3. Hálózatok

3.1 Hálózatok definíciója

A **hálózat** fogalmát a legtömörebben az alábbiak szerint alapozhatjuk meg:

Hálózat = elemkészlet + összekötési szabály

azaz, egy hálózat-típus adott, ha adott egy elemhalmaz és az ezeken értelmezett összekötési szabályok rendszere. A hálózat-osztályon az elemhalmaz elemeiből az összekapcsolási szabályok szerint létrejöhető struktúrák halmazát értjük. Korábbi tanulmányokból ismertként hivatkozhatunk a Kirchoff-féle hálózatokra vagy például a jelfolyam hálózatokra.

A továbbiakban csak jelfolyam hálózatokról lesz szó.

Jelfolyam hálózat definíciója:

Elemkészlet: input-output dobozok (jelfeldolgozó rendszerek, ahogy az előző fejezetben erről szó volt)

Összekötési szabály: kimenetet kimenettel nem szabad összekötni.

A jelfolyam hálózatok halmazán belül definiáljuk a diszkrétidejű lineáris hálózatok osztályát:

Diszkrétidejű lineáris hálózat definíciója:

Elemkészlet: elemi alkatrészek
és minden olyan doboz, ami diszkrétidejű hálózattal létrehozható (rekurzív szabály)

Elemi alkatrészek: összeadó,
szorzó (konstanssal),
késleltető (órajelnyi)

Összekötési szabály: kimenetet kimenettel nem szabad összekötni.

Ebben a kurzusban hálózatok alatt (ha adott esetben másként nem hangsúlyozzuk) diszkrétidejű, lineáris, jelfolyam hálózatokat értünk.

Az előző fejezetben tárgyalt rendszer-fogalom és ezen fejezet hálózat-fogalma közötti kapcsolatot illetően az alábbi két megjegyzést tehetjük:

- egyrészt a hálózatok input-output rendszerek összekapcsolásaival jönnek létre,
- másrészt a hálózatok input- és az output-pontjaik közt input-output rendszert hoznak létre, rendszereket hálózatokkal implementálunk, valósítunk meg (általában elemibb rendszerek felhasználásával),

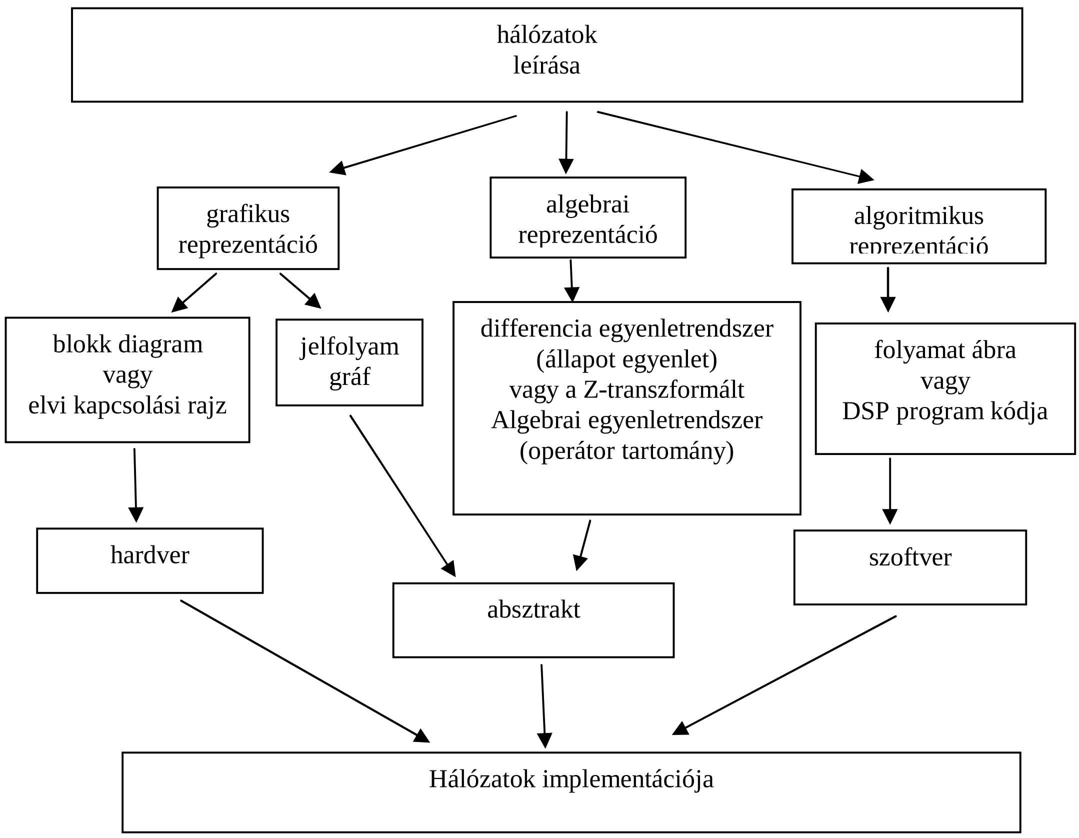
tehát a rendszer és hálózat fogalmunk egymást rekurzívan feltételező, egymást kölcsönösen magában foglaló fogalom.

3.2 Hálózatok reprezentációi, implementációi

A hálózatokat különböző módon írhatjuk le. Néhány fontosabb “műfaj”:

- blokkdiagram, áramköri rajz
- jelfolyam gráf
- egyenletrendszer
- algoritmus, utasítássorozat, program kód

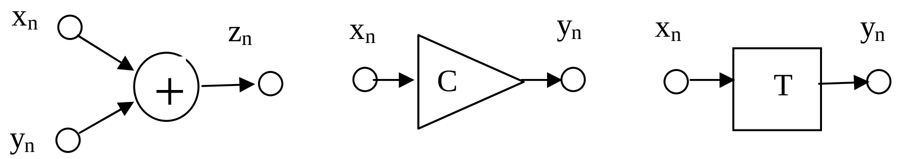
A továbbiakban a hálózatok reprezentációjának, leírásának (formális definíciójának, elvi implementációjának) az alábbi ábrán összefoglalt különböző lehetőségeit, módszereit tekintjük át.



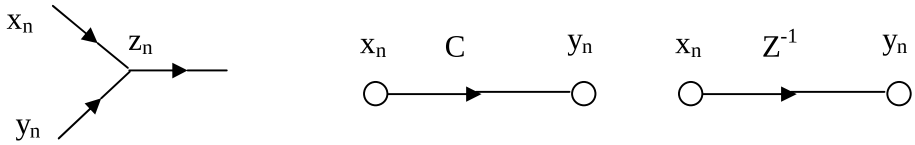
A továbbiakban ezen különböző hálózatleírásokra és ekvivalens átszámításaira nézünk példákat.

A diszkrét idejű lineáris hálózatokat felépítő elemi “alkatrészek” ábráit, gráfjait és egyenleteit az alábbi táblázatban foglaltuk össze:

Elem	Összeadó	szorzó (erősítő)	késleltető



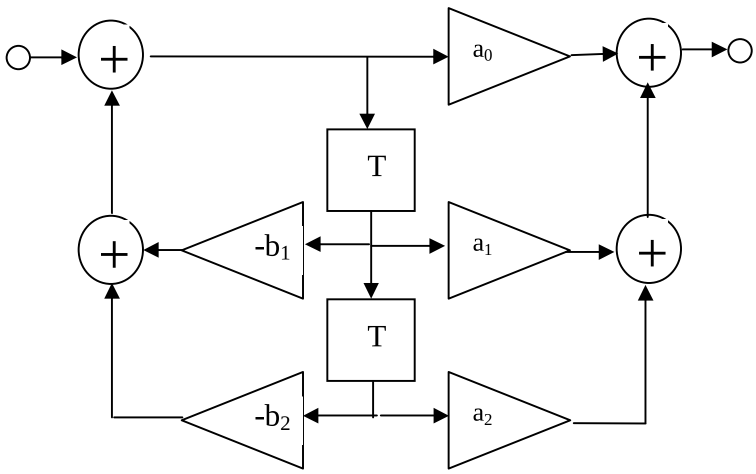
Gráf



Egyenlet	$z_n = x_n + y_n$	$y_n = c \cdot x_n$	$y_n = x_{n-1}$

Egy hálózat elemeinek dobozai és az összeköttetései lerajzolhatóak, ekkor kapjuk a hálózat **blokkdiagramját**.

Tekintsük példaként az alábbi, (két késleltetőt, öt szorzót és négy összeadót tartalmazó) hálózatot:

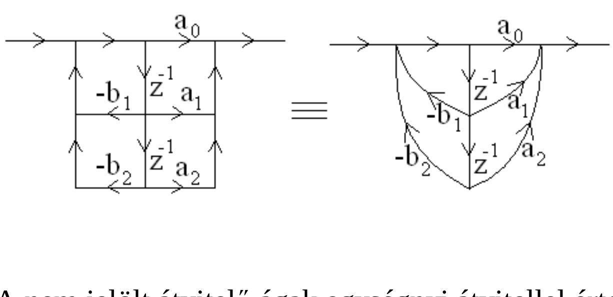


Az adott elemkészletből felépíthető blokkdiagramok triviális módon átírhatóak jelfolyam gráffá.

Jelfolyam gráf, melyre a következő állítások igazak**:**

- Irányított gráf: csomópontok és irányított élek (ágak) halmaza
- A gráf paraméterei az irányított ágak ágátvitelei (jelöletlen ágátvitel egységnyi)
- A gráf csomópontjaihoz tartozó értékek a gráf változói
- A gráf csomópontjai összegezik a befutó ágak kimenő értékeit
- Az él bemenő értéke az él kiinduló csomópontjához tartozó változó értéke
- Az él kimenő értéke a bemenő él ágátvitelszerese

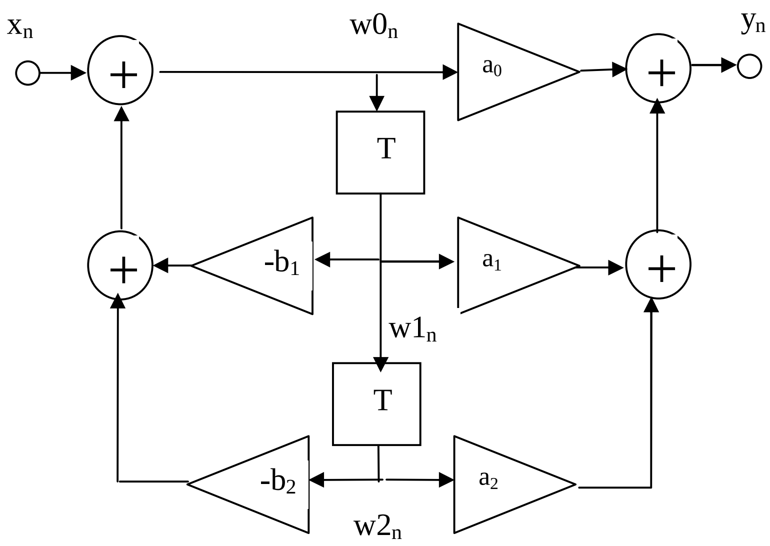
A példaként lerajzoltuk a fenti, blokk diagrammal adott hálózat gráfját:



A nem jelölt átvitelű ágak egységnyi átvitellel értendőek.

A fenti példa alapján nyilvánvaló, hogy minden blokkdiagrammal adott jelfolyam hálózat (melyeknek input-output dobozai transzferfüggvénnyel jellemezhetőek, azaz lineáris és invariáns rendszerek) leírható jelfolyam gráffal is, illetve minden jelfolyam gráf átrajzolható blokk diagrammá.

A blokk diagramban szereplő dobozok (jelátvívő) tulajdonságai és az összeköttetésekkel létrejövő kényszerek **egyenletekkel** is kifejezhetőek. Példaként vezessük be azt az adhoc egyenletrendszert, melynek változói az x_n bemeneti-, az y_n kimeneti- és a késleltetők bemenetein és kimenetein fellépő $w0_n$, $w1_n$, $w2_n$ jelek:



A blokkdiagram alapján felírható egyenletrendszer:

$w0_n = x_n - b_1\, w1_n - b_2\, w2_n$

$y_n = a_0\, w0_n + a_1\, w1_n + a_2\, w2_n$

$w2_n = w1_{n-1}$

$w1_n = w0_{n-1}$

A fenti időtartományi egyenlet egy – az x_n , $w1_n$, $w2_n$, y_n számsorozatokra vonatkozó - differencia egyenlet rendszer. Az egyenletek Z-transzformálásával kapjuk az operátortartománybeli algebrai egyenleteket:

W0(z) = X(z) – b_1 W1(z) – b_2 W2(z)

Y(z) = a_0 W0(z) + a_1 W1(z) + a_2 W2(z)

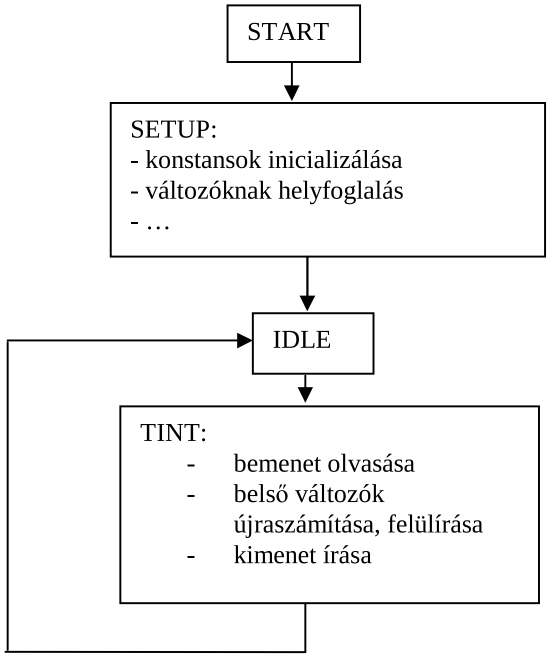
W2(z) = z^{-1} W1(z)

W1(z) = z^{-1} W0(z)

Gyakran valamilyen jelfeldolgozó processzoron implementáljuk a hálózatot.

Egy – a fenti típusú differencia egyenlet szerinti működést megvalósító – általános célú **digitális jelfeldolgozó processzor valós idejű** jelfeldolgozást megvalósító (azt implementáló) **programja** tipikusan két részből áll:

- A start után egyszer végrehajtandó SETUP program részből
- A processzálási ciklus órajelének megfelelő időzítés szerinti végtelen ciklusban vérehajtott TINT (timer interrup routin) program részből áll.



A példánkban fentebb blokkdiagrammal, jelfolyam gráffal és egyenletrendszerrel is megadott hálózat DSP kódja egy képzeletbeli pszeudo nyelven a következő lehet:

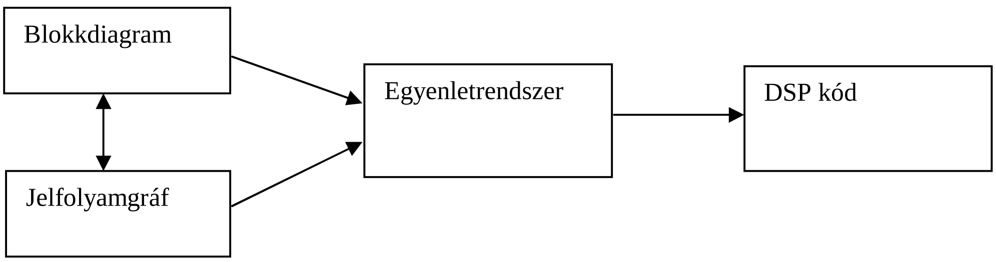
SETUP:

– processzor és hardver platform függő kezdeti beállítások
- változók számára memória helyfoglalások: XN, YN, W0N, W1N, W2N
- együtthatók számára helyfoglalás és értékadás: MB1(-b_1), MB2(-b_2), A0, A1, A2.

```
LOOP:  IDLE                 ; várakozás interruptra
    B LOOP

TINT:                       ; timer interrupt kezelő eljárás
    IN     PORTAD, XN       ; ADC -> x(n)
    LD     XN, A            ; load accumulator x(n) -> ACC
    MAC    MB1,W1N,A        ; multiplay&accumulate  ACC-b1*w1(n) -> ACC
    MAC    MB2,W2N,A        ; multiplay&accumulate  ACC-b2*w2(n) -> ACC
    ST     A, W0N           ; store accumulator  ACC -> w0(n)
    LD     #0, A            ; zero accumulator
    MAC    A0, W0N, A       ; multiplay&accumulate  ACC+a0*w0(n) -> ACC
    MAC    A1, W1N, A       ; multiplay&accumulate  ACC+a1*w1(n) -> ACC
    MAC    A2, W2N, A       ; multiplay&accumulate  ACC+a2*w2(n) -> ACC
    ST     A, YN            ; store accumulator  ACC -> y(n)
    OUT    YN, PORTDA       ; y(n) -> DAC
    DELAY  W1N              ; minta öregítés következő ciklushoz   W1N -> W2N
    DELAY  W0N              ; minta öregítés következő ciklushoz   W0N -> W1N
    RET                     ; visszatérés a LOOP ciklusba
```

Az eddigi példában a hálózat különböző leírásait, lehetséges implementációit az alábbi sorrendben néztük végig:



Egy másik példaként induljunk ki egy másik hálózatnak DSP kóddal megadott implementációjából.

SETUP:

– processzor es hardver platform függő kezdeti beállítások
- változók számára memória helyfoglalások: XN, YN, W1N, W2N
- együtthatók számára helyfoglalás és értékadás: MB1(-b_1), MB2(-b_2), A0, A1, A2.

```
LOOP:  IDLE                     ; várakozás interruptra
       B LOOP

TINT:                           ; timer interrupt kezelő eljárás
       IN     PORTAD, XN        ; ADC -> x(n)
       LD     W1N, A            ; load accumulator w1(n) -> ACC
       MAC    A0N, XN, A        ; multiplay&accumulate  ACC+a0*x(n) -> ACC
       ST     A, YN             ; store accumulator  ACC -> y(n)
       OUT    YN, PORTDA        ; y(n) -> DAC
       LD     W2N, A            ; load accumulator w2(n) -> ACC
       MAC    A1N, XN, A        ; multiplay&accumulate  ACC+a1*x(n) -> ACC
       MAC    MB1,YN,A          ; multiplay&accumulate  ACC-b1*y(n) -> ACC
       ST     A, W1N            ; store accumulator  ACC -> w1(n+1)
       LD     #0, A             ; zero accumulator
       MAC    A2, XN, A         ; multiplay&accumulate  ACC+a2*x(n) -> ACC
       MAC    MB2, YN, A        ; multiplay&accumulate  ACC-b2*y(n) -> ACC
       ST     A, W2N            ; store accumulator  ACC -> w2(n+1)
       RET                      ; visszatérés a LOOP ciklusba
```

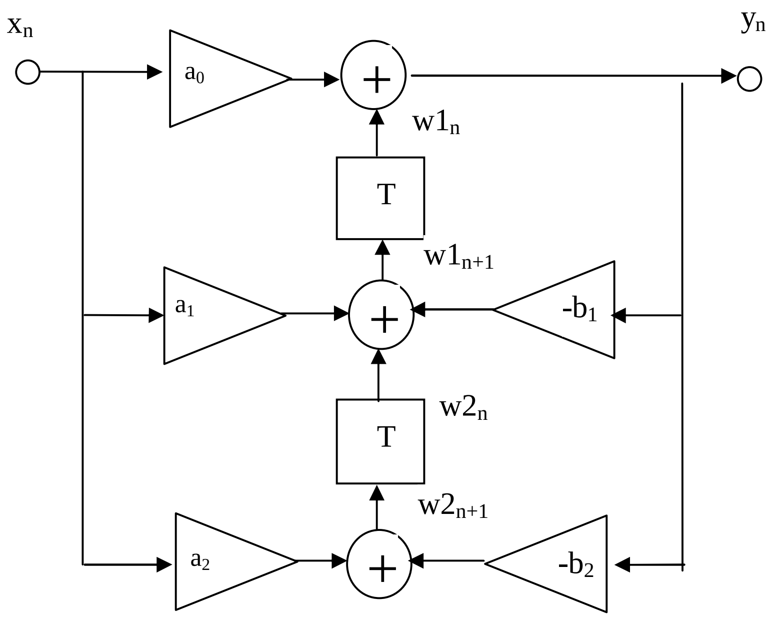
A DSP kód alapjám felírhatjuk a hálózat egyenletrendszerét:

$y_n = w1_n + a_0 x_n$

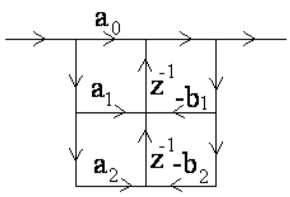
$w1_{n+1} = w2_n + a_1 x_n - b_1 y_n$

$w2_{n+1} = a_2 x_n - b_2 y_n$

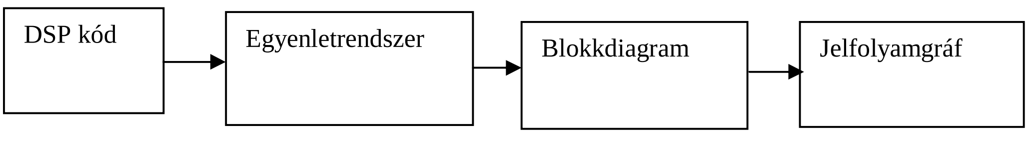
Az egyenletrendszer alapján felrajzolható a hálózat blokkdiagramja:



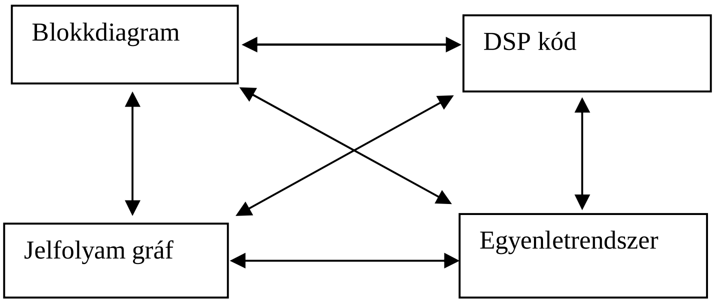
Az egyenletrendszer és/vagy a blokkdiagram alapján felrajzolható a hálózat jelfolyam gráfja:



A második hálózat példája kapcsán a hálózat különböző leírási módjait alábbi út mentén jártuk be:



A két példa tanulságainak általánosításaként megállapíthatjuk, hogy a hálózatokat grafikus, algebrai és algoritmikus reprezentációikkal egyaránt leírhatjuk a különböző leírásokat egymásba átalakíthatjuk. Egy hálózat különböző reprezentációi egymásba kölcsönösen átszámíthatóak.



Valamely reprezentációval adott hálózatnak bármely más reprezentációval való leírása, implementációja előállítható. A hálózatnak mindig egy célszerű reprezentációjával dolgozzunk.

A továbbiakban hálózat alatt azt a struktúrát értjük, mely akár blokkdiagrammal, akár jelfolyam gráffal, akár egyenletrendszerrel, akár folyamatábrával, akár DSP kóddal megadható, leírható, implementálható.

A hálózatok kapcsán alapvetően három feladatot fogalmazhatunk meg:

- **Modell (implementáció) váltás**: a hálózat egyik elvi leírási lehetőségéről áttérünk ugyan annak a hálózatnak egy más leírására, más modelljére, más implementációjára.
- **Hálózat analízis**: adott a hálózat valamely teljes leírása. Meghatározandó, hogy a hálózat adott be- és kimenet pontjai között a hálózat milyen átviteli tulajdonságú (milyen transzfer függvényű) rendszert hoz létre. Adott a struktúra, kérdezzük a funkciót.
- **Hálózat szintézis**: Adott egy megvalósítandó transzferfüggvény, meghatározandó az adott elemkészlet és összekapcsolási szabály felhasználásával az a hálózat, mely az adott transzferfüggvényű rendszert létrehozza, implementálja. Adott funkcióhoz keresünk az azt megvalósító hálózatot.

Az eddigiekben az első feladatra néztünk több példát. A továbbiakban az analízis és szintézis feladat megoldási módszereinek elvi alapjait tekintjük át.

3.3 Hálózat analízis

A hálózat analízisnek sok lehetséges koncepciójú módszerei közül az alábbi két alapvetö és egyben nagyon különböző megközelítését emeljük ki:

- egyenletrendszer felírása és megoldása
- jelfolyamgráf felírása és megoldása

3.3.1 Egyenletrendszer és megoldása

Az első helyen említett módszer közismert és alapvető, itt csak egy adhoc példaval illusztráljuk.

A fentebbi példában adott hálózat által létrehozott rendszer H(z) transzferfüggvényét az alábbiak szerint határozhatjuk meg: Felírjuk a hálózat egyenletrendszerét az operátortartományban, majd elimináljuk az állapotváltozókat, és kifejezzük a H(z) = Y(z) / X(z) hányadost.

Az egyenletrendszer:

(1) $\mathrm{W}_0(z) = \mathrm{X(z)} - b_1\mathrm{W}_1(z) - b_2\mathrm{W}_2(z)$,

(2) $\mathrm{W}_1(z) = z^{-1} \cdot \mathrm{W}_0(z)$,

(3) $\mathrm{W}_2(z) = z^{-1} \cdot \mathrm{W}_1(z)$,

(4) $\mathrm{Y}(z) = a_0\mathrm{W}_0(z) + a_1\mathrm{W}_1(z) + a_2\mathrm{W}_2(z)$

Egy út a megoldáshoz:

(2,4)$\rightarrow$ $\mathrm{W}_2(z) = z^{-2} \cdot \mathrm{W}_0(z)$. Ezt és (2)-t behelyettesítve (1)-be és (4)-be,

$$\mathrm{W}_0(z) = \mathrm{X(z)} - b_1\mathrm{z}^{-1}\mathrm{W}_0(z) - b_2\mathrm{z}^{-2}\mathrm{W}_0(z) \rightarrow \mathrm{W}_0(z) = \frac{\mathrm{X(z)}}{\left(1 + b_1\mathrm{z}^{-1} + b_2\mathrm{z}^{-2}\right)} \text{ és}$$

$$\mathrm{Y}(z) = a_0\mathrm{W}_0(z) + a_1\mathrm{z}^{-1}\mathrm{W}_0(z) + a_2\mathrm{z}^{-2}\mathrm{W}_0(z) \rightarrow \frac{\mathrm{Y}(z)}{\left(a_0 + a_1\mathrm{z}^{-1} + a_2\mathrm{z}^{-2}\right)} = \mathrm{W}_0(z).$$

A két egyenletet összevetve:

$$\mathrm{H}(z) = \frac{\mathrm{Y(z)}}{\mathrm{X(z)}} = \frac{a_0 + a_1\mathrm{z}^{-1} + a_2\mathrm{z}^{-2}}{1 + b_1\mathrm{z}^{-1} + b_2\mathrm{z}^{-2}}.$$

Adott hálózatra felírható egyenletek adhoc módszere mellett vannak szisztematikus módszerek is. Itt most csak megemlítjük az egyébként közismert állapotváltozós egyenletek módszerét. Az állapotvátozok a hálózat tárolóinak kimeneti értékei, melyeket az **w**(n) vektorban foglalhatunk össze. Ekkor a tárolók bemenetén a **w**(n+1) értékek lesznek, azaz a következő órajelre beíródó állapot változó értékek. A bemeneti és a kimeneti jeleket az **x**(n) és **y**(n) vektorokban összefogva mindig felírhatjuk az alábbi alakú állapotváltozós egyenletrendszert:

w(n+1) = **A w**(n) + **B x**(n)

y(n) = **C w**(n) + **D x**(n)

ahol **A, B, C, D** a megfelelő együttható mátrixok.

A továbbiakban részletesebben ismertetjük a hálózat analízis feladatának, a gráfelméleti meggondolásokon alapuló, és a Mason-formula néven ismeretes, különösen hatékony módszerét.

3.3.2 Jelfolyam gráf ekvivalens átalakításai, fokozatos egyszerűsítése

Adott forrás- és nyelő-csomópont közötti átvitel szempontjából két gráf ekvivalens, ha ugyanazt a transzferfüggvényt valósítják meg, azaz kívülről nem megkülönböztethetőek.

Két csomópont közötti átvitel szempontjából a gráf egyszerűsítése a belső csomópontokat és éleket már nem tartalmazó, ekvivalens gráf meghatározását jelenti. Ekvivalens gráfokat kapunk, ha részgráfokat ekvivalens részgráfokkal helyettesítünk.

Részgráfok nevezetes ekvivalenciái:
- lánc-csomópont elliminálása:
- párhuzamos ágak összevonása
- Y-V átalakítások:
- önhurok megszüntetése:

Nevezetes hibák, félreértések:

3.3.3 A Mason formula

A hálózat analízisnek egy másik hatékony módszere az, amikor a hálózat jelfolyam gráfjának a grafikus analízise (hurkok, direkt utak) alapján írjuk fel a létrejövő H(z) transzferfüggvényt.

Ehhez definiáljunk néhány egyszerű fogalmat!

Forrás csomópont:
: A jelfolyam gráf azon csomópontja, amelybe nem fut él.

Nyelő csomópont:
: A jelfolyam gráf azon csomópontja, amelyikből nem indul él.

Lánc csomópont:
: A jelfolyam gráf azon csomópontja, amelyből csak egy ág megy ki és csak egy megy be.

Hurok:
: A jelfolyam gráfnak az az összefüggő részgráfja, amely csak lánccsomópontokból áll.

Hurokátvitel:
: A hurokban lévő ágak átviteleinek szorzata.

Különböző hurkok:
: Két hurok különböző, ha legalább egy ágban különböznek.

Független hurok pár:

Két hurok független hurok pár, ha közös csomópontjuk sincs.

Független hurok N-es:

N darab különböző hurok független N-est alkot, ha közülük bármely kettőt kiválasztva független hurok párt kapunk.

Független hurok N-es átvitele:

A független hurok N-esben levő hurokátvitelek szorzata.

Gráf determinánsa:

$$\Delta = 1 - \sum_{i=1}^{N_1} L_{i_1}^{(1)} + \sum_{i=1}^{N_2} L_{i_1}^{(2)} \ldots - \ldots + \ldots \sum_{i=1}^{N_N} L_{i_1}^{(N)} ,$$

Ahol N_1 a gráfban lévő különböző hurkok száma,

ha $N_1 > 0$, akkor hol $L_{i_1}^{(1)}$ az i-edik hurokátvitelek, i= 1 … N_1

N_k a gráfban lévő független hurok k-asok száma, k= 2… N_1

$L_{i_1}^{(k)}$ az i-edik f-ggetlen hurok k-as átvitele.

Direkt út:

Az a részgráf, mely egy forrás és egy nyelő csomóponton kívül csak lánc csomópontokat tartalmaz.

Direktátvitel:

A direkt útban lévő ágak átviteleinek szorzata.

Különböző direkt utak:

Két direkt út különböző, ha legalább egy ágban különböznek.

Direkt út maradék gráfja:

A direkt úthoz tartozó csomópontok és az ezekbe és –ből futó élek törlésével kapott maradék gráf.

Direkt út aldeterminánsa:

A direkt úthoz tartozó maradék gráf determinánsa.

Ezen előkészítés után már megfogalmazhatjuk a Mason formula állítását:

Mason formula:

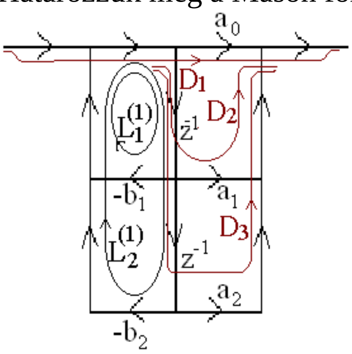
A jelfolyam gráffal megadott hálózat adott forrás és adott nyelő pontja által megvalósított rendszer transzferfüggvénye:

$$H(z) = \frac{\sum_i D_i(z)\cdot\Delta_i(z)}{\Delta(z)}, \text{ ahol}$$

- Δ a teljes jelfolyam gráf determinánsa,
- D_i-k a hálózatban lévő különböző direkt utak átvitelei.
- Δ_i : az i-edik direkt út aldeterminánsa

Példa:

Határozzuk meg a Mason formulával az alábbi hálózat átviteli függvényét!



$L_1^{(1)} = -b_1 z^{-1}$, $L_2^{(1)} = -b_2 z^{-2}$. Mivel független hurkok nincsenek a gráfban:
$\Delta = 1 - \left(L_1^{(1)} + L_2^{(1)}\right) = 1 + b_1 z^{-1} + b_2 z^{-2}$.

$D_1 = a_0$, $D_2 = a_1 z^{-1}$, $D_3 = a_2 z^{-2}$. Bármely direkt utat levonva a gráfból, hurokmentes részgráfot kapunk: $\Delta_i = 1$, i = 1,2,3 (mivel hurokmentes gráfoknak egy az aldeterminánsa).
Ebből

$$\mathrm{H}(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 + b_1 z^{-1} + b_2 z^{-2}},$$ ami pontosan megegyezik az első pontban kapott eredménnyel.

A jelfolyam gráfokkal –és így hálózatokkal is – kapcsolatban vezessük be az alábbi definíciókat:

Kiszámíthatóság:

Egy hálózat **kiszámítható** ha nem tartalmaz késleltetés mentes hurkot.

Rekurzív hálózató:

Egy hálózatot **rekurzívnak** nevezzük, ha gráfja tartalmaz hurkot. Ha nem tartalmaz hurkot, akkor **non-rekurzívnak** nevezzük.

Transzponált gráf:

Egy gráf **transzponált** gráfját a gráf minden éle irányának megfordításával kapjuk.

Transzponált hálózat:

A transzponált gráfhoz tartozó hálózat.

Az előző pont példáiban bemutatott hálózatok egymásnak transzponáltjai.

Ekvivalens hálózat:
Azonos transzfer függvényeket megvalósító hálózatok.

A Mason formula fontos következményeiként az alábbi alapvető megállapításokat tehetjük:

Egy hálózat gráfjában a hurkokat és a direkt utakat felismerve közvetlenül felírhatóak a hálózat által megvalósított transzferfüggvények.

Hurokmentes hálózat determinánsa egységnyi, $\Delta = 1$.

Diszkrét idejű, lineáris hálózat (összeadóból, szorzóból és késleltetőből felépíthető) determinánsa, direkt átvitele (és természetesen aldeterminánsa is) z^{-1} –nek (legfeljebb) polinomja lehet.

Diszkrét idejű, lineáris, kiszámolható hálózattal megvalósított rendszer H(z) transzferfüggvénye legfeljebb racionális tört lehet, azaz összeadóból, szorzóból és késleltetőből felépíthető hálózattal legfeljebb ARMA rendszer valósítható meg.

Nonrekurzív hálózat mindig FIR (MA) rendszert valósít meg.

ARMA rendszer pólusai a hálózat determinánsának gyökei.

Pólust létrehozni csak rekurzív hálózattal lehet.

ARMA rendszer stabilitását a hálózat determinánsa határozza meg (stabil a hálózat, ha determinánsának gyökei az egységkörön belül vannak).

Egy hálózat és transzponáltja ekvivalensek, azonos transzfer függvény valósítanak meg.

3.4 Hálózat szintézis, nevezetes struktúrák

A hálózatszintézis feladatát úgy fogjuk fel, hogy egy adott H(z) transzferfüggvényt megvalósító hálózatot kell meghatározni. Mint a szintézis feladatok általában, úgy ez a feladat is többféle módon oldható meg. Különböző struktúrájú hálózatokkal lehet ugyan azt az input-output rendszert megvalósítani.

A szintézis feladata megoldásának tehát két lépése van:

- struktúrát választunk,
- a kiválasztott struktúrát méretezzük, azaz a paramétereit (a benne lévő szorzók együtthatóit) meghatározzuk

Az alábbiakban a nevezetes struktúrákat mutatjuk be.

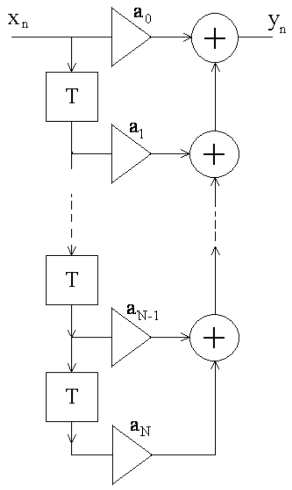
Egy input-output transzfer függvényt megvalósító hálózati struktúra akkor kanonikus, ha minimális számú (N) késleltetőt tartalmaz (ami egyenlő a transzfer függvény fokszámával).

Direkt struktúráknak azokat nevezzük, melyek szorzói azonosak a rendszer polinom/polinom alakú transzferfüggvényének együtthatóival.

3.4.1 A transzverzális struktúra, all-zero modell

Az N-ed fokú transzverzális struktúrának azt a hálózatot nevezzük, melyben a bemeneti minták egy N-tagú késleltető láncba lépnek be, a mindenkori kimenet pedig a késleltetett minták adott együtthatókkal súlyozott összege:

$$y_n = a_0 x_n + a_1 x_{n-1} + ... + a_N x_{n-N}$$



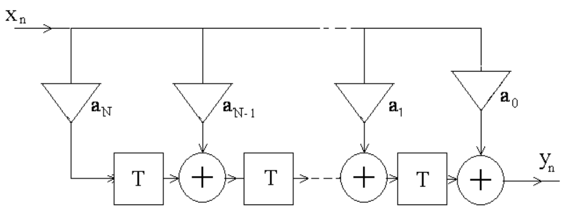
Ezen hálózat tulajdonságai:

- nonrekurzív hálózat, struktúrálisan stabil, FIR rendszert valósít meg
- impulzus válasz sorozata az együtthatók a_0, a_1, ... a_N sorozatának véges tartójú sorozata
- direkt konvolúciós struktúrának is nevezik, a bemeneti x_n, sorozatot a „tárolt" együtthatók a_n, sorozatával konvolválva adja a kimeneti y_n, sorozatot:

$$y_n = x_n * h_n, \text{ ahol } h_n = \begin{cases} 0, & \text{ha } n < 0 \wedge n > N \\ a_n, & \text{ha } 0 \le n \le N \end{cases}$$

- a transzfer függvény: $H(z) = A(z^{-1}) = a_0 + a_1 z^{-1} + ... + a_N z^{-N}$, mely MA rendszert valósít meg, csak nem-triviális zérusai vannak, all-zero modell.
- minden FIR (MA) szűrő megvalósítható transzvezális struktúrával
- minden non-rekurzív hálózathoz létezik ekvivalens transzverzális struktúra
- direkt, kanonikus struktúra

A transzponált transzverzális struktúra:



Ez egy másik, lehetséges „szabályos” struktúra FIR szűrők implementálására.

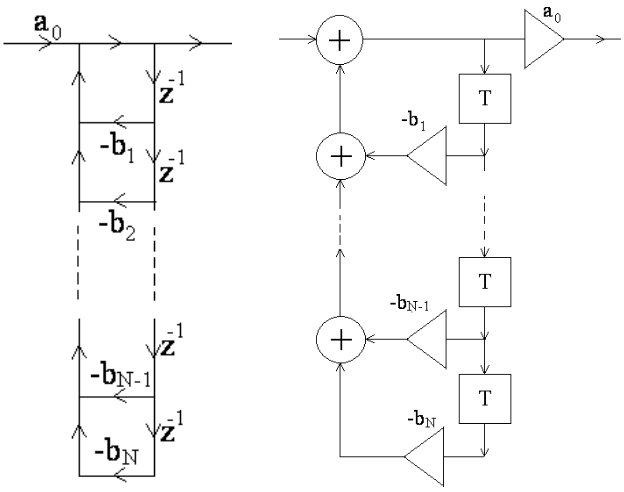
3.4.2 Az AR struktúra, all-pole modell

Ezen struktúránál a kimeneti minták íródnak egy N-elemű késleltető láncba, és a mindenkori kimenet az aktuális bemenet és a korábbi kimenetek b_i $i = 1, 2, \ldots N$ súlyokkal vett összege különbségeként adódik:

$$y_n = a_0 x_n - b_1 y_{n-1} - b_2 y_{n-2} - \ldots - b_N y_{n-N}$$

A hálózat által megvalósított transzfer függvény:

$$\mathrm{H}(z) = \frac{a_0}{B(z^{-1})} = \frac{a_0}{1 + b_1 z^{-1} + \ldots + b_N z^{-N}}$$



Tulajdonságai:

- rekurzív hálózat, IIR rendszert valósít meg
- stabilitása a b_1, ... b_N együtthatóktól függ
- csak nem-triviális pólusai vannak, all-pole modell.
- minden AR szűrő, azaz minden nevező polinom megvalósítható ezzel a struktúrával

- direkt, kanonikus struktúra
- ezen struktúrából is származtatható ekvivalens transzponált struktúra

3.4.3 ARMA rendszerek direkt struktúrái (D0, D1, D2)

Legyen H(z) polinom/polinom alakban megadva (és az általánosság megszorítása nélkül tegyük fel, hogy a számláló és a nevező polinom azonos fokszámú):

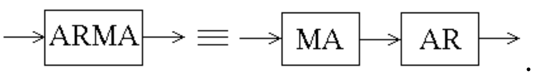
$$\mathrm{H}(z) = \frac{a_0 + a_1 \mathrm{z}^{-1} + \ldots + a_N \mathrm{z}^{-N}}{1 + b_1 \mathrm{z}^{-1} + \ldots + b_N \mathrm{z}^{-N}} \, .$$

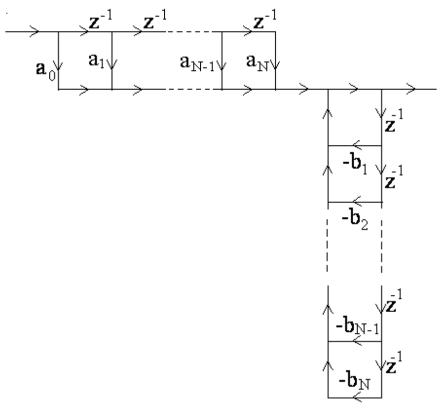
A direkt struktúrákban a hálózatban lévő szorzó együtthatók – mint látni fogjuk – megegyeznek a polinom együtthatókkal.

D0 struktúra:

A racionális H(z) törtfüggvény mindig felírható egy MA és egy AR rendszer szorzataként, azaz egy, a számlálót megvalósító transzferzális és egy, a nevezőt megvalósító AR hálózat láncba kapcsolásával:

$$\mathrm{H(z)} = \mathrm{A}(\mathrm{z}^{-1}) \cdot \frac{1}{\mathrm{B}(\mathrm{z}^{-1})} \, ,$$



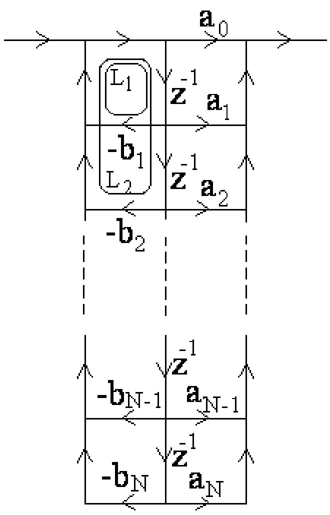


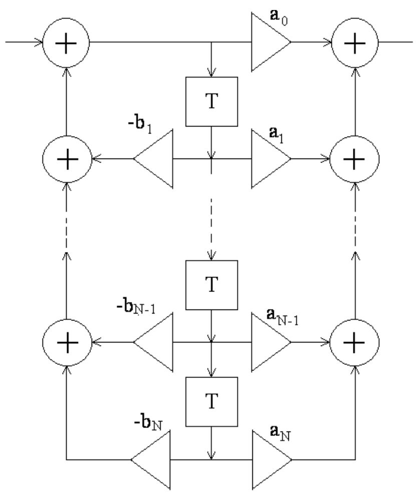
A D0 struktúra nem kanonikus, direkt struktúra.

Minden ARMA rendszerhez egyértelműen létezik D0 struktúrájú realizációs lehetőség.

D1 struktúra

Az számlálót és a nevezőt megvalósító részhálózatok sorrendjét felcserelve, majd az azonos tartalmú, párhuzamosan létrejövő késleltető láncokat egy lánccá összevonva kapjuk az alábbi jelfolyam gráffal ill. blokkdiagrammal megadott hálózatot, melyet D1 struktúrának nevezünk:





A hálózat átviteli függvényét, gyakorlásképpen határozzuk meg a Mason formulával:

- N számú hurkot tartalmaz a gráf:
  $L_1 = -b_1 \cdot z^{-1}$
  $L_2 = -b_2 \cdot z^{-2}$
  $\vdots$
  $L_N = -b_N \cdot z^{-N}$
  A "zölddel jelölt ág" minden huroknak ága, ezért nincsenek független hurokpárok:
  $\Delta = 1 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2} + ... + b_N \cdot z^{-N}$.

- N+1 db. direkt út van:
  $D_1 = a_0$
  $D_2 = a_1 \cdot z^{-1}$
  $\vdots$
  $D_{N+1} = a_N \cdot z^{-N}$

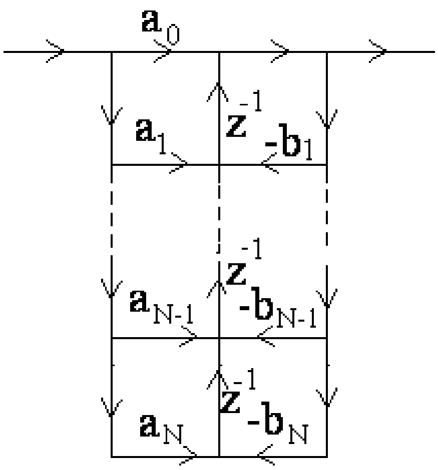
- Mivel van olyan ág, mely minden direkt útban és hurokban szerepel, így minden ut törlése után maradó részgráf hurokmentes: $\Delta_i = 1$, $i = 1...N+1$.

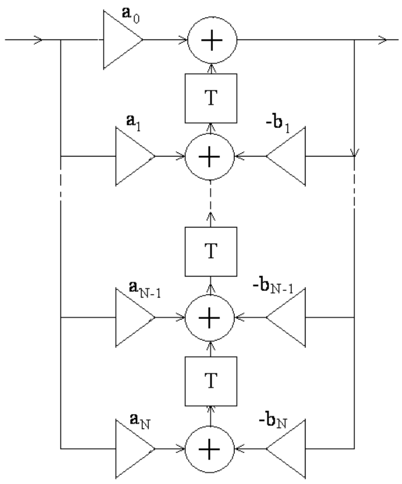
- $\mathrm{H}(z)=\dfrac{\sum_i D_i(z)\cdot\Delta_i(z)}{\Delta(z)}=\dfrac{a_0+a_1\mathrm{z}^{-1}+...+a_N\mathrm{z}^{-N}}{1+b_1\mathrm{z}^{-1}+...+b_N\mathrm{z}^{-N}}$.

Tehát a D1 struktúra bármely ARMA rendszerhez egyértelműen létező direkt és kanonikus struktúra.

D2 struktúra

A D2 struktúra a D1 hálózat transzponáltja, így ugyanazok igazak rá, mint amit a D1 kapcsán megállapítottunk.



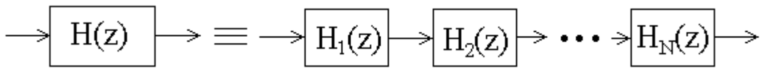


3.4.4 Kaszkádstruktúra

Tegyük föl, hogy H(z) számlálója és nevezője azonos fokszámú (N). Ekkor H(z) felírható N elsőfokú alaptag szorzataként:

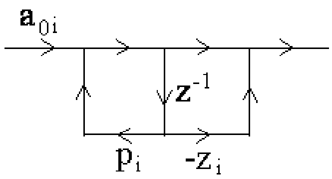
$$\mathrm{H}(z)=\frac{\mathrm{A}(z^{-1})}{\mathrm{B}(z^{-1})}=\frac{a_0+a_1\cdot z^{-1}+...+a_N\cdot z^{-N}}{1+b_1\cdot z^{-1}+...+b_N\cdot z^{-N}}=\frac{a_0\cdot\prod_{i=1}^{N}(1-\mathrm{z_i}\cdot z^{-1})}{\prod_{j=1}^{N}(1-\mathrm{p_j}\cdot z^{-1})}=\prod_{i=1}^{N}a_{0i}\frac{1-\mathrm{z_i}\cdot z^{-1}}{1-\mathrm{p_j}\cdot z^{-1}},$$

ahol $\prod_{i=1}^{N} a_{0i} = a_0$,



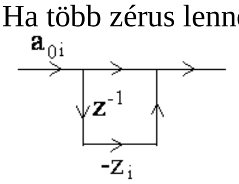
Vagyis $\mathrm{H}(z) = \prod_{i=1}^{N} \mathrm{H}_i(z)$, ahol $\mathrm{H}_i(z) = a_{0i} \dfrac{1 - \mathrm{z_i} \cdot z^{-1}}{1 - \mathrm{p_j} \cdot z^{-1}}$.

Minden $\mathrm{H_i}(z)$ realizálható pl. D1 struktúrával:

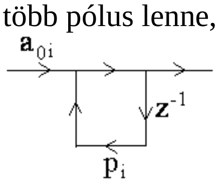


Megjegyzések:

–· Ha több zérus lenne, mint pólus, akkor lesz olyan doboz, amely csak zérust tartalmaz:



Ha több pólus lenne, mint zérus, akkor lesz olyan doboz, amely csak pólust tartalmaz:



– Ez kanonikus struktúra, mert ez is minimális számú késleltetőt tartalmaz. Azonban nem direkt struktúra, mert nem az a_0, a_1,..., a_N és b_1, b_2,..., b_N együtthatók szerepelnek benne, hanem a $\mathrm{p_i}$ pólusok és $\mathrm{z_i}$ zérusok.

– A kaszkádstruktúra nem egyértelmű, mert
- nagyon sokféleképpen lehet az egyes alaptagokban megvalósított zérusokat és pólusokat összepárosítani,
- az alaptagok többféle sorrendben összekapcsolhatók és
- az a_0 erősítés többféleképpen szétosztható az a_{0i}-kre.

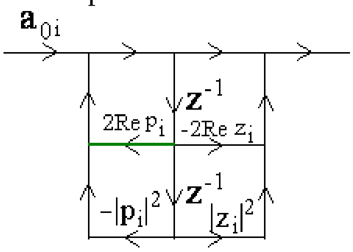
Az ekvivalens transzferfüggvényű lehetséges megoldások közül kiválasztható az egyéb szempontok (éezékenység, dinamika, zaj, stb.) szerint optimalizált változat.

– Adaptív rendszereknél igen előnyös megvalósítás, mert a pólusok és zérusok az együtthatók. Ezáltal könnyen kézben lehet tartani, hogy a pólusok az egységkörön belül maradjanak.

– Valós rendszereknél a komplex gyökök konjugált párjaikkal lépnek fel. Ahhoz, hogy a megvalósítás során ne kelljen komplex együtthatókkal számolni, konjugált-komplex gyökpárokat két elsőfokú alaptag helyett egy *másodfokú alaptaggal* valósítsuk meg (biquad):

$$\mathrm{H}_{\mathrm{i}}^{(\mathrm{II})}(\mathrm{z}) = a_{0i} \cdot \frac{\left(1 - z_i \cdot z^{-1}\right)\cdot\left(1 - z_i^* \cdot z^{-1}\right)}{\left(1 - p_i \cdot z^{-1}\right)\cdot\left(1 - p_i^* \cdot z^{-1}\right)} = a_{0i}\, \frac{1 - 2\,\mathrm{Re}\{z_i\} z^{-1} + |z_i|^2 z^{-2}}{1 - 2\,\mathrm{Re}\{p_i\} z^{-1} + |p_i|^2 z^{-2}}\,.$$

Ennek pl. a D1 struktúrával történő megvalósítása:

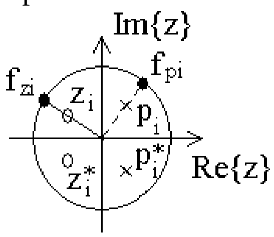


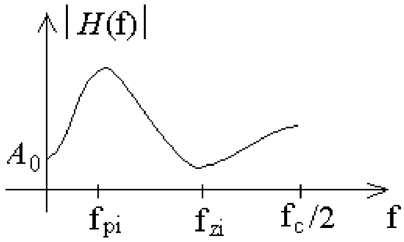
Vezessük be a következő jelöléseket:

$\mathrm{p}_i = \mathrm{r}_i \mathrm{e}^{\mathrm{j}\varphi_i}$, $\varphi_\mathrm{i} = 2\pi \mathrm{f}_{\mathrm{pi}}\mathrm{T}$,

$\mathrm{z}_i = \mathrm{d}_i \mathrm{e}^{\mathrm{j}\vartheta_i}$, $\vartheta_i = 2\pi \mathrm{f}_{\mathrm{zi}}\mathrm{T}$,

és rajzoljuk fel a pólis-zérus elrendezés segítségével a másodfokú alaptag amplitúdókarakterisztikáját!





ahol $A_0 = \dfrac{1 - 2\mathrm{d}_i \cos 2\pi \mathrm{f}_{zi}\mathrm{T} + \mathrm{d}_i^2}{1 - 2\mathrm{p}_i \cos 2\pi \mathrm{f}_{pi}\mathrm{T} + \mathrm{p}_i^2}$. Ilyen típusú karakterisztikák szorzataként áll elő a hálózat teljes szűrőkarakterisztikája.

3.4.5 Párhuzamos struktúra

A H(z) racionális törtfüggvény nemcsak hányados, vagy szorzat alakban, hanem összeg alakban is felírható. Ehhez a parciális törtekre felbontás módszerével juthatunk (az egyszerűség kedvéért feltesszük, hogy a pólusok egyszeresek):

$$\mathrm{H}(z) = \frac{a_0 + a_1 \mathrm{z}^{-1} + ... + a_M \mathrm{z}^{-M}}{1 + b_1 \mathrm{z}^{-1} + ... + b_N \mathrm{z}^{-N}} = \sum_{i=1}^{N} \frac{r_i}{1 - p_i z^{-1}}, \text{ ha } M < N.$$

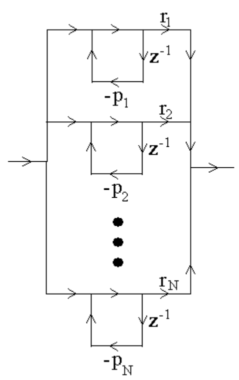
Így a hálózatot egy-egy pólust megvalósító alaptagok párhuzamos kapcsolásával építjük fel.

Ekkor, tehát

$$\mathrm{Y(z)} = \sum_{\mathrm{i}} \mathrm{H_i(z)X(z)},$$

$$\text{ahol } \mathrm{H_i}(z) = \frac{r_i}{1 - p_i z^{-1}}.$$

Az egyenletet jelfolyam gráffal felrajzolva kapjuk a párhuzamos struktúrát:



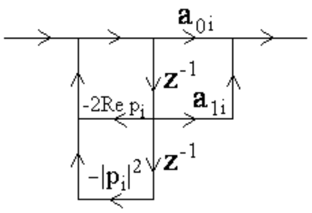
A párhuzamos struktúra kanonikus, azaz minimális késleltető számú. Együtthatói a pólusokkal és a hozzájuk tartozó reziduumokkal vannak kapcsolatban.

Konjugált-komplex póluspárok esetén ahhoz, hogy valós együtthatókkal (és jelekkel) dolgozhassunk, másodfokú alaptagokat kell alkalmazni.

Legyen $p_i = x_{pi} + \mathrm{j}y_{pi}$ és $r_i = x_{ri} + \mathrm{j}y_{ri}$. Továbbá

$$\left(\frac{r_i}{1-p_i z^{-1}}+\frac{r_i^*}{1-p_i^* z^{-1}}\right)=\frac{r_i+r_i^*-\left(r_i p_i^*+r_i^* p_i\right)z^{-1}}{1-2\operatorname{Re}p_i\cdot z^{-1}+\left|p_i\right|^2 z^{-2}}=\frac{a_{0i}+a_{1i}z^{-1}}{1-2\operatorname{Re}p_i\cdot z^{-1}+\left|p_i\right|^2 z^{-2}}$$

Mivel $r_i+r_i^*=2x_{ri}$,
$r_i p_i^*=x_{ri}x_{pi}+y_{ri}y_{pi}+\mathrm{j}\left(y_{ri}x_{pi}-x_{ri}y_{pi}\right)$ és
$r_i^* p_i=x_{ri}x_{pi}+y_{ri}y_{pi}+\mathrm{j}\left(-y_{ri}x_{pi}+x_{ri}y_{pi}\right)$, a fenti másodfokú alaptagnak már valósak az együtthatói: $a_{0i}=2x_{ri}$ és $a_{1i}=2(x_{ri}x_{pi}+y_{ri}y_{pi})$.



Ha $p_i=\mathrm{e}^{\mathrm{j}2\pi f_{pi}T}$ $(\left|p_i\right|=1)$, akkor a fenti másodfokú alaptag egy f_{pi} frekvenciájú csillapítatlan oszcillátor, egyébként a stabil (egységkörön belüli) pólusokhoz csillapított (exponenciálisan lecsengő) oszcillátorok tartoznak. Ezt a struktúrát **rezonátoros struktúrá**nak is nevezik, a bemeneti számsorozat, egyidejűleg gerjeszti a pólusokhoz tartozó rezonátorokat, a kimenet a gerjesztett rezonátorok jeleinek összege.

3.4.6 Lagrange-féle interpoláció struktúra

Induljunk ki a következő interpolációs feladatból:

Keressük meg azt a H(z) függvény, mely a komplex sík N darab, adott $z_1, z_2, \ldots z_N$ pontjaiban adott $h_1, h_2, \ldots h_N$ értékeket veszi fel, azaz

$h_i = H(z_i)$, i = 1, 2, ... N.

Kitűzött feladatunk megoldása az úgynevezett Lagrange-féle interpolációs képlettel adott függvény:

$$H(z)=P(z)\sum_{i=1}^{N}H_i(z),$$

ahol $P(z)=\prod_{i=1}^{N}\left(1-z_i/z\right)$, $H_i(z)=\dfrac{h_i}{P_i(z_i)\left(1-z_i/z\right)}$, $P_i(z)=\prod_{\substack{k=1\\k\neq i}}^{N}\left(1-z_k/z\right)$

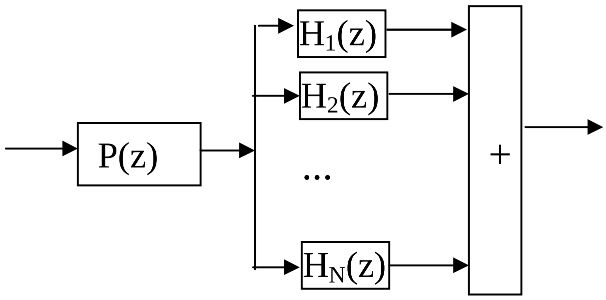
$P(z)\ z^{-1}$ –nek N-ed fokú polinomja, melynek gyökei z_i , i = 1, 2, ... N.

$P_i(z)\ z^{-1}$ –nek (N-1)-ed fokú polinomja, gyökei z_k -k, k = 1, 2, ... N. kivéve z_i –t. $i \neq k$.

$P_i(z_i)$ egy szám, a $P_i(z)$ polinom, $z=z_i$ –nél felvett értéke,

$H_i(z)$ z –nek elsőfokú törtfüggvénye, melynek z_i –nél van pólusa.

Ezek után nyilvánvaló, hogy $P(z)H_i(z)$ z –nek (N-1)-ed fokú polinomjával egyenlő, melynek értéke z_i –nél h_i és a többi z_k $k \neq i$ interpolációs pont felett pedig nulla (itt vannak a z_i –hez tartozó gyöktényezővel történt egyszerűsítés után maradó zérusai) és P(z) is egy (N-1)-ed fokú polinom, mely az eredeti interpolációs feladat megoldása.



A Lagrange struktúra, tehát egy N-ed fokú FIR szűrő (akár transzverzális, akár kaszkád struktúrában realizálva) és egy szintén N-ed fokú párhuzamos struktúrában realizálható ARMA szűrő kaszkád eredője, mely ekvivalens egy (N-1)-ed fokú FIR szűrővel..

Ez a struktúra nyilvánvalóan nem kanonikus, további érdekessége, hogy rekurzív hálózattal valósít meg FIR szűrőt.

3.4.7 Frekvencia-mintavételező struktúra

Az előző pontban megismert Lagrange interpolációt és struktúrát alkalmazzuk arra a speciális esetre, amikor a z_i interpolációs pontokat a komplex sík N-ed rendű egység gyökeinek választjuk:

$$z_k = e^{j\,k\,2\pi/N},\quad k = 0,1, \ldots (N-1).$$

Ekkor a $h_k = H(\ z_k\ ) = H\ (\ e^{j\,k\,2\pi/N})$, k = 0,1, ... (N-1) értékek egyenlők a

$\mathit{H}(f) = H(z=e^{j\,2\pi fT}\ )$ frekvencia karakterisztika $f_k = k\ F/N$, k = 0,1, ... (N-1) frekvenciák feletti értékeivel:

$h_k = H(z_k) = H(e^{j 2\pi k F/N T}) = H(f_k) = H_k$, k = 0,1, ... (N-1)

Ekkor a P(z) FIR szűrő az N-ed fokú $P(z) = (1 – z^{-N})$ fésű szűrő (zérusai az egység-gyökök), a parallel kapcsolódó $H_k(z)$ szűrők pedig komplex z –sík egységkörén az f_k = k F/N frekvenciáknál pólussal rendelkező veszteségmentes rezonátorok, melyeknek előre csatoló együtthatóival állíthatóak be a szűrő átvitelének, adott frekvenciás H_k komplex értékei:

$$H(z) = \left(1 - z^{-N}\right)\frac{1}{N}\sum_{k=0}^{N-1}\frac{h_k}{1 - e^{j2\pi\frac{k}{N}}z^{-1}}$$

Ugyanis a Lagrange interpolációs formulába helyettesítve kapjuk:

$$H(z) = P(z)\sum_{k=0}^{N-1} H_k(z),$$

ahol $$P(z) = \prod_{k=0}^{N-1}\left(1 - z_k/z\right) = \prod_{k=0}^{N-1}\left(1 - e^{j2\pi\frac{k}{N}}/z\right) = \left(1 - z^{-N}\right),$$

$$H_k(z) = \frac{H_k}{P_k(z_k)\left(1 - z_{ik}/z\right)}, \qquad P_i(z) = \prod_{\substack{k=1\\k\neq i}}^{N}\left(1 - z_k/z\right)$$