x_1' = -0{,}1x_1 - 0{,}25x_2 + 2u,\ x_2' = 0{,}8x_1 - 0{,}5x_2 + 4u;\ y = 0{,}2x_1 + 0{,}4x_2 .$$

A DI illetve a FI rendszer MATLAB megadásának egyik módszere

DI: **rd1=ss([-0.1 -0.25; 0.8 -0.5], [2; 4], [0.2 0.4], 0, 1)**

FI: **rc1=ss([-0.1 -0.25; 0.8 -0.5], [2; 4], [0.2 0.4], 0)**

A rendszerek megadásának egy másik módszere

A=[-0.1 -0.25; 0.8 -0.5]; B=[2; 4]; C=[0.2 0.4]; D=0;

DI: **rd2=ss(A, B, C, D, 1)**

FI: **rc2=ss(A, B, C, D)**

Az első alakban megadott rendszer mátrixainak értelmezése

DI: **[Ad1, Bd1, Cd1, Dd1]=ssdata(rd1, 'v')**

FI: **[Ac1, Bc1, Cc1, Dc1]=ssdata(rc1, 'v')**

A második módszerrel megadott **A, B, C, D** mátrixok adódnak.

A rendszermátrix sajátértékeit a

DI: **lad=eig(rd1)**

FI: **laf=eig(rc1)**

utasítással kapjuk. Ugyanezt adja az **eig(rd2)** illetve az **eig(rc2)** utasítás is. Ha **A** adott, akkor az **eig(A)** vagy a **la=eig(A)** utasítás is használható.

2. példa Átviteli függvény megadása polinomok hányadosaként
Diszkrét idejű illetve folytonos idejű rendszer átviteli függvénye

$$\mathrm{DI}:\ H_D(z)=\frac{2z+0{,}8}{z^2+0{,}6z+0{,}25} \qquad \mathrm{FI}:\ H_C(s)=\frac{2s+0{,}8}{s^2+0{,}6s+0{,}25}$$

A rendszer megadásának egyik módszere

DI: **rd3=tf([2, 0.8], [1 0.6 0.25], 1)**

FI: **rc3=tf([2, 0.8], [1 0.6 0.25])**

A rendszer megadásának egy másik módszere

b=[2, 0.8]; a=[1 0.6 0.25];

DI: **rd4=tf(b, a, 1)** FI: **rc4=tf(b, a)**

Az első alakban megadott rendszer számlálójának és nevezőjének értelmezése

DI: **[bd3, ad3]=tfdata(rd3, 'v')** FI: **[bf3, af3]=tfdata(rc3, 'v')**

A második módszernél megadott **b** és **a** vektorok adódnak.
Az átviteli függvény pólusai és zérusai a

DI: **pd=pole(rd3), zd=zero(rd3)** FI: **pc=pole(rc3), zc=zero(rc3)**

utasítással állítható elő. A pólusok és a zérusok közvetlenül meghatározhatók a rendszer másféle (például állapotváltozós) leírásából is.

3. példa Átviteli függvény megadása gyöktényezős alakban
Diszkrét idejű illetve folytonos idejű rendszer átviteli függvényének egyetlen zérusa $z_1=-0{,}4$, két pólusa $p_{1,2}=-0{,}3\pm \mathrm{j}0{,}4$, a kiemelt tényező $K=2$, vagyis

$$\mathrm{DI}:\ H_D(z)=2\,\frac{z+0{,}4}{(z+0{,}3-\mathrm{j}0{,}4)(z+0{,}3+\mathrm{j}0{,}4)}$$

$$\mathrm{FI}:\ H_C(s)=2\,\frac{s+0{,}4}{(s+0{,}3-\mathrm{j}0{,}4)(s+0{,}3+\mathrm{j}0{,}4)}$$

A rendszer megadásának egyik módszere

DI: **rd5=zpk(-0.4, [-0.3+j0.4, -0.3-j0.4], 1)**

FI: **rc5=zpk(-0.4, [-0.3+j0.4, -0.3-j0.4])**

A rendszer megadásának egy másik módszere

z=-0.4; p=[-0.3+j0.4, -0.3-j0.4]; K=2;

DI: **rd6=zpk(z, p, K, 1)**

FI: **rc6=zpk(z, p, K)**

Az **ENTER** hatására az átviteli függvény képletében nem a komplex gyöktényezők, hanem a szorzatuk, vagyis egy másodfokú kifejezés adódik. A pólusokat vagy a zérusokat (például ellenőrzés céljából) a **pole(rd6)** vagy a **zero(rd6)** utasítással kaphatjuk.

4. példa Pólusok és zérusok meghatározása
Határozzuk meg a folytonos idejű rendszer átviteli függvényének pólusait, ha

$$H(s)=\frac{2s+1}{s^2+s+0{,}25}.$$

Az 5.2. példa mintájára értelmezhetünk egy **rc** azonosítójú rendszert, majd alkalmazzuk a **p=pole(rc)** utasítást. Egy másik lehetőség

pole(tf([2 1], [1 1 0.25]) ENTER -0.5000 -0.5000

A nevezőnek –0,5 valóban kétszeres zérushelye. Az átviteli függvény azonban egyszerűsíthető $H(s)=2/(s+0.5)$ alakra, tehát a pólus csak egyszeres. Az átviteli függvény egyszerűsítése az ML-1.5. pontban leírt módon végezhető el (v.ö. 6. példa).

5. példa Rendszerleírás átalakítása
Az előző példákban vizsgált **rd1, rd2, ..., rd6** illetve **rc1, rc2, ..., rc6** azonosítójú rendszerek ekvivalensek abban az értelemben, hogy azonos gerjesztés-válasz kapcsolatú rendszereket írnak le, például impulzusválaszuk ugyanaz.

Az **rd1** megadása után az **rd7=tf(rd1)** illetve az **rd8=zpk(rd1)** utasítás az átviteli függvénynek azt az alakját adja, mint amelyet a 2. illetve a 3. példában megadtunk. Az **rd9=ss(rd7)** vagy az **rd10=ss(rd8)** utasítás azonban nem adja vissza az 1. példában megadott állapotváltozós leírást. Az utóbbiakból előállított átviteli függvény azonban megegyezik az **rd1** szerintivel.

Hasonló a helyzet a vizsgált hat FI rendszer különböző leírásainak kapcsolatával is.

***6. példa** Rendszer minimál reprezentációja
Két FI rendszer átviteli függvénye

$$H_a(s)=\frac{s+0{,}6}{s^2+1{,}1s+0{,}3},\quad H_b(s)=\frac{1}{s+0{,}5}.$$

Ha az együtthatók (esetünkben 0,6 illetve 1,1 és 0,3) pontosnak tekinthetők, akkor a két átviteli függvény egyenlő, vagyis ekvivalens rendszereket ír le.

A két rendszer MATLAB megadása

ra=tf([1 0.6], [1 1.1 0.3]); rb=tf(2, [1 0.5]);

Az **[Aa, Ba, Ca, Da]=ssdatsa(ra, 'v')** utasítás egy másodrendű rendszer állapotváltozós leírásának mátrixait adja, míg az **[Ab, Bb, Cb, Db]=ssdatsa(rb, 'v')** utasítás egy elsőrendűét.

Az **rc=minreal(ra)** utasítással értelmezett rendszer az **rb** rendszerrel ekvivalens, a MATLAB a $H_b(s)$ átviteli függvényt szolgáltaja.

Az **ra1=ss(Aa, Ba, Ca, Da)** rendszerből az **rb1=minreal(ra1)** utasítás egy elsőrendű, az **rb** és az **rc** rendszerrel ekvivalens rendszert ad. Ennek **Aa1, Ba1, Ca1, Da1** mátrixai az **rb** és **rc** rendszer mátrixaitól különbözhetnek.

Mivel mindegyik rendszer GV stabilis, ezért az egyszerűsítés elfogadható.

Ha viszont a két átviteli függvény

$$H_p(s)=\frac{s-0{,}6}{s^2-1{,}1s+0{,}3},\quad H_b(s)=\frac{1}{s+0{,}5},$$

akkor az **rq=minreal(rp)** utasítással értelmezett rendszer ekvivalens az **rb** rendszerrel. Az **rp** rendszer azonban csak akkor stabilis, ha az adatok pontosak, ezért az ekvivalencia kétséges. Az **impulse(rp)** ábrájából úgy tűnhet, hogy az impulzusválasz nullához tart, de az **impulse(rp, 100)** ábrájából az derül ki, hogy az impulzusválasz $t = 90$ után növekedni kezd. Az **rp** impulzusválaszát különféle más módon (például különböző időlépésekkel) számítva lényegesen eltérő eredmények adódhatnak. A $H_p(s)$ átviteli függvényt a megadottnál pontosabban kell ismerni.

7. példa Diszkrét idejű szimuláció
Az első három példában valamelyik módon megadott **rc** *azonosítójú FI rendszer diszkrét* idejű szimulátorának **rcs** azonosítója $T = 0{,}1$ mintavételi periódusidővel a bilineáris transzformációval a

rcs=c2d(rc, 0.1, 'tustin')

utasítással állítható elő. Ebből a szimulátor átviteli függvényét a **Hs=tf(rcs)** utasítással kaphatjuk meg. Az **abs(pole(rcs))** megmutatja, hogy a stabilis FI rendszer szimulátora is GV stabilis. A FI rendszernek és szimulátorának egyaránt két pólusa van, a szimuláció az egy zérus helyett kettőt ad (az egyik a $z = -1$ helyen).

ML-2. Impulzusválasz és ugrásválasz meghatározása

ML-2.1. Általános elvek

Az **ss, tf, zpk** utasítások bármelyikével értelmezett **rsz** azonosítójú DI illetve FI rendszer impulzusválaszának és ugrásválaszának ábrája vagy kijelölt időpontokban fellépő értékei a MATLAB felhasználásával többféleképpen is meghatározhatók. A FI rendszer impulzusválaszát és ugrásválaszát a MATLAB diszkrét idejű szimulációval határozza meg.

Ha a FI rendszer állapotváltozós leírásában $D \neq 0$, vagy az átviteli függvény számlálójának és nevezőjének fokszáma megegyezik (ezért $D \equiv \lim_{s\to\infty} H(s) \neq 0$), akkor a MATLAB által szolgáltatott impulzusválaszhoz még $D\,\delta(t)$ hozzáadandó. A MATLAB a Dirac-impuzust közvetlenül nem tudja kezelni.

Ha olyan átviteli függvényt adunk meg, amelyre a számláló fokszáma nagyobb a nevező fokszámánál, akkor a MATLAB hibát jelez.

ML-2.2. Az impulzusválasz meghatározása

Az **rd** illetve **rc** azonosítójú rendszer impulzusválaszának ***ábrája*** **például az**

DI: **impulse(rd)**

FI: **impulse(rd)**

utasítással határozható meg. Ekkor a $k_{\max}$ illetve a $t_{\max}$ értékét a MATLAB választja meg.

A vizsgálandó maximális L illetve T időpont elő is írható:

DI: **impulse(rd, L)**   FI: **impulse(rc, T)**

Az ábrából az értékek leolvashatók a **ginput, ENTER** utasítással. Ennek hatására fonálkereszt jelenik meg az ábrán, amelyet az egérrel a kívánt pontra vagy pontokra állítunk, majd kattintunk. **ENTER,** majd **ESC** után a képernyőn megjelennek a független és a függő változó vagy változók összetartozó értékei. Ez az eljárás más módon előállított ábrával is lehetséges.

Ez az utasítás a DI impulzusválaszt intervallumonként állandó (lépcsős) ábraként adja meg. Egy más módszert alább ismertetünk.

Az **rsz** azonosítójú rendszer impulzusválaszának numerikus *értékei* táblázatos formában a

DI: **[h, k]= impulse(rd); [k, h]**

FI: **[h, t]= impulse(rc); [t, h]**

utasítással határozhatók meg. Ekkor a figyelembe vett időpontokat a MATLAB választja meg.

Az impulzusválasz *értékeit* a $k = 0,1,2,\ldots,L$ illetve a $t = 0, dt, 2 \cdot dt, \ldots, L \cdot dt$ időpontokban a

DI: **k=0:L; [h, k]= impulse(rd, k); [k, h]**

FI: **t=0:dt:L*dt; [h, t]= impulse(rc, t); [t, h]**

utasításokkal számíthatjuk. A FI impulzusválaszt a MATLAB közelítőleg számítja. A **t** vektor másként is megadható (ML-3.2. pont).

Az impulzusválasz értékét a $k = p$ illetve a $p \cdot dt$ időpontban a **h(p+1)** adja (mivel a **h** vektor első eleme a $k = 0$ illetve a $t = 0$ időpontnak felel meg).

Az impulzusválasz *ábráját* (a DI esetre oszlop-diagram a célszerű) a választott pontokra a **h** meghatározása után a

DI: **bar(k, rd, w)** vagy **bar(k, rd)**

FI: **plot(t, rc)**

utasításra kapjuk. A **w** $(0 < w < 1)$ az oszlopok szélességét adja, a **w** elhagyása $w = 0{,}8$ választást jelent.

Az időpontokat és az impulzusválasz értékeit táblázatos alakban rendszerint csak decimálva állítjuk elő. Legyen **m** (például **m=5**) az a természetes szám amellyel decimálni akarunk, akkor a **h** meghatározása után a $k = k_0, k_0 + m, k_0 + 2m, \cdots, k_0 + L_1 m$ illetve a $t = k_0\,dt, (k_0 + m)dt, (k_0 + 2m)dt, \cdots, (k_0 + L_1\,m)dt$ időpontokra

DI: **km=k0 : m : L1*m; [km', h(km+1)]**

FI: **km=k0 : m : L1*m; [dt*km', h(tm+1)]**

megadja a kijelölt időpontokban az impulzusválaszt. Rendszerint **k0=0.** Természetesen csak a $k_0 + L_1 \le L$ feltételt kielégítő L_1 választható.

A decimálás elvégezhető a **decimate** utasítással is. Ezt nem részletezzük.

ML-2.3. Az ugrásválasz meghatározása

Az **rsz** azonosítójú rendszer ugrásválaszának ábráját és numerikus értékeit ugyanúgy határozhatjuk meg, mint az impulzusválaszét csak az **impulse** utasítás helyére a **step** utasítást kell írni.

A $D \neq 0$ esetben a FI ugrásválasz a $t = +0$ helyen nem nulla, de ez nem igényel külön kezelést, a MATLAB ezt adja meg (nem az ugrásválasz bal és jobb oldali határértékének számtani közepét).

ML-2.4. Rendszer-identifikáció

A DI rendszer impulzusválaszát meg tudjuk határozni a $k = 0,1,\ldots,L$ diszkrét időpontokban, ha ismerjük a gerjesztés és a válasz értékét ugyanezen időpontokban. A DI rendszer lehet egy FI rendszer diszkrét idejű szimulátora.

E rendszer-identifikációs eljárás MATLAB utasításai („deconvolution")

u=[u[0] u[1] ... u[L]]; y=[y[0] y[1] ... y[L] zeros(1, L)];

[h]=deconv(y, u); k=0:L; [k; h]'

Adott k esetén $h[k]$ értékét **h(k+1)** adja.

Az impulzusválasz oszlop-diagramját a **bar(k, h, w)** utasítással kapjuk.

Ha $u[0] = 0$ és ezért $y[0] = 0$, akkor ezeket el kell hagyni az adatokból, vagyis csak L számú hasznos adat-párral rendelkezünk. Ekkor $h[k]$ csak $k = 0,1,\ldots,L-1$ ütemekre állítható elő. Ekkor a fenti utasításokban **zeros(1, L-1)** és **k=0:L-1** írandó.

Ugyanígy járhatunk el ha az impulzusválasz és a válasz ismeretében a gerjesztés értékeit akarjuk meghatározni.

ML-2.P. Példák

1. példa DI rendszer impulzusválasza

Határozzuk meg annak a DI rendszernek az impulzusválaszát, amelynek átviteli függvénye

$$H(z) = \frac{z}{z+0{,}5}.$$

A rendszer azonosítója legyen

rd=tf([1 0], [1 0.5];

Az impulzusválasz értékeit táblázatosan illetve lépcsős ábráját az

[h, k]=impulse(rd); [k, h] illetve **impulse(rd)**

utasítások adják ez esetben $0 \leq k \leq 10$ értékeire.

Előírt maximális k értékre $0 \leq k \leq 15$ esetén a

k=0:15; [h, k]=impulse(rd); [k, h] illetve **k=0:15; bar(k,h, 0.2)**

utasítást alkalmazhatjuk. Ennek egy részletét az

k=2:5; [k'; h(k+1)] illetve **k=2:5; bar(k, h(k+1), 0.2)**

utasítások adják. A $h[2]$ értékét **h(3)** adja meg.

Ha eleve csak egy intervallumot akarunk meghatározni, akkor a

k=2:5; [h, k]=impulse(rd); [k, h] illetve **k=2:5; bar(k, h, 0.2)**

utasítások alkalmazhatók.

2. példa FI rendszer impulzusválasza
Határozzuk meg annak a FI rendszernek az impulzusválaszát, amelynek átviteli függvénye

$$H(s)=\frac{1}{s+0,5}.$$

A rendszer azonosítója legyen

rf=tf(1, [1 0.5]);

Az impulzusválasz értékeit táblázatosan illetve ábráját az

[h, t]=impulse(rf); [t, h] illetve **impulse(rf)**

utasítások adják ez esetben $0 \le t < 12$ értékeire $dt \approx 0{,}11$ lépésközzel.

Előírt maximális t értékre és dt lépésközzel $0 \le t \le 1$, $dt = 0{,}05$ esetén a

t=0:0.05:1; [h, t]=impulse(rf, t); [t, h] illetve **t=0:0.05:1; plot(t,h)**

utasítást alkalmazhatjuk. A $h(0.5)=h(10\cdot 0.05)$ értéket a **h(11) adja.** Az ábra egy részletét az

p=10:14; plot(p*0.05, h(p+1))

utasítás adja a $0{,}5 \le t \le 0{,}7$ intervallumban.

A MATLAB ugyanezt az impulzusválaszt adja a

$$H_1(s)=\frac{4s+3}{s+0,5}=4+\frac{1}{s+0,5} \Rightarrow h_1(t)=4\delta(t)+h(t)$$

átviteli függvényű, tehát például **rf1=tf([4 3], [1 0.5])** azonosítójú rendszerre is, tehát a Dirac-impulzus összetevőt „kézzel" kell számítani és figyelembe venni.

***3. példa** DI rendszer identifikációja
Egy DI rendszer gerjesztése $u[k]=\varepsilon[k]\{1-0.5^k\}$. Az erre adott válasz következő értékeit ismerjük: $y[0]=0$, $y[1]=0{,}5$, $y[2]=1{,}15$, $y[3]=1{,}795$. Noha ez négy adat, csak három $h[k]$ értéket tudunk meghatározni, mivel $u[0]=0$ és ezért $y[0]=0$ nem információ.

u=[0.5 0,75 0.875]; y=[0.5 1.15 1.795];

h=deconv(y, u); k=0:2; [k;h]'

Érdemes összehasonlítani az eredményt, azzal, amit az $y[3]=1{,}8$ kerekített érték felhasználásával kapunk.

Ha $u[0]$ nem lenne nulla, akkor $y[0]=0$ is hasznos információ lenne.

Használható identifikációhoz természetesen sokkal több válsz-érték ismeretére van szükség.

ML-3. A válasz számítása

ML-3.1. Általános elvek

A DI rendszer $u[k]$ gerjesztéshez tartozó $y[k]$ válasza $k = 0,1,2,\dots,L$ értékeire illetve a FI rendszer $u(t)$ gerjesztéshez tartozó $y(t)$ válasza $t = 0, dt, 2\cdot dt,\dots, L\cdot dt$ értékeire számítható ha megadjuk a rendszer **rd** illetve **rc** azonosítóját (vö. ML-1.), az időpontokat tartalmazó **k** illetve **t** vektort és a gerjesztés értékeit tartalmazó **u** vektort. A válasz értékeit az **lsim** („linear simulation") utasítás hatására az **y** vektor adja meg.

A FI választ a MATLAB közelítőleg számítja.

A válasz értékét a $k = p$ illetve a $t = p\cdot dt$ időpontban a **y(p+1)** adja (mivel az **y** vektor első eleme a $k = 0$ illetve a $t = 0$ időpontnak felel meg).

Ha a rendszer kezdeti állapota nem nulla, akkor a rendszer állapotváltozós leírását kell megadni az **ss** utasítással, a kezdeti állapotot egy **x0** oszlopvektorba kell foglalni. A MATLAB a választ a $k = 0$ illetve a $t = 0$ időponttól adja meg a kezdeti állapot figyelembe vételével.

ML-3.2. A gerjesztés megadása

A **k** illetve **t** vektorba foglalt időpontokhoz tartozó **u** gerjesztés-vektor megadható bármely, a MATLAB által megengedett módon, akár pontonként is. Az $u[k]$ illetve az $u(k\cdot dt)$ értékét **u(k+1)** adja meg.

A legegyszerűbb a helyzet, ha a gerjesztés a vizsgált idő-intervallumon belül egyetlen elemi függvénnyel (elemi függvények szorzatával és ilyenek összegével) kifejezhető. Például

DI: **k=0:10; u=2+3*(0.8.^k).*cos(0.5*k);** FI: **t=0:0.01:1; u=1+2*exp(-2*t);**

A FI idővektort nem csak **t=0:dt:T** alakban adhatjuk meg, hanem **t=0:T/L:T** vagy **t=0:dt:L*dt** alakban is, amelyek $T = L\cdot dt$ esetén ekvivalensek. Még célszerűbb egy egész értékű k változót bevezetni:

FI: **dt=0.01; k=0:100:1; t=k*dt; u=1+2*exp(-2*t);**

A gerjesztés ábráját a következő utasítással állíthatjuk elő:

DI: **bar(k, u, 0.2)** FI: **plot(t, u)**

Ha a függvény intervallumonként írható le elemi függvényekkel, akkor az $L = 10$ illetve a $T = 1$ hosszúságú háromszögimpulzus megadásának módja a $0 \le k \le 20$ illetve a $0 \le t \le 2$ intervallumban például

DI: **k=0:10; u(k+1)=10-k; k=11:20; u(k+1)=0;**

FI: **dt=0.05; k=0:20; t=k*dt; u(k+1)=10-10*t; k=11:20; t=k*dt; u(k+1)=0;**

A gerjesztés értékeit vagy ábráját (például ellenőrzés érdekében) a következő utasítással állíthatjuk elő:

DI: **k=0:20; [k; u(k+1)]'** vagy **k=0:20; bar(k, u, 0.2)**

FI: **k=0:40; t=k*dt; [t, u]** vagy **k=0:40; t=k*dt; plot(t, u)**

ML-3.3. A válasz meghatározása

A DI rendszer **rd** azonosítójának, **k** idővektorának, **u** gerjesztés-vektorának illetve a FI rendszer **rc** azonosítójának, **t** idővektorának, **u** gerjesztés-vektorának megadása után meghatározhatjuk az **y** válaszvektor ábráját és értékeit.

Bekapcsolási folyamat (belépő gerjesztés, zérus kezdeti állapot) esetén a válasz *ábráját* a következő utasítással kapjuk:

DI: **lsim(rd, u, k)** FI: **lsim(rc, u, t)**

A válasz *értékeit* a következő utasítás adja:

DI: **y=lsim(rd, u, k)** FI: **y=lsim(rc, u, t)**

A válasz értékét a $k = p$ illetve a $t = p \cdot dt$ időpontban **y(p+1)** adja. Az összes számított értéket táblázatos alakban a **[k', y]** illetve **[t', y]** adja.

A válasz ábráját (DI esetben oszlop-diagram a célszerű)

DI: **bar(k, y, w)** FI: **plot(t, y)**

állítja elő. A FI esetben a gerjesztést és a választ együtt is ábrázolhatjuk a **plot(t, y, t, u)** utasítással. Több változó együttes ábrázolásának további módjait a **help plot** segítségével ismerhetjük meg.

Az időpontokat és a válasz értékeit táblázatos alakban rendszerint csak decimálva (minden m-edik időpontban, például minden tizedik) akarjuk csak elő állítani. Az **y** meghatározása után a $k = k_0, k_0 + m, k_0 + 2m, \cdots, k_0 + L_1 m$ DI illetve a $t = k_0\, dt, (k_0 + m)dt, (k_0 + 2m)dt, \cdots, (k_0 + L_1 m)dt$ FI időpontokra ($k_0 + L_1 \le L$ és rendszerint $k_0 = 0$) a válasz értékeit a kijelölt időpontokban a következő utasítás adja:

DI: **km=k0 : m : L1*m; [km', y(km+1)]**

FI: **km=k0 : m : L1*m; [dt*km', y(tm+1)]**

Tetszőleges *kezdeti állapot* (vagyis nem belépő gerjesztés) esetén a rendszer állapotváltozós alakjával, az **ss** utasítással adandó meg, a kezdeti állapotot egy **x0** oszlopvektorba foglaljuk. A válasz meghatározása ugyanúgy történhet, mint bekapcsolási folyamat esetén csak most az ábra

DI: **lsim(rd, u, k, x0)** FI: **lsim(rc, u, t, x0)**

a válasz értékei pedig a

DI: **[y, k, x]=lsim(rd, u, k, x0)**

FI: **[y, t, x]=lsim(rc, u, t, x0)**

utasításra adódnak. A táblázatos alakba vagy az ábrába belefoglalhatjuk **az x** állapotvektort vagy annak r-edik rendezőjét amelyet **x(:, r)** azonosít. Például

plot(t, y, t, x(:, 1), t, x(:, 3))

megadja $y(t), x_1(t), x_3(t)$ ábráját. Másodrendű rendszerre a **plot(x(:,1), x(:,2))** utasítással a síkbeli trajektória ábrája adódik.

További részletek **help lsim** hatására tudhatók meg.

ML-3.P Példák

1. példa DI rendszer válasza belépő gerjesztésre
Egy DI rendszernek az állapotváltozós leírása és gerjesztése

$$\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} 0 & -0{,}24 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} -0{,}24 \\ 1 \end{bmatrix} u, \quad y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + u;$$

$$u[k] = \{\varepsilon[k] - \varepsilon[k-6]\}\{5 + 0{,}2 \cdot k^2\} + \{\varepsilon[k-6] - \varepsilon[k-10]\} 5 \sin\frac{\pi}{6}(k-6).$$

Meghatározandó az $y[k]$ válasz a $0 \le k \le 15$ intervallumban táblázatosan és oszlop-diagramban, külön táblázatban $y[k]$ a $k = 0, 5, 10, 15$ diszkrét időpontokra.

A rendszer megadása

A=[0 -0.24; 1 1]; B=[-0.24; 1.5]; C=[0 1]; D=1; rd=ss(A, B, C, D, 1);

A gerjesztés megadása a vizsgálandó intervallumra

k=0:5; u(k+1)=5+0.2*k.^2; k=6:9; u(k+1)=5*sin(pi*(k-6)/6); k=10:15; u(k+1)=0;

Ellenőrzésként például **k=0:15; bar(k, u, 0.2).** A válasz értékei illetve ábrája

k=0:15; y=lsim(rd, u, k); [k', y] illetve **bar(k, y, 0.2)**

A válasz rövidített táblázata

k1=0:5:15; [k1', y(k1+1)] vagy **k2=[0 5 10 15]; [k2', y(k2+1)]**

A második alak akkor is alkalmazható, ha az időpontok távolsága nem állandó.

2. példa FI rendszer válasza belépő gerjesztésre
Egy FI rendszernek az állapotváltozós leírása és gerjesztése

$$\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -3 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \quad y = \begin{bmatrix} 1 & 5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix};$$

$$u(t) = \{\varepsilon(t) - \varepsilon(t-2)\} 2\{1 - e^{-3t}\} + 2\varepsilon(t-2).$$

Meghatározandó az $y(t)$ válasz a $0 < t < 5$ intervallumban táblázatosan és grafikusan, külön táblázatban $y(t)$ a $t = 0,\ 0{,}1,\ 0{,}2, \ldots,\ 5$ időpontokban, a válasz maximuma és állandósult értéke.

A rendszer megadása

A=[0 1; -3 -4]; B=[0; 1]; C=[1 5]; D=0; rf=ss(A, B, C, D);

A gerjesztés megadása a vizsgálandó intervallumra $dt = 0{,}01$ lépésközzel

dt=0.01; p=0:200; t=p*dt; u(p+1)=2-2*exp(-3*t); p=201:500; t=p*dt; u(p+1)=2;

Ellenőrzésként például **p=0:500; t=p*dt; plot(t, u).** A válasz értékei illetve ábrája

p=0:500; t=p*dt; y=lsim(rf, u, t); [t', y] illetve **plot(t, y,)**

A válasz rövidített (51 sorból álló) táblázata

p1=0:10:500; t1=p1*dt; [t1', y(p1+1)]

A részletes táblázatból $y_{max} \approx y(1{,}07) \approx 1{,}8511$ adódik. Az utolsó számított válaszérték $y(5) = 0{,}7068$. Látható azonban, hogy a válasz állandósult értéke ennél valamivel kisebb, tehát a számítást hosszabb intervallumra kellene kiterjeszteni. A válasz állandósult állapotban a pontos számítás szerint $\bar{y} = 2/3 \approx 0{,}6667$. Ha a fenti számítás során az **500** értéket **5000** értékkel helyettesítjük, akkor **y(5001)=0.6667** adódik.

ML-4. Az átviteli karakterisztika

ML-4.1. Általános elvek

A DI rendszer $H(e^{j\vartheta})$ illetve a FI rendszer $H(j\omega)$ átviteli karakterisztikáját a MATLAB az átviteli függvényből képzi $z = e^{j\vartheta}$ illetve $s = j\omega$ helyettesítéssel. Az eredmény ezért csak akkor fogadható el, ha előzőleg meggyőződtünk a rendszer stabilitásáról a rendszer rendszermátrixa sajátértékeinek vagy átviteli függvénye pólusainak meghatározásával a **la=eig(A)** vagy a **p=pole(rsz)** utasítással.

Alább a DI és FI rendszer közös tárgyalása során a ϑ (rad mértékegységben) illetve az ω (rad/s mértékegységben) körfrekvenciákat közösen egy **w** vektorba foglaljuk. Ha az FI esetben a frekvenciákat akarjuk egy **f** vektorban megadni, akkor a **w=2*pi*f** értelmezést használhatjuk. (Más eljáráshoz: **help frd** „frequency response data", ahol további információk is találhatók.)

Az alább megadott módokon értelmezett átviteli karakterisztikát nem tekinthetjük egy rendszer azonosítójának, tehát abból más rendszerleírások vagy azok jellemzői nem határozhatók meg egyszerűen. Az impulzusválasz inverz Fourier-transzformációval számítható az itt nem részletezett **ifft** utasítással. (A **help fft** szolgál részletekkel, az eljárás alkalmazása figyelmet igényel.)

ML-4.2. Az átviteli karakterisztika számítása

Az **rsz** azonosítójú DI illetve FI rendszer amplitúdó-karakterisztikáját és fázis-karakterisztikájának a **w** vektorba foglalt körfrekvenciákon (l. alább) a

[K1, fi1] = bode(rsz, w); p=1:length(w); K(p)=K1(p); fi(p)=fi1(p);

utasítás adja. Az amplitúdó-karakterisztikát decibelben a **k(p)=20*log10(K);** bevezetésével kapjuk. Az összes frekvencia-értékre vonatkozó táblázatot a

[w; K; fi]' vagy **[w; k; fi]'**

utasítással kapjuk. A **w** vektor q-adik eleméhez tartozó értékek a **w(q), K(q), fi(q), k(q)** utasítások szolgáltatják. Több frekvencia esetén **q** vektorként is definiálható (például a ML-3.3. pontban megadott módon).

A **w** vektor megadható a számítandó körfrekvenciák egyenkénti megadásával:

w=[wa wb wc wd];

Lineáris skálázás esetén a **w0** (ez gyakran **0**) és a **w1** (ez DI esetben gyakran π azaz **pi)** által határolt körfrekvencia-intervallumban a következő megadás használható:

w=[w0:dw: w1];

A **dw** megadható **(w1-w0)/L** alakban is $L+1$ számú pontban. Ennek egy más alakja

w=linspace(w0, w1, L);

A FI esetben előnyös lehet a logaritmikus skálázás $\omega_0 = 10^a, \omega_1 = 10^b$ körfrekvenciák között L számú pontban a

w=logspace(a, b, L)

utasítással.

A **w** vektor q-adik eleméhez tartozó értékek a **w(q), K(q), fi(q), k(q)** utasítások szolgáltatják. Több frekvencia esetén **q** vektorként is definiálható (például a ML-3.3. pontban megadott módon). Ekkor egy rövidített (például ritkább osztású vagy egy frekvencia-intervallumra vonatkozó) táblázatot állíthatunk elő.

Az átviteli karakterisztika valós és képzetes része a

[re1, im1] = nyquist(rsz, w); p=1:length(w); re(p)=re1(p); im(p)=im1(p);

utasítással állítható elő, majd kezelhető az előbb leírt módon.

ML-4.3. Az átviteli karakterisztika ábrázolása

Az **rsz** azonosítójú DI illetve FI rendszer átviteli karakterisztikájának *Nyquist-diagramja* a

nyquist(rsz) vagy **nyquist(rsz, w)**

utasítással állítható elő. Az első esetben a **w** vektort a MATLAB választja, a második esetben megválaszthatjuk az előző pontban leírt valamelyik módon.

A *Bode diagram* (a logaritmikus amplitúdó- és fázis-diagram) a

bode(rsz) vagy **bode(rsz, w)**

utasítással állítható elő az előzőhöz hasonló értelmezéssel.

Az előző pontban leírt módon előállított **K** és **fi** meghatározása után a

plot(w, K) plot(w, k) plot(w, fi) plot(w, K, w, fi)

subplot(2, 1, 1), plot(w, K); subplot(2, 1, 2), plot(w, fi)

utasítással állíthatók elő az ábrák. A második sor esetén a két görbe az osztott ábra felső és alsó felében jelenik meg.

Az előző pontban leírt módon előállított **re** és **im** meghatározása után ábrázolható az átviteli karakterisztika valós és képzetes része egyetlen ábrán vagy osztott ábrán. Ennek alapján is előállítható az átviteli karakterisztika Nyquist-diagramja.

ML-4.4. Az átviteli karakterisztika pontonkénti megadása

Az DI illetve FI rendszer átviteli karakterisztikája megadható a **w** vektorba foglalt ϑ_p illetve ω_p körfrekvenciákhoz tartozó, a **H** vektorba foglalt $H\left(e^{j\vartheta_p}\right)$ illetve $H\left(j\omega_p\right)$ értékekkel. Ez tipikusan mérési eredmények feldolgozásánál jelentkezik. A **w** utasításként is megadható például **w=[w0:dw: w1]** alakban.

Az átviteli karakterisztika értékei táblázatos alakban az

[w; H].' vagy **[w; abs(H); angle(H)]'**

utasítással állítható algebrai vagy exponenciális alakban..

Egy más lehetőség egy DI vagy FI rendszert értelmezni a

DI: **rf=frd(H, w, 1)** FI: **rf=frd(H, w)**

utasítással. Az **ENTER** hatására kapjuk a táblázatos alakot (**H** algebrai alakban). Ekkor a Nyquist- vagya a Bode-diagram az előző pontban leírt módon határozható meg. Ez az **rf** illetve **rd** azonosító azonban nem alkalmas az ML-1. szakaszban leírt műveletekre (például az állapotváltozós leírás előállítására).

ML-4.P. Példák

1. példa DI rendszer átviteli karakterisztikája

Határozzuk meg annak a DI rendszernek az amplitúdó- és fázis- karakterisztikáját, amelyet az ML-1.P. első három példájában vizsgáltunk. Adjuk meg az amplitúdó-karakterisztika értékét a $\vartheta = 0$ és a $\vartheta = \pi$ helyen, továbbá maximumát és annak helyét.

rd=tf([2 0.8], [1 0.6 0.25], 1); abs(pole(rd))

Mindkét pólus abszolút értéke 0,5, a rendszer stabilis

te=0:0.01*pi:pi; [K1, fi1]=bode(rd, te); p=1:length(te);

K(p)=K1(p);

plot(te, K(p))

Az ábrából látható, hogy a $K(\vartheta)$ függvénynek egyetlen lokális maximuma van $\vartheta = 2{,}1\,\mathrm{rad}$ környékén értéke 2,7 körüli. A $K(0)$, $K(\pi)$ és $K_{\max}$ értékét

K(1), K(101), max(K)

adja. A maximum helye például az (előző sor felhasználásával könnyen előállítható)

te=2.1:0.01:2.2; [K1, fi1]=bode(rd, te); p=1:length(te); K(p)=K1(p);

[te; K(p)]', max(K)

táblázata alapján határozható meg: $K_{\max} = 2{,}6849, \vartheta = 2{,}13\,\mathrm{rad}$.

2. példa FI rendszer átviteli karakterisztikája

Határozzuk meg annak a FI rendszernek az amplitúdó- és fázis- karakterisztikáját, amelyet az ML-1.P. első három példájában vizsgáltunk. Adjuk meg az amplitúdó-karakterisztika maximumát és annak helyét.

rc=tf([2 0.8], [1 0.6 0.25]); real(pole(rc))

Mindkét pólus valós része negatív, a rendszer stabilis

Az átviteli karakterisztika meghatározásához első tájékozódásra a **bode(rc)** utasítás hatására kapjuk hogy esetünkben a lényeges frekvencia-tartomány 0,1 rad/s és 10 rad/s közé esik. Az amplitúdó-karakterisztikának lapos maximuma van 0,5 rad/s környékén, ettől balra és jobbra monoton csökken. (Nagyobb rendszámú rendszerekre a MATLAB szélesebb intervallumot jelöl ki.) Az

om1=0.4:0.01:0.6; [K1, fi1]=bode(rc, om1); p=1:length(om1); K(p)=K1(p);

[om; K(p)]', max(K)

táblázatból illetve utasításból kiderül, hogy a maximum valóban ebben az intervallumban van 0,41 rad/s körfrekvencián és értéke 0,4189. Kisebb frekvencialépéssel ezek helyett 0,415 rad/s és 0,4191 adódik, de ez a pontosítás valószínűleg fölösleges. Az amplitúdó-karakterisztika a maximum környezetében a **plot(om1, K(p))** utasítással ábrázolható. A fázis-karakterisztikáról (fokban) a **fi(p)=180*fi1(p)/pi** bevezetésével kaphatunk egy választott intervallumban áttekinthető ábrát.

IRODALOMJEGYZÉK

A témakör hatalmas irodalmából csak néhány alapvető könyvet adtunk meg. Ezek irodalomjegyzékében számos további mű található

Anderson, B. D. O. - Vongpanitlerd, S.: Network Analysis and Synthesis. Prentice-Hall, 1973

Antoniou, A.: Digital Filters. McGraw-Hill, 1979

Balabanian, N. - Bickart, Th.: Electrical Network Theory. Wiley, 1969

Britte, R. – Leich, H.: Les filtres numeriques. Masson, 1í990

Candy, J. V.: Signal Processing. McGraw-Hill, 1988

Champaney, D. C.: Fourier Transforms and their Physical Applications. Academic Press, 1973

Chua, L. O. : Introduction to Nonlinear Network Theory. Wiley, 1969

Chua, L. O. - Desoer, C. A. - Kuh, E. S. : Linear and Nonlinear Circuits, McGraw-Hill, 1987

Chua, L. O. - Lin, P. M.: Computer-Aided Analysis of Electronic Circuits, Englewood Cliffs, 1975

Coyle, R. G.: Management System Dynamics. Wiley, 1977

Csáki F.: Korszerű szabályozáselmélet. Akadémiai Kiadó, 1988

Csáki F. (szerk.): Lineáris szabályozási rendszerek analízise. Műszaki Könyvkiadó, 1976

***Davies, B.**: Integral Transforms and Their Applications. Springer, 1978*

Director, S. W. - Kuh, E. S.: Introduction to System Theory. McGraw-Hill, 1972

Doetsch, G.: Anleitung zum praktischen Gebrauch der Laplace-Transformation und der Z-Transformation. Oldenbourg, 1967

Doetsch,G.: Handbuch der Laplace-Transformation I.-II.-III. Birkhauser, 1950-1955-1956

Erdélyi, A. : Tables of Integral Transforms I. McGraw Hill, 1954

Fodor Gy.: A Laplace-transzformáció mûszaki alkalmazása. Műszaki Könyvkiadó, 1966

Fodor Gy.: Lineáris rendszerek analízise. Műszaki Könyvkiadó, 1967

Fodor, G.: Signals, Systems and Networks. Akadémiai Kiadó, 2003

Föllinger, O.: Laplace- und Fourier-Transformation. Hüthig, 1990

Géher, K.: Theory of Network Tolerances. Akadémiai Kiadó, 1971

Géher K. - Solymosi J.: Lineáris áramkörök tervezése. Tankönyvkiadó, 1992

Gabel, R. A. - Roberts, R. A.: Signals and Linear Systems. Wiley, 1963

Guillemin, E. A.: Theory of Linear Physical Systems. Wiley, 1963

Hale, J. K.: Nonlinear Oscillations. McGraw-Hill, 1963

Harkevic, A. A.: Spektry i analiz. Izd. fiz.-mat. lit., 1962

Hayashi, C.: Nonlinear Oscillations in Physical Systems. McGraw-Hill, 1964

Hughes, W. L.: Nonlinear Electrical Networks. Ronald Press, 1960

Jury, E. I. : Sampled-data Control Systems. Wiley, 1958

Kailath, T.: Linear Systems. Prentice-Hall, 1980

Küpfmüller, K.: Die Systemtheorie der elektrischen Nachrichtenübertragung. S. Hirzel Verlag, 1968

Lange, F. H.: Signale und Systeme. Vieweg, 1966

Lynn, P. A. - Fuerst, W.: Digital Signal Processing. Wiley, 1989

Oppenheim, A. - Shafer, R. : Digital Signal Processing. Prentice-Hall, 1975

Oppenheim, A. - Willsky, A. S. - Young, I.: Signals and Systems. Prentice-Hall, 1983

Papoulis, A.: The Fourier-Integral and its Applications. McGraw-Hill, 1962

Papoulis, A.: Signal Analysis. McGraw-Hill, 1977

Papoulis, A.: Circuits and Systems. Holt, 1980

Phillipow, E.: Nichtlineare Elektrotechnik. Akad. Verlagsges. 1963

Rohrer, S. A.: Circuit Theory: An introduction to the State Variable Approach. McGraw-Hill,1970

Rózsa P.: Lineáris algebra és alkalmazásai. Műszaki Könyvkiadó, 1974

Schüssler, H. W.: Netzwerke, Signale und Systeme 1.-2. Springer, 1991

Silijan, D.: Nonlinear Systems. Wiley, 1969

Simonyi E.: Digitális szűrők. Műszaki Könyvkiadó, 1984

Unbehauen, R.: Systemtheorie. Oldenbourg, 1993

Vlach, J. - Singhal, K.: Computer Methods for Circuit Analysis and Design. Van Nostrand, 1983

Zadeh, L. A. - Desoer, C. A.: Linear System Theory - The State Space Approach. McGraw-Hill, 1963

Zemanian, A. H.: Distribution Theory and Transform Analysis. McGraw-Hill, 1965

Zinovev, A. L. - Filippov, L. I.: Vvedenie v teoriju signalov i cepej. Vysshaja Shkola, 1975

Zubov, V. I.: Kolebanija v nelinejnih i upravljemuh sistemah. Szudpromgiz, 1962