

Tarmo Lipping

DIGITAALNE
SIGNAALI- JA
PILDITÖÖTLUS

Loengukonspekt

Digitaalne signaali- ja pilditöötlus

Loengukonspekt

Tarmo Lipping

22. jaanuar 2007. a.

Sisukord

1	Sissejuhatus	5
1.1	Füsioloogiliste signaalide salvestamine	6
2	Diskreetsed signaalid ja süsteemid aja tasandil	9
2.1	Diskreetsed elementaarfunktsioonid	9
2.2	Diskreetsed süsteemid ja nende omadused	10
2.2.1	Lineaarsus	12
2.2.2	Aja (nihke) suhtes invariantus	13
2.2.3	Kausaalsus	13
2.2.4	Stabiilsus	13
2.3	Lineaarsed aja suhtes invariantid (LTI) süsteemid	14
2.4	LTI-süsteemide omadusi	15
2.5	Diferentsvõrrandi abil väljendatavad LTI-süsteemid	15
3	Diskreetsed signaalid ja süsteemid sageduse tasandil	17
3.1	Signaalide esitamine Fourier' teisenduse abil	17
3.1.1	Pideva perioodilise signaali sagedusesitus	17
3.1.2	Pideva mitteperioodilise signaali sagedusesitus	18
3.1.3	Diskreetse mitteperioodilise signaali sagedusesitus	19
3.1.4	Diskreetse perioodilise signaali sagedusesitus, DFT	20
3.2	Sagedusesitus süsteemide analüüsil	25

3.3	Pideva signaali diskreetimine ja selle mõju sagedusesitusele	26
3.4	Diskreetse Fourier' teisenduse arvutamine	28
4	LTI-süsteemide analüüs z-tasandil	31
4.1	Z-teisendus ja selle koonduvuspiirkond	31
4.2	Diferentsvõrrandi abil esitatavad süsteemid z-tasandil	33
4.3	LTI-süsteemide omadused z-tasandil	34
4.4	Nullide ja pooluste paiknemise seos sageduskarakteristikuga	36
4.5	<i>All-pass</i> -süsteemid	37
4.6	Amplituudi- ja faasikarakteristiku omavaheline sõltuvus	40
4.7	Minimaalse faasiga süsteemid	40
4.8	Lineaarne faasikarakteristik	43
5	Filtrite disainimine	49
5.1	FIR-filtrite disainimine	49
5.1.1	FIR-filtrite disainimine akna meetodil	49
5.1.2	Kaiseri aknafunktsioonide pere	50
5.1.3	FIR-filtrite disainimine Remezi meetodil	51
5.2	IIR-filtrite disainimine	52
5.2.1	Analoogsete filtrite põhiklassid	53
5.2.2	IIR-filtrite disainimine impulsskaja diskreetimise teel	55
5.2.3	IIR-filtrite disainimine bilineaarse teisenduse meetodil	57
6	Meditseenilise kujutise töötlemine	61
6.1	Kujutise formaadid	61
6.2	Kujutise filtreerimine	63
6.3	Kahedimensiooniline Fourier' teisendus ja konvolutsioon	65
6.4	Kujutise ruumilised teisendused	66
6.4.1	Afiinsed teisendused	67
6.4.2	Projektiivsed teisendused	70
6.5	Kujutise registreerimine	71

1 Sissejuhatus

Vaevalt leidub tänapäeval eluvaldkonda, mis ei kasutaks ühel või teisel moel signaalitöötluste võimalusi. Käesoleva materjali eesmärk on tutvustada digitaalse signaalitöötluste teoreetilisi aluseid. Materjal on mõeldud kasutamiseks biomeditsiinitehnika eriala tudengitele ja seetõttu lähtutakse siin digitaalse signaalitöötluste rakendusvõimalustest meditsiinis ja meditsiinitehnikas.

Meditsiinitehnika on üks huvitavamaid digitaalse signaalitöötluste rakendusvaldkondi. Seda eelkõige seetõttu, et inimorganismist mõõdetavate parameetrite hulk on äärmiselt mitmekesine ja mõõdetavate signaalide töötlemisel leiab kasutust kogu digitaalse signaalitöötluste meetodite arsenal. Füsioloogiliste signaalide töötlustes leidub harva üldlevinud standardeid, nagu näiteks JPEG kujutiste või mp3 audiosignaalide kompressioonil. Biomeditsiinitehnikas kasutatavate seadmete tootjad töötavad tavaliselt oma seadmes kasutatavad signaalitöötlustsalgoritmid ise välja ja enamasti jätaavad nende täpse kirjelduse endi teada. See teebki antud valdkonna huvitavaks ja väljakutsuvaks.

Käesolevas materjalis käsitletakse kõigepealt digitaalseid signaale ja süsteeme aja tasandil. Tuuakse näiteid elementaarfunktsioonidest, defineeritakse mitmeid digitaalsete süsteemide omadusi ning jõutakse välja lineaarse diferentsvõrrandi abil esitatavate süsteemideni. Seejärel vaadeldakse diskreetsete signaalide ja süsteemide esitamist sageduse tasandil Fourier' teisenduse abil. Kuna digitaalsete süsteemide disainimisel on tihti aluseks analoogsed süsteemid, siis on oluline käsitleda ka seda, mis toimub sageduse tasandil, kui pidev signaal ajas diskreeditakse.

Neljandas peatükis käsitletakse Z-teisendust. Selgub, miks just lineaarse diferentsvõrrandi abil esitatavad süsteemid on eriti kasutuskõlblikud. Vaadeldakse, kuidas süsteemide omadused on seotud süsteemi nullide ja pooluste asukohtadega, samuti seda, millisel puhul on süsteemil lineaarne faasikarakteristik. 5. ja 6. peatükis tutvutakse digitaalsete filtrite disainimise põhilisemate algoritmidega.

Viimane peatükk käsitleb kujutise töötlemist. Kujutist võib vaadelda kahedimensionaalse signaalina ja kõiki varem käsitletud meetodeid on võimalik rakendada ka kujutistele, üldistades neid enne kahedimensioonilis-

teks. Praktiliselt kasutatakse kujutise töötlemisel tihti hulka spetsiaalseid operatsioone, mis on osutunud otstarbekaks soovitud visuaalse tulemuse saavutamiseks. Sellised operatsioonid on näiteks objektide servade detekteerimine, kujutise segmenteerimine, kujutise ruumilised teisendused jne.

Diskreetsete signaalide töötlemise põhitõdede kohta leidub rahvusvahelises kirjanduses arvukalt materjali. Nii-öelda klassikalisteks kujunenud õpikutest võiks mainida kahte:

- Alan V. Oppenheim, Ronald W. Schaffer ja John R. Buck. *Discrete-Time Signal Processing*. Teine väljaanne, Prentice-Hall, 1999.
- Emmanuel Ifeachor ja Barrie Jervis. *Digital Signal Processing – a Practical approach*. Teine väljaanne, Pearson Education, 2002.

Pilditötluse osas on käesolevat materjali kirjutades võetud aluseks järgmine teos:

- John L. Semmelow. *Biosignal and Biomedical Image Processing: MATLAB-Based Applications*. Marcel Dekker Inc., 2004.

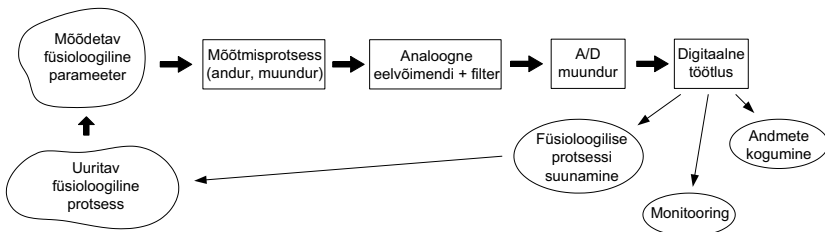
Käesoleva materjaliga samaaegselt on valmistatud ette Digitaalse signaali- ja pilditötluse harjutuste materjal, mille autoriks on Rain Ferenets. Järgnevas kasutan viiteid mainitud materjalile teemade puhul, mida on seal põhilikult ja illustreerivalt käsitletud.

1.1 Füsioloogiliste signaalide salvestamine

Signaalide töötlemise eesmärk on otseselt seotud sellega, milline on antud signaali allikas, kuidas signaal on saadud ning milleks töötlemise tulemusi soovitakse kasutada. Joonisel 1.1 on kujutatud operatsioonide ahel, mis üldjuhul kaasneb signaali digitaalse töötlemisega. Järgnevas käsitletakse neid operatsioone lähemalt.

- Tihti füsioloogilist parameetrit, mille kohta soovime informatsiooni saada, pole võimalik otseselt mõõta. Selle asemel on võimalik mõõta mõnda teist parameetrit, mille väärtus korreleerub meid huvitava protsessiga. Näitena võib tuua ajupiirkonna aktiivsuse mõõtmise lähi-infrapunakiirgusel põhineva meetodi NIRS (*Near Infrared Spectroscopy*) abil. Selle meetodiga mõõdetakse otseselt verevarustuse intensiivsust mõõtmisalusel aju piirkonnas. Kuna on teada, et närvirakkude aktiveerudes suureneb ka nende verevarustus, on NIRSi abil võimalik kaudselt hinnata närvirakkude funktsionaalset aktiivsust.

- Selleks et signaale töödelda, peavad nad olema elektrilisel kujul. Seetõttu tuleb sellised füsioloogilised parameetrid nagu vererõhk, hapniku osarõhk väljahingatavas õhus, südameklappidest lähtuv heli jne kõigepealt muundada elektrilisele kujule. Selleks on olemas mitmesugused muundurid (*transducer*). Ka bioelektriliste signaalide puhul (näiteks elektroentsefalogramm (EEG), elektrokardiogramm (EKG) jne) on vaja ionide liikumine organismis muundada elektronide liikumiseks juhtmes ehk elektrivooluks. See operatsioon tehakse elektrootodide abil. Signaali salvestamisel põhjustavad kasutatavad andurid ja muundurid mitmeid moonutusi signaali kujus, seega on mõõtmisprotsessi tundmine signaalide töötlemise seisukohalt oluline. Näiteks madalad sagedused EEG-signaalis võivad olla pärit mõõdetavast allikast ehk ajurakkude tegevusest, aga võivad olla tingitud ka elektrootodi ja peanaha vahelise elektrilise kontakti muutustest.
- Enamik füsioloogilistest signaalidest on sedavõrd nõrgad, et enne digitaalseks muutmist on neid vaja analoogselt võimendada. Samuti on signaalist vaja eemaldada nii madalad kui ka kõrged sagedused. Vajadus eemaldada madalad sagedused tuleneb sellest, et mõõtmistehnoloogiast tingitult tihti mõõdetud signaal sisaldab suuri alalis-komponendi muutusi, mis muidu põhjustaksid eelvõimendi küllastumise. Kõrged sagedused eemaldatakse selleks, et signaal diskreetimisel ei moonutuks (lähemalt lõigus 3.3). Analooorse signaali filtreerimine mõõtmisel kujutab endast alati kompromissi – mida laiem on mõõdetud signaali sagedusriba, seda rohkem informatsiooni signaal sisaldab, samas peab mõõtmisprotsessil olema hoolikam (näiteks elektrootodide ja naha vahelise kontakti jälgimisel), et signaali kvaliteet ei kannataks. Ka mõõtmisaparatuuri kvaliteedil on siin oma roll.
- Pideva signaali muundamine digitaalseks sisaldab endas kahte operatsiooni: signaali diskreetimist ajas (signaali väärtuste "noppimist" teatud ajavahemiku järel) ning signaali väärtuste kvantiseerimist. Diskreetida tuleb piisavalt kõrge diskreetimissagedusega, et signaalist ei läheks informatsiooni kaduma (vt lõik 3.3). Kvantiseerimise vajadus tuleneb sellest, et digitaalses süsteemis on iga signaali väärtuse esitamiseks võimalik kasutada vaid piiratud arvu bittide (üldiselt kasutatakse 8-, 12- või 16-bitiseid süsteeme) ja seega on esitatavate väärtuste hulk 2^{N_b} , kus N_b on ühe väärtuse esitamiseks kasutada olevate bittide arv. Pidev signaal võib taas omada mis tahes väärtusi. Seega asendatakse kvantiseerimisel pideva signaali väärtused lähima



Joonis 1.1 Füsioloogiliste signaalide salvestamise ja töötlemise operatsioonid

kvantiseerimistaseme väärtusega. Käesolevas materjalis ei vaadelda kvantiseerimisest tulenevaid probleeme ja seega räägitakse tihti diskreetsetest signaalidest, s.o signaalidest, mis on diskreeditud, kuid mille diskreedid võivad saada mis tahes väärtusi.

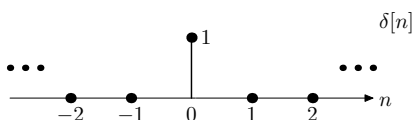
- Füsioloogiliste signaalide töötlemisel võib olle erinevaid eesmärke. Joonisel 1.1 on esitatud kolm võimalikku valdkonda: füsioloogilise protsessi suunamine, patsiendi monitooring ning andmete kogumine. Esimesest kategooriast võib tuua näiteks südame stimulaatori, ravimi manustamise diabeedi korral või biotagasiside (*biofeedback*) kasutamise depressiooni puhul. Patsiendi monitooringu osas on oluline anda võimalikult täpset informatsiooni patsiendi seisundi kohta. Samas peab see info olema lihtsalt tõlgendatav.

2 Diskreetsed signaalid ja süsteemid aja tasandil

2.1 Diskreetseid elementaarfunktsioone

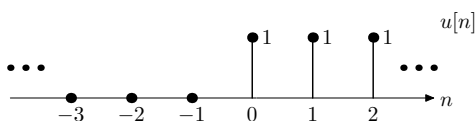
Edaspidi kasutame korduvalt mõningaid diskreetseid elementaarfunktsioone, mida defineerime järgnevalt.

Ühikimpulss $\delta[n]$ on funktsioon, mille väärtus kohas $n = 0$ on 1 ja mujal 0:



$$\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & \text{mujal} \end{cases} \quad (2.1)$$

Ühikhüpe $u[n]$ on funktsioon, mille väärtus kohtades $n \geq 0$ on 1 ja mujal 0:



$$u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} \quad (2.2)$$

Mis tahes diskreetse signaali $x[n]$ võime esitada nihutatud ja skaleeritud ühikimpulsside summana:

$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k] \quad (2.3)$$

Avaldise 2.3 parem pool kujutab endast lihtsalt signaali $x[n]$ väärtuste "väljanõppimist" ükshaaval.

Elementaarfunktsioonid $\delta[n]$ ja $u[n]$ on võimalik avaldada teineteise kaudu järgmiselt:

$$u[n] = \sum_{k=-\infty}^n \delta[k]; \quad \delta[n] = u[n] - u[n-1] \quad (2.4)$$

Oluline koht edaspidises materjalis on veel *diskreetsetel sinusoididel* $\cos(\omega_0 n - \varphi)$. ω_0 on diskreetse sinusoidi sagedus ja φ diskreetse sinusoidi faas. On oluline märkida, et kui pidevate funktsioonide¹ puhul on sageduse ühikuks [rad/s], siis diskreetsete funktsioonide puhul on sageduse ühikuks radiaani diskreedi kohta ehk lihtsalt [rad]. See tuleneb tõsiasiast, et diskreetsete funktsioonide ja signaalide puhul ei ole aeg määratletud, vaid horisontaalteljel on diskreedi järjekorra number. Ajavahemik kahe diskreedi vahel sõltub *diskreetimissagedusest* f_s .

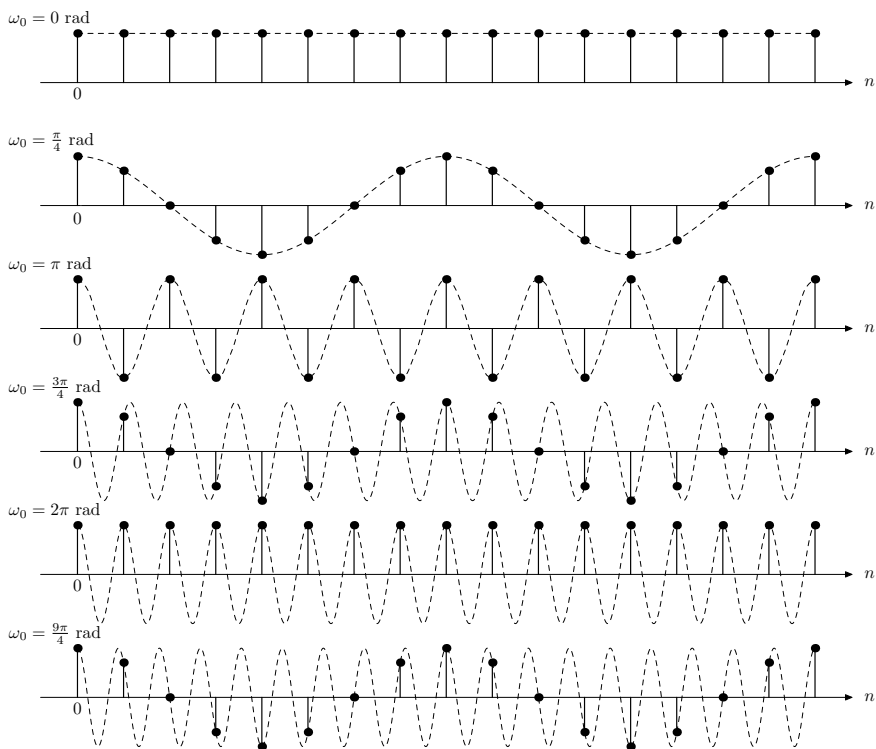
Teine oluline erinevus diskreetse ja analoogse "maailma" vahel ilmneb jooniselt 2.1, kus on kujutatud diskreetne koosinusfunktsioon sagedustel 0 rad, $\frac{\pi}{4}$ rad, π rad, $\frac{3\pi}{4}$ rad, 2π rad ning $\frac{9\pi}{4}$ rad. On näha, et sagedused, mis erinevad üksteisest 2π võrra, annavad täpselt samasuguse diskreetse funktsiooni. See väljendab seda, et diskreetsete signaalide sagedusspekter on perioodiline perioodiga 2π . Jooniselt 2.1 nähtub ka, et sagedused, mis on sümmeetrilised π suhtes, annavad sama diskreetse funktsiooni. See peab paika ainult reaalsete väärtustega funktsioonide korral – sel puhul on diskreetsete signaalide sagedusspekter sümmeetriline π suhtes.

Nende erinevuste esiletoomiseks tähistame edaspidi sagedust pidevate signaalide puhul sümboliga Ω ja diskreetsete signaalide puhul sümboliga ω .

2.2 Diskreetsed süsteemid ja nende omadused

Diskreetsed signaalid kujutavad endast numbrite jada arvuti või mõne muu digitaalse süsteemi mälus. Käesolevas materjalis tähistame diskreetseid signaale $x[n]$ -ga, kus n on signaali diskreedi järjekorra number. Pidevate signaalide puhul kasutame tähistust $x(t)$, kus t on üldjuhul aeg. Kirjanduses siiski alati sellist erinevust diskreetsete ja pidevate signaalide tähistuses ei tehta ja see, kas tegemist on diskreetse või pideva signaaliga, selgub kontekstist.

¹Käesolevas materjalis on mõisted funktsioon ja signaal üldjuhul ekvivalentsed.



Joonis 2.1 Diskreetne koosinusfunktsioon erinevatel sageduse ω_0 väärtustel

Diskreetseks süsteemiks võib üldiselt nimetada mis tahes arvutuslikku algoritmi, mis seob omavahel sisendsignaali väärtuste jada $x[n]$ ja väljundsignaali väärtuste jada $y[n]$. Üldiselt tähistame siin süsteemi rakendamist signaalile operaatori $T\{\cdot\}$ abil:

$$y[n] = T\{x[n]\} \quad (2.5)$$

Näide 2.1 Jooksev keskmine

Jooksev keskmine on väljendatav järgmise arvutusliku algoritmi kaudu:

$$y[n] = \frac{1}{1 + M_1 + M_2} \sum_{k=-M_1}^{M_2} x[n - k] \quad (2.6)$$

Seega tuleb jooksva keskmise puhul väljundsignaali väärtuseks kohal n sisendsignaali $x[n]$ aknasse $\{n - M_2, \dots, n + M_1\}$ jäävate diskreetide keskmine.

Operaatoril $T\{\cdot\}$ ja seega vastaval diskreetsel süsteemil võib olla mitmesuguseid omadusi. Järgnevalt vaatleme neist mõnda olulisemat.

2.2.1 Lineaarsus

Süsteem $T\{\cdot\}$ on lineaarne, kui kehtib

$$T\{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\}, \quad (2.7)$$

kus a ja b on suvalised kordajad ning $x_1[n]$ ja $x_2[n]$ mis tahes kaks diskreetset signaali. Öeldakse, et lineaarse süsteemi puhul kehtib superpositsiooni printsiip, teiste sõnadega: lineaarne süsteem on nii aditiivne kui ka homogeenne.

Näide 2.2 Akumulaator

Akumulaator on süsteem, mis on väljendatav järgmise avaldise abil:

$$y[n] = \sum_{k=-\infty}^n x[k] \quad (2.8)$$

Akumulaator on lineaarne süsteem, kuna

$$\begin{aligned} \sum_{k=-\infty}^n ax_1[n] + bx_2[n] &= \sum_{k=-\infty}^n ax_1[n] + \sum_{k=-\infty}^n bx_2[n] \\ &= a \sum_{k=-\infty}^n x_1[n] + b \sum_{k=-\infty}^n x_2[n] \end{aligned}$$

2.2.2 Aja (nihke) suhtes invariantus

Süsteem $T\{\cdot\}$ on aja suhtes invariantne järgmisel juhul:

$$x[n] \xrightarrow{T} y[n] \Rightarrow x[n - n_0] \xrightarrow{T} y[n - n_0], \quad \forall n_0$$

Näide 2.3 Diskreetimissageduse vähendamine

Diskreetimissageduse vähendamine (kompessioon, ingl.k downsampling) on matemaatiliselt väljendatav järgmiselt:

$$y[n] = x[Mn]$$

Antud süsteem "nopib" sisendsignaalist $x[n]$ välja iga M -nda diskreedi. See süsteem ei ole aja suhtes invariantne, kuna

$$x[n - n_0] \xrightarrow{T} x[Mn - Mn_0] \neq x[Mn - n_0] = y[n - n_0]$$

2.2.3 Kausaalsus

Süsteem on kausaalne, kui tema väljundsignaali väärtus mingil ajahetkel n_0 ($y[n_0]$) sõltub ainult sisendi väärtustest hetkedel $n < n_0$ ja seda iga n_0 korral. Teiste sõnadega – väljundi arvutamiseks praegusel ajahetkel ei vaja me informatsiooni sisendsignaali käitumisest tulevikus. Samuti kausaalse süsteemi puhul, kui $x_1[n] = x_2[n]$, $n < n_0$, siis ka $y_1[n] = y_2[n]$, $n < n_0$, ehk kui kaks sisendit käituvad ekvivalentset ajahetkeni n_0 , siis ka vastavad kausaalse süsteemi väljundid käituvad selle ajahetkeni ekvivalentset.

2.2.4 Stabiilsus

Süsteem on stabiilne, kui iga piiratud sisendi puhul ($|x[n]| \leq B_x < \infty$, $\forall n$) leidub B_y , nii et ($|y[n]| \leq B_y < \infty$, $\forall n$). Teiste sõnadega – iga piiratud sisend annab ka piiratud väljundi.

Järgnevalt vaadeldakse põhiliselt süsteeme, mis on lineaarsed ja aja suhtes invariantse. Kausaalsus on omadus, mis on eelduseks, et süsteemi oleks võimalik realiseerida rangelt reaajas. Süsteem töötab rangelt reaajas, kui ta väljastab väljundi väärtuse kohal n , enne kui sisendi väärtus kohal $n + 1$ on mõõdetud (s.t saabunud süsteemi sisendisse). Stabiilsus on soovitatav omadus selleks, et süsteemi väljund ei läheks küllastusse või et süsteem ei hakkaks genereerima.

2.3 Lineaarsed aja suhtes invariantised (LTI) süsteemid

Lineaarsete aja suhtes invariantsete (edaspidi LTI, ingl.k *Linear Time Invariant*) süsteemide analüüsil ja disainil on olulisi eeliseid, võrreldes süsteemidega, millel nimetatud omadusi pole. Avaldame süsteemi väljundi, kasutades valemit 2.3:

$$\begin{aligned}
 y[n] &= \mathbb{T}\{x[n]\} \\
 &= \mathbb{T}\left\{\sum_{k=-\infty}^{\infty} x[k]\delta[n-k]\right\} \\
 \mathbb{T}\{\cdot\} \text{ lineaarne} &= \sum_{k=-\infty}^{\infty} x[k]\mathbb{T}\{\delta[n-k]\} \\
 &= \sum_{k=-\infty}^{\infty} x[k]h_k[n] \\
 \mathbb{T}\{\cdot\} \text{ ajas invariantne} &= \sum_{k=-\infty}^{\infty} x[k]h[n-k] \tag{2.9}
 \end{aligned}$$

Siin on $h[n]$ süsteemi impulsskaja (süsteemi väljund, kui sisendis on ühikimpulss). On oluline märkida, et juhul kui süsteem ei ole ajas invariantne, sõltub impulsskaja kuju üldjuhul sellest, millisel ajahetkel sisendis ühikimpulss esines. Matemaatiliselt võib seda väljendada nii:

$$\delta[n-k] \xrightarrow{\mathbb{T}} h_k[n]$$

Eeldades aja suhtes invariantset, ei sõltu impulsskaja kuju ühikimpulsi ajahetkest, vaid nihutatud ühikimpulsi puhul saame väljundisse nihutatud impulsskaja:

$$\delta[n-k] \xrightarrow{\mathbb{T}} h[n-k]$$

Avaldises 2.9 saadud tulemus tähendab seda, et *LTI-süsteemid on täielikult määratletud nende impulsskaja abil*. Teiste sõnadega – kui teame LTI-süsteemi vastet ühikimpulsile, võime leida tema vaste mis tahes sisendsignaalile. Tehe, mis seob LTI-süsteemi väljundi tema sisendi ja impulsskajaga, on konvolutsioon. Konvolutsiooni tulemus ei sõltu operandide järjekorrast:

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = \sum_{k=-\infty}^{\infty} h[k]x[n-k] = h[n] * x[n] \tag{2.10}$$

2.4 LTI-süsteemide omadusi

Järgnevalt vaatleme mõningaid LTI-süsteemide omadusi ja LTI-süsteemidega seotud mõisteid.

LTI-süsteemide jadaühendus. Kahe LTI-süsteemi (impulsskajadega $h_1[n]$ ja $h_2[n]$) jadaühendus on ekvivalentne süsteemiga, mille impulsskaja on nende jadamise ühendatud süsteemide impulsskajade konvolutsioon: $h[n] = h_1[n] * h_2[n]$.

LTI-süsteemide paralleelühendus. Kahe LTI-süsteemi (impulsskajadega $h_1[n]$ ja $h_2[n]$) paralleelühendus on ekvivalentne süsteemiga, mille impulsskaja on nende paralleelselt ühendatud süsteemide impulsskajade summa: $h[n] = h_1[n] + h_2[n]$.

Kausaalsed LTI-süsteemid. Kausaalse LTI-süsteemi puhul kehtib $h[n] = 0$, $n < 0$.

Stabiilsed LTI-süsteemid. Stabiilse LTI-süsteemi puhul kehtib $\sum_{n=-\infty}^{\infty} |h[n]| < \infty$. Teiste sõnadega, stabiilse LTI-süsteemi impulsskaja on absoluutselt summeeruv.

Süsteemi $h[n]$ **pöörd süsteemiks** $h_i[n]$ nimetatakse süsteemi, mis tühistab täielikult antud süsteemi mõju. LTI-süsteemide puhul kehtib $h_1[n] * h_i[n] = \delta[n]$. Süsteem, mille impulsskaja on $\delta[n]$, kopeerib sisendsignaali muutmatusena väljundisse, teiste sõnadega, selline süsteem ei avalda sisendsignaali mingit mõju.

Diskreetsete LTI-süsteemide disainimise seisukohalt on oluline süsteemide jaotus **lõpliku impulsskajaga** ehk **FIR-** (ingl.k *Finite Impulse Response*) ja **lõpmatu impulsskajaga** ehk **IIR-** (ingl.k *Infinite Impulse Response*) süsteemideks.

2.5 Diferentsvõrrandi abil väljendatavad LTI-süsteemid

Järgnevalt piirame meid huvitavate süsteemide hulka veelgi. Praktilist tähtsust omavad sellised LTI-süsteemid, mis on esitatavad diferentsvõrrandi kujul:

$$\sum_{k=0}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k] \quad (2.11)$$

Kui vasakult poolt viia kõik muud liikmed peale liikme $k = 0$ paremale poole ning jagada mõlemad pooled läbi kordajaga a_0 , on tulemuseks diferentsvõrrandi kuju, mis kujutab endast antud süsteemi realiseerimise algoritmi (s.t algoritmi väljundi arvutamiseks ajahetkel n eelmiste väljundi väärtuste ning praeguse ja eelmiste sisendi väärtuste kaalutud summana):

$$y[n] = \sum_{k=0}^M b_k x[n-k] - \sum_{k=1}^N a_k y[n-k] \quad (2.12)$$

On oluline märkida, et esitatud valemites alustatakse summeerimist kohast $k = 0$ ja seega hõlmab selline diferentsvõrrandi kuju kausaalseid süsteeme.

Juhul kui eelnevates valemites $N = 1$, s.t väljundi $y[n]$ arvutamisel ei kasutata väljundsignaali varasemaid väärtusi, on süsteem realiseeritud **mitterekursiivselt**. Kui aga $N > 1$, on tegemist **rekursiivse** süsteemiga. Rekursiivsus puudutab süsteemi realiseerimist – tihti on FIR-süsteeme võimalik realiseerida nii rekursiivselt kui ka mitterekursiivselt. Samas ei ole IIR-süsteemi võimalik realiseerida mitterekursiivselt, sest sel juhul oleks diferentsvõrrandis b_k kordajate arv lõpmatu.

Näide 2.4 Jooksva keskmise realiseerimine rekursiivselt ja mitterekursiivselt

Valemi 2.6 abil esitatud süsteemi (jooksev keskmine) on võimalik realiseerida nii rekursiivselt kui ka mitterekursiivselt. Vaatleme süsteemi, mille puhul $M_1 = 0$ ja $M_2 = 4$. Sellist süsteemi on võimalik rekursiivselt esitada võrrandiga

$$y[n] = \sum_{k=0}^4 \frac{1}{5} x[n-k]$$

Kui võrrelda seda esitust diferentsvõrrandi üldkujuga 2.11, näeme et antud juhul $N = 1$, $a_0 = 1$, $M = 4$, $b_0 \dots b_4 = \frac{1}{5}$.

Sama süsteemi võime realiseerida ka rekursiivselt:

$$y[n] = y[n-1] - \frac{1}{5}x[n-5] + \frac{1}{5}x[n],$$

mille puhul $N = 2$, $a_0 = 1$, $a_1 = -1$, $b_0 = \frac{1}{5}$, $b_5 = -\frac{1}{5}$, $b_1 \dots b_4 = 0$.

Erinevad esitus- (ja realiseerimis-) moodused võivad olla erineva arvutusliku keerukusega. Samuti võivad signaali ning süsteemi kordajate kvantiseerimisest tekkivad vead mõjuda sama süsteemi eri realiseerimise puhul erinevalt.

3 Diskreetsed signaalid ja süsteemid sageduse tasandil

3.1 Signaalide esitamine Fourier' teisenduse abil

Fourier' teisenduse abil on mis tahes signaali võimalik üheselt esitada sinusoidide summana. Signaali spekter (signaali Fourier' teisendus) näitab, kui suure amplituudiga ja millises faasis erineva sagedusega sinusoidide antud signaali esitus sisaldab. Seega on signaali spekter sageduse funktsioon.

Fourier' teisenduse kuju sõltub sellest, kas esitatav signaal on pidev või diskreetne ning kas ta on perioodiline või mitteperioodiline. Kuna perioodilise signaali sagedusesitus on diskreetne, sisaldades vaid harmoonilisi sagedusi, räägitakse sel puhul enamasti Fourier' jadast. Järgnevalt vaatleme, millise kuju signaali Fourier' teisendus saab neljal eelnevalt mainitud juhul. Teema käsitus ei pretendeeri siinkohal täielikkusele, vaid eesmärgiks on aidata mõista signaali sagedusesituse olemust.

3.1.1 Pideva perioodilise signaali sagedusesitus

Võime esitada mis tahes pideva perioodilise signaali sinusoidide summana:

$$\tilde{x}(t) = \sum_{k=1}^{\infty} A_k \cos(\Omega_k t + \theta_k) \quad (3.1)$$

Antud esituses on Ω_k sinusoidi sagedus, A_k on vastava sinusoidi amplituud ja θ_k faas. Laineline joon signaali kohal \tilde{x} tähistab signaali perioodilisust.

Koosinusfunktsiooni võime Euleri valemi abil kirjutada järgmiselt:

$$\cos(\alpha) = \frac{e^{j\alpha} + e^{-j\alpha}}{2} \quad (3.2)$$

Kui valemit 3.2 kasutada valemis 3.1, saame

$$\begin{aligned}\tilde{x}(t) &= \sum_{k=1}^{\infty} \left[\frac{A_k}{2} e^{j(\Omega_k t + \theta_k)} + \frac{A_k}{2} e^{-j(\Omega_k t + \theta_k)} \right] \\ &= \sum_{k=1}^{\infty} \left[\frac{A_k}{2} e^{j\theta_k} e^{j\Omega_k t} + \frac{A_k}{2} e^{-j\theta_k} e^{-j\Omega_k t} \right]\end{aligned}\quad (3.3)$$

Kui defineerida kordajad

$$\begin{aligned}c_k &= \frac{A_k}{2} e^{j\theta_k} \\ c_0 &= 0 \quad (\text{signaali alaliskomponent}) \\ c_{-k} &= \frac{A_k}{2} e^{-j\theta_k}\end{aligned}$$

ja määratleda $-\Omega_k = \Omega_{-k}$, võime summa 3.3 kirjutada kompaktsemalt:

$$\tilde{x}(t) = \sum_{k=-\infty}^{\infty} c_k e^{j\Omega_k t} \quad (3.4)$$

Valemis 3.4 toodud esitus sisaldab negatiivseid sagedusi. Füüsikaliselt võttes ei või sagedus olla negatiivne; selline kompleksne esitus on siiski süsteemide ja signaalide analüüsi seisukohalt väga otstarbekas.

Kui signaali $\tilde{x}(t)$ periood on T , siis summa 3.4 sisaldab ainult põhisageduse $\Omega_0 = \frac{1}{T}$ kordseid ja võime summa kirjutada järgmisel kujul:

$$\tilde{x}(t) = \sum_{k=-\infty}^{\infty} c_k e^{jk\Omega_0 t} \quad (3.5)$$

Kordajad c_k võime signaali $\tilde{x}(t)$ kaudu leida

$$c_k = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} \tilde{x}(t) e^{-jk\Omega_0 t} dt, \quad k = 0, \pm 1, \pm 2, \dots \quad (3.6)$$

Valemid 3.5 ja 3.6 moodustavad Fourier' teisenduse paari pidevatele perioodilistele signaalidele.

3.1.2 Pideva mitteperioodilise signaali sagedusesitus

Järgnevalt vaatleme Fourier' teisendust pidevale mitteperioodilisele signaalile. Seda on kõige lihtsam teha, käsitledes mitteperioodilist signaali

kui perioodilist signaali, mille periood on lõpmatu: $T \rightarrow \infty$. Kui periood läheneb lõpmatusale, siis harmooniliste sageduste vahekaugus sagedusteljel läheneb nullile ehk sagedusesitus muutub pidevaks. Valemi 3.6 T -ga läbikorrutatud vasakut poolt võime seega tähistada pideva funktsiooniga sagedusest $\lim_{T \rightarrow \infty} Tc_k = X(\Omega)$. Summa signaali sagedusesituses muutub integraaliks ja Fourier' teisenduse paar pidevate mitteperioodiliste signaalide puhul saab järgneva kju:

$$\begin{cases} x(t) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\Omega) e^{j\Omega t} d\Omega \\ X(\Omega) &= \int_{-\infty}^{\infty} x(t) e^{-j\Omega t} dt \end{cases} \quad (3.7)$$

Kuna signaal ei ole perioodiline, on vaja $X(\Omega)$ avaldises integreerida kogu ajatelje ulatuses. Tegur $\frac{1}{2\pi}$ on vajalik selleks, et energia aja ja sageduse tasandil oleks võrdne.

3.1.3 Diskreetse mitteperioodilise signaali sagedusesitus

Minnes üle diskreetsetele signaalidele, muutub valemi 3.7 teises avaldises integraal summaks. Võttes arvesse, et diskreetseid signaale tähistame $x[n]$ -ga ja diskreetses "maailmas" tähistame sagedust sümboliga ω , võime diskreetsete mitteperioodiliste signaalide puhul Fourier' teisenduse kirjutada järgnevalt:

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n} \quad (3.8)$$

Vastav pöördteisendus erineb Fourier' pöördteisendusest pidevate funktsioonide puhul selle poolest, et kuna diskreetsete signaalide sagedusesitus on perioodiline perioodiga 2π , on mõtet integreerida vaid ühe perioodi ulatuses:

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) e^{j\omega n} d\omega \quad (3.9)$$

Valemid 3.8 ja 3.9 moodustavad Fourier' teisenduse paari diskreetsetele mitteperioodilistele signaalidele. Käesolevas materjalis vaatlemegi üldjuhul just selliseid signaale.

3.1.4 Diskreetse perioodilise signaali sagedusesitus, DFT

Diskreetsete signaalide puhul tähendab perioodilisus seda, et signaali diskreedid korduvad iga N diskreeidi järel (N on perioodi pikkus). Matemaatiliselt väljendatult: $\tilde{x}[n] = \tilde{x}[n + kN]$, $k = 0, \pm 1, \pm 2, \dots$. Sarnaselt pideva perioodilise signaali sagedusesitusega sisaldab ka diskreetse perioodilise signaali sagedusesitus vaid põhisageduse kordseid. Põhisageduseks on nüüd $\frac{2\pi}{N}$. Sagedusesitust võime tähistada $\tilde{X}[k]$ -ga, kuna ta on nii perioodiline kui ka diskreetne. Kuna ka vastav komplekseksponent $e^{j\frac{2\pi}{N}kn}$ on perioodiline perioodiga N , piisab, kui summeerida vahemikus $0 \dots N - 1$, ja Fourier' pöördteisendus diskreetsetele perioodilistele signaalidele saab kuju:

$$\tilde{x}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}[k] e^{j\frac{2\pi}{N}kn} \quad (3.10)$$

Vastav Fourier' teisendus on järgmine:

$$\tilde{X}[k] = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}kn} \quad (3.11)$$

Tihti kasutatakse tähistust $W_N = e^{-j\frac{2\pi}{N}}$. W_N abil võime diskreetsete perioodiliste signaalide Fourier teisenduse paari kirjutada kujul:

$$\begin{cases} \tilde{x}[n] &= \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}[k] W_N^{-kn} \\ \tilde{X}[k] &= \sum_{n=0}^{N-1} \tilde{x}[n] W_N^{kn} \end{cases} \quad (3.12)$$

Nagu eespool olevatest valemitest näha, on diskreetsete perioodiliste signaalide Fourier' teisendus ainus Fourier' teisenduse kuju, mille puhul nii aja tasandi kui ka sageduse tasandi funktsioonid on diskreetsed. Digitaalsetes süsteemides on võimalik opereerida ainult diskreetsete signaalidega. Praktiliselt on kõik mõõtmise tulemusel saadud signaalid aga mitteperioodilised ja seega on nende sagedusesitus pidev. Järgnevalt uurime, milline seos eksisteerib diskreetsete perioodiliste signaalide ja diskreetsete mitteperioodiliste signaalide Fourier' teisenduste vahel.

Olgu meil kaks diskreetset signaali, $x[n]$ ja $\tilde{x}[n]$, nii et

$$x[n] = \begin{cases} \tilde{x}[n], & 0 \leq n \leq N - 1 \\ 0, & \text{mujal} \end{cases} \quad (3.13)$$

Teiste sõnadega, mitteperioodiline (lõplik) signaal $x[n]$ langeb kokku perioodilise signaaliga $\tilde{x}[n]$ põhiperioodi ulatuses ja on mujal 0. Kirjutades signaali $\tilde{x}[n]$ Fourier' jada valemi 3.11 järgi ja võrreldes seda signaali $x[n]$ Fourier' teisendusega (valem 3.8), saame

$$\begin{aligned}\tilde{X}[k] &= \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}kn} \\ &= \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn} \\ &= X(\omega) \Big|_{\omega=\frac{2\pi}{N}k}\end{aligned}\quad (3.14)$$

kus $X(\omega) = \sum_{-\infty}^{\infty} x[n] e^{-j\omega n}$.

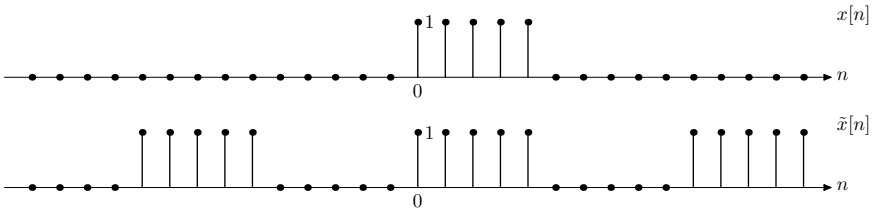
Tulemus osutab, et *diskreetse perioodilise signaali $\tilde{x}[n]$ Fourier' jada on vastava diskreetse mitteperioodilise (lõpliku) signaali $x[n]$ Fourier' teisenduse diskreeditud versioon!* Diskreetimine on siinkohal toimunud sageduse teljel.

Fourier' teisendust, mis põhineb valemil 3.12, kuid jätab arvesse võtmatu perioodilisuse nii aja- kui ka sageduse tasandil, nimetatakse *diskreetseks Fourier' teisenduseks* (DFT). DFT on seega arvutuslikult ekvivalentne diskreetsete perioodiliste signaalide Fourier' jadaga. Diskreetse Fourier' teisenduse valem on sarnane valemiga 3.12, ainus erinevus on perioodilisuse tähise puudumine:

$$\begin{cases} x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-kn} \\ X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn} \end{cases} \quad (3.15)$$

Rakendades digitaalse süsteemi abil Fourier' teisendust signaalilõigule pikkusega N , saame sageduse tasandil N ühtlaselt jaotunud diskreeti sagedustel $\frac{2\pi}{N}k$. Lühikeste signaalide puhul võib saadav sagedusesitus olla ebasoovitavalt harv. On võimalik näidata, et lisades signaalilõigule nullise väärtusega diskreete, tema DFT kuju ei muutu, kuid sagedusesitus tiheneb. On oluline märkida, et nullide lisamine informatsiooni juurde ei too – signaal on täielikult taastatava ka esialgse N diskreedi põhjal. Küll aga aitab tihedam sagedusesitus paremini visuaalselt analüüsida signaali sageduslikku koostumust.

Järgnev näide illustreerib lõpliku signaali $x[n]$ ja vastava perioodilise signaali $\tilde{x}[n]$ sagedusesituste omavahelist seost ning seda, kuidas sagedusesitus muutub tihedamaks signaalilõigule nulle lisades.



Joonis 3.1 Näite 3.1 signaalid $x[n]$ ja $\tilde{x}[n]$

Näide 3.1 Lõpliku signaali ja vastava perioodilise signaali sagedusesituste omavaheline seos

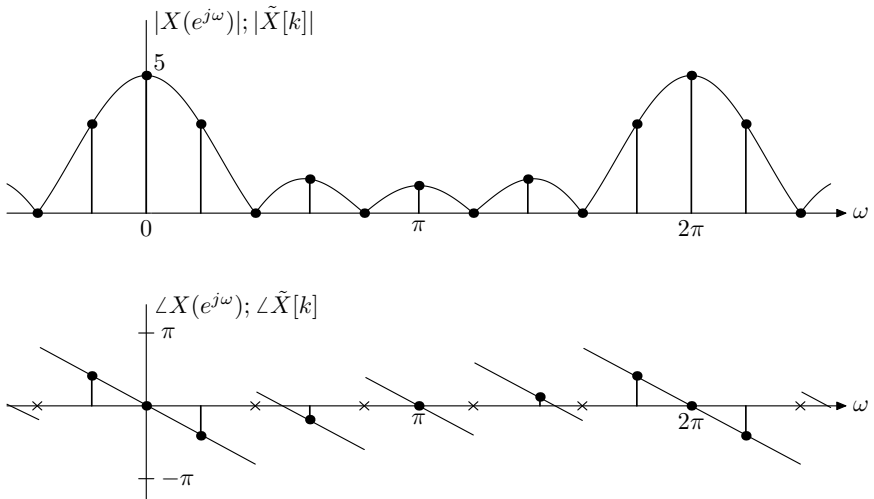
Olgu meil valemi 3.13 järgi moodustatud kaks signaali: $x[n]$ ja $\tilde{x}[n]$ (joonis 3.1). Signaali $x[n]$ Fourier' teisenduse $X(\omega)$ võime avaldada järgmiselt:

$$\begin{aligned}
 X(\omega) &= \sum_{k=-\infty}^{\infty} x[k]e^{-j\omega k} \\
 &= \sum_{k=0}^4 \left(e^{-j\omega} \right)^k \\
 &= \frac{1 - e^{-5j\omega}}{1 - e^{-j\omega}} \\
 &= \frac{e^{-\frac{5}{2}j\omega} \left(e^{\frac{5}{2}j\omega} - e^{-\frac{5}{2}j\omega} \right)}{e^{-\frac{1}{2}j\omega} \left(e^{\frac{1}{2}j\omega} - e^{-\frac{1}{2}j\omega} \right)} \\
 &= e^{-2j\omega} \frac{\sin\left(\frac{5}{2}\omega\right)}{\sin\left(\frac{1}{2}\omega\right)} \tag{3.16}
 \end{aligned}$$

Siinkohal on kasutatud geomeetrilise jada valemit

$$\sum_{k=N_1}^{N_2} a^k = \frac{a^{N_1} - a^{N_2+1}}{1 - a}, \quad N_2 > N_1 \tag{3.17}$$

Funktsiooni $X(\omega)$ magnituud ja faas on esitatud pideva joonega joonisel 3.2. Faasi kujutatakse tihti vahemikus $-\pi \dots \pi$. Nendel ω väärtustel, mil amplituudispekter on võrdne 0-ga, ei ole faas määratud. Teiste sõnadega, kui mingil sagedusel on sinusoidi amplituud 0, siis pole võimalik öelda, milline on tema faas. Sellistel sagedustel võib faasispektris toimuda hüpe.



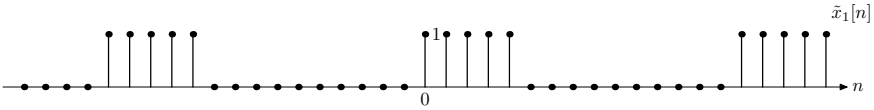
Joonis 3.2 Signaalide $x[n]$ ja $\tilde{x}[n]$ Fourier' teisendused. Ülemisel joonisel on kummagi signaali Fourier' teisenduse magnituud ja alumisel joonisel faas

Signaali $\tilde{x}[n]$ Fourier' jada $\tilde{X}[k]$ võime leida järgnevalt:

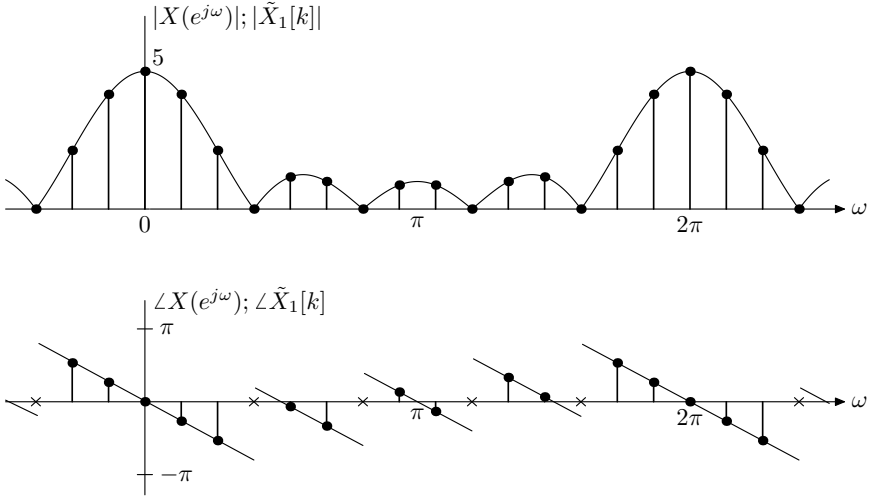
$$\begin{aligned}
 \tilde{X}[k] &= \sum_{k=0}^9 \tilde{x}[n] e^{-j\frac{2\pi}{10}kn} \\
 &= \sum_{k=0}^4 \left(e^{-j\frac{2\pi}{10}k} \right)^n \\
 &= \frac{1 - e^{-j\pi k}}{1 - e^{-j\frac{\pi k}{5}}} \\
 &= e^{-j\frac{2\pi k}{5}} \frac{\sin\left(\frac{\pi k}{2}\right)}{\sin\left(\frac{\pi k}{10}\right)} \quad (3.18)
 \end{aligned}$$

Funktsiooni $\tilde{X}[k]$ magnituud ja faas on esitatud joonisel 3.2 diskreetidena. Nendel k väärtustel, mil $\tilde{X}[k]$ amplituud on võrdne nulliga, pole faasi võimalik määrata ja selle tähistamiseks on kasutatud sümbolit \times . Nagu jooniselt näha, on perioodilise signaali $\tilde{x}[n]$ Fourier' jada vastava lõpliku signaali $x[n]$ pideva Fourier' teisenduse diskreeditud versioon, kusjuures diskreetimine toimub sageduse teljel.

Vaatleme veel, kuidas sagedusesitus muutub, kui perioodilise signaali perioodi jätkata nullidega. Vastava lõpliku signaali puhul tähendab see DFT arvutamisel



Joonis 3.3 Lõplikule signaalile $x[n]$ vastav perioodiline signaal perioodiga $N = 15$



Joonis 3.4 Signaalide $x[n]$ ja $\tilde{x}_1[n]$ Fourier' teisendused

signaalilõigule nullide lisamist. Joonisel 3.3 on kujutatud signaal $\tilde{x}_1[n]$, mille perioodi on jätkatud viie nulliga, võrreldes signaaliga $\tilde{x}[n]$. Sellise signaali Fourier' jada võime avaldada järgmiselt (vt joonis 3.4):

$$\tilde{X}_1[k] = \dots = e^{-j\frac{4\pi k}{15}} \frac{\sin\left(\frac{\pi k}{3}\right)}{\sin\left(\frac{\pi k}{15}\right)} \quad (3.19)$$

Võrreldes omavahel jooniseid 3.2 ja 3.4, võime täheldada, et kui perioodi $N = 10$ puhul signaali $\tilde{x}[n]$ Fourier' jada väärtused jagasid signaali $x[n]$ pideva Fourier' teisenduse vahemikus $0 \dots 2\pi$ kümneks võrdseks osaks vahemike $\frac{2\pi}{10}$ järgi, siis kasvates signaali perioodi viie nulli võrra, saame Fourier' jada, mis järgib sama funktsiooni $X(\omega)$, kuid jagab vahemiku $0 \dots 2\pi$ vahemike $\frac{2\pi}{15}$ järgi 15 võrdseks osaks. Perioodi lõputult pikenedes saame pideva funktsiooni $X(\omega)$.

3.2 Sagedusesitus süsteemide analüüsil

Komplekseksponentfunktsiooni $e^{j\omega}$ tõttu on Fourier' teisenduse valemis sagedusesitus üldjuhul kompleksne – seega võime sagedusesituse kirjutada

- reaalosa ja imaginaarosa abil: $X(\omega) = \Re\{X(\omega)\} + \Im\{X(\omega)\}$;
- magnituudi ja faasi abil: $X(\omega) = |X(\omega)|e^{j\varphi(\omega)}$.

Kui rakendame kahele funktsioonile aja tasandil konvolutsiooni, vastab see sageduse tasandil korrutamisele (ja vastupidi!). Seega, *kuna LTI-süsteemi väljund on sisendi ja impulsskaja konvolutsioon, on väljundi Fourier' teisendus sisendi ja impulsskaja Fourier' teisenduste korrutis:*

$$\begin{aligned} x[n] \quad * \quad h[n] &= y[n] \\ \downarrow F \quad \quad \downarrow F \quad \quad \downarrow F & \\ X(\omega) \quad \cdot \quad H(\omega) &= Y(\omega) \end{aligned} \tag{3.20}$$

Signaali puhul

- nimetame Fourier' teisendust $X(\omega)$ sagedusspektriks;
- nimetame Fourier' teisenduse magnituudi $|X(\omega)|$ amplituudispektriks;
- nimetame Fourier' teisenduse faasi $\angle\{X(\omega)\} = \varphi(\omega)$ faasispektriks.

Süsteemi impulsskaja puhul

- nimetame Fourier' teisendust $H(\omega)$ süsteemi sageduskarakteristikuks;
- nimetame Fourier' teisenduse magnituudi $|H(\omega)|$ süsteemi amplituudikarakteristikuks ja see näitab, kui palju iga sageduskomponent süsteemis võimendub;
- nimetame Fourier' teisenduse faasi $\angle\{H(\omega)\}$ süsteemi faasikarakteristikuks, mis näitab, kui palju iga sageduskomponent nihkub süsteemis faasinurga osas;
- Fourier' teisenduse faasikarakteristiku negatiivset tuletist sageduse järgi nimetame rühmahilistuseks, mis näitab, kui palju iga sageduskomponent nihkub süsteemis absoluutselt ajas:

$$\text{Grd}(\omega) = -\frac{d\angle\{H(\omega)\}}{d\omega} \tag{3.21}$$

Kui süsteemi faasikarakteristik on lineaarne, siis rühmahilistus on konstantne, kõik sageduskomponendid nihkuvad ajas võrdselt ja signaali ku-

ju ei muutu (v.a võimalikust sageduskomponentide sumbumisest tingitud muutus).

3.3 Pideva signaali diskreetimine ja selle mõju sagedusesitusele

Joonisel 3.5 on esitatud muutused signaali sagedusesituses, kui pidev signaal diskreeditakse. Diskreetimist võime ette kujutada impulssjadaga $s(t)$ läbikorrutamisena:

$$s(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT) \quad (3.22)$$

Siinkohal on kasutatud analoogset δ -funktsiooni $\delta(t)$, mis kujutab endast lõpmatult lühikese kestvusega ja lõpmatult suure amplituudiga impulssi ja mille kohta kehtib $\int_{-\infty}^{\infty} \delta(t) dt = 1$. T on impulssidevaheline kaugus ajateljel ehk antud juhul diskreetimisperiood. On võimalik näidata, et funktsiooni $s(t)$ Fourier' teisendus on samuti impulssjada:

$$S(\Omega) = \frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta(\Omega - k\Omega_s), \quad (3.23)$$

kus Ω_s on seotud impulssidevahelise kaugusega T funktsioonis $s(t)$ järgnevalt:

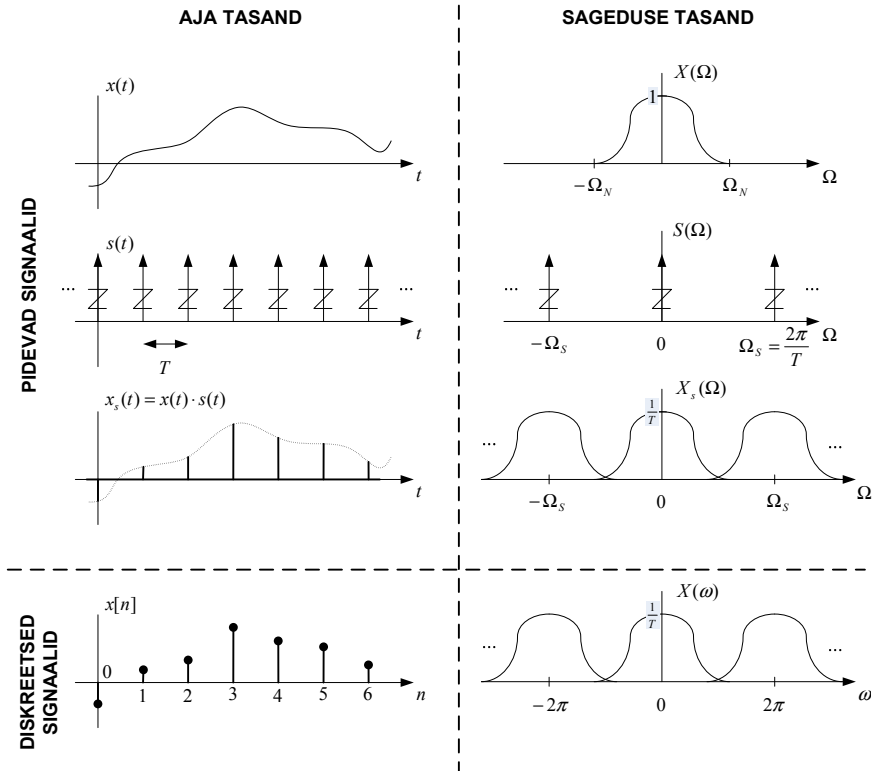
$$\Omega_s = \frac{2\pi}{T} \quad (3.24)$$

Korrutades pideva signaali $x(t)$ impulssjadaga $s(t)$, saame signaali $x_s(t)$, mille väärtused kohtades $t = kT, k = 0, \pm 1, \pm 2, \dots$ on võrdsed signaali $x(t)$ väärtustega vastavates kohtades ja mis mujal on võrdne nulliga (vt joonis 3.5). On oluline märkida, et signaal $x_s(t)$ on endiselt pidev.

Eelnevas lõigus tõdesime, et konvolutsioonile aja tasandil vastab korrutamine sageduse tasandil ja vastupidi. Seega saame signaali $x_s(t)$ sagedusesituse, kui võtame funktsioonide $X(\Omega)$ ja $S(\Omega)$ vahelise konvolutsiooni

$$X_s(\Omega) = \frac{1}{2\pi} X(\Omega) * S(\Omega) \quad (3.25)$$

$$= \frac{1}{T} \sum_{k=-\infty}^{\infty} X(\Omega - k\Omega_s) \quad (3.26)$$



Joonis 3.5 Pideva signaali diskreetimine vaadelduna impulssjadaga läbikorrutamisenä. Diskreeditud signaali spekter on vastava pideva signaali spektri nihutatud koopiate summa

Joonisel 3.5 on näha, et $X_s(\Omega)$ koosneb $X(\Omega)$ koopiatest, mis on skaalatud teguriga $\frac{1}{T}$ ja nihutatud üksteise suhtes $k\Omega_s$ võrra. Siit nähtub ka see, miks diskreetse signaali sagedusesitus on alati perioodiline.

Diskreetsesse "maailma" üleminek kujutab endast signaali $x_s(t)$ nullist erinevate väärtuste "väljanõppimist" kusjuures ajatelg asendub diskreedi järjekorranumbriga. Sealjuures kaob informatsioon selle kohta, kui kaugel teineteisest diskreedid ajas paiknevad. Sageduse tasandil jääb diskreetse signaali sagedusspekter üldjuhul pidevaks, siin kujutab üleminek diskreetsesse "maailma" endast sageduse telje skaala muutumist:

$$\omega = \Omega T \quad (3.27)$$

Kui oletada, et signaali $x(t)$ sagedusriba on piiratud ja seal esinev kõrgeim sagedus on Ω_N , võime diskreetimisperioodi T valida

- nii, et $\Omega_s = \frac{2\pi}{T} > 2\Omega_N$, mispuhul $X(\Omega)$ koopiad funktsioonis $X_s(\Omega)$ ei kattu. Sel juhul on võimalik algne signaal $x(t)$ täielikult taastada diskreetse signaali $x[n]$ diskreetide põhjal ja diskreetimisel informatsiooni ei kao;
- nii, et $\Omega_s = \frac{2\pi}{T} < 2\Omega_N$, mispuhul $X(\Omega)$ koopiad funktsioonis $X_s(\Omega)$ kattuvad ja diskreetimisel kaotatakse informatsiooni.

Sagedust $2\Omega_N$ kutsutakse *Nyquisti sageduseks*.

3.4 Diskreetse Fourier' teisenduse arvutamine

Fourier' teisenduse olulisus diskreetsete signaalide analüüsil tuleneb suures osas sellest, et DFT arvutamiseks on olemas kiireid algoritme. Järgnevalt vaatleme ühte neist, nn *Decimation-In-Time*-algoritmi.

Kasutades diskreetse Fourier' teisenduse valemit originaalkujul (valem 3.15), on pikkusega N signaalilõigu DFT arvutamiseks vaja sooritada N^2 kompleksset korrutustehet ja $N(N - 1)$ kompleksset liitmistehet. Arvutuslikku mahtu on võimalik oluliselt vähendada, kasutades funktsiooni W_N^{kn} sümmeetria- ja ortogonaalsusomadusi.

Jagame kõigepealt pikkusega N signaalilõigu DFT arvutamise kaheks $\frac{N}{2}$ -punktiseks DFT-ks, võttes ühte summasse signaali paaris diskreedid ja teise

paaritud diskreedid:

$$\begin{aligned}
 X[k] &= \sum_{n=0}^{N-1} x[n] W_N^{kn} \\
 &= \sum_{r=0}^{\frac{N}{2}-1} x[2r] W_N^{2rk} + \sum_{r=0}^{\frac{N}{2}-1} x[2r+1] W_N^{(2r+1)k} \\
 &= \sum_{r=0}^{\frac{N}{2}-1} x[2r] W_N^{2rk} + W_N^k \sum_{r=0}^{\frac{N}{2}-1} x[2r+1] W_N^{2rk} \quad (3.28)
 \end{aligned}$$

Kuna $W_N^2 = W_{\frac{N}{2}}$, võime kirjutada:

$$X[k] = \sum_{r=0}^{\frac{N}{2}-1} x[2r] W_{\frac{N}{2}}^{rk} + W_N^k \sum_{r=0}^{\frac{N}{2}-1} x[2r+1] W_{\frac{N}{2}}^{rk} \quad (3.29)$$

Valemi 3.29 mõlemad summad kujutavad endast $\frac{N}{2}$ punkti diskreetset Fourier' teisendust. Seega on selle avaldise realiseerimiseks vaja sooritada $N + \frac{N^2}{2}$ korrutustehet.

Kui N on kahe aste, võime jätkata algoritmi poolitamist kuni 2-punktise DFT-ni. Korrutustehete määraks moodustub sel juhul $N \log_2 N$. Näiteks kui $N = 1024$, on DFT võtmiseks otse valemi 3.15 järgi vaja 1048576 korrutustehet, samal ajal kui ülal esitatud algoritm (FFT-algoritm, ingl.k *Fast Fourier Transform*) annab korrutustehete määraks 10240.

FFT-algoritmi on põhjalikumalt käsitletud käesoleva materjali harjutuste osas. Selle algoritmi positiivsete omadustena võib veel mainida järgmist:

- Kuigi arvutus toimub mitme sammu kaupa, on nii sisend-, väljund- kui ka vahemuutujate salvestamiseks vaja vaid üht mäluvektorit pikkusega N . Seda seetõttu, et igal sammul arvutatakse kahe sisendväärtuse põhjal täpselt kaks väljundväärtust.
- Algoritm loeb sisendvektori väärtusi pööratud bittidega aadressi järjekorras (ingl.k *bit-reversed order*). Mitmed signaaliprotsessorid toetavad sellist andmete adresseerimist.
- Algoritmi on võimalik muuta nii, et
 - sisendeid loetakse normaalses järjekorras, kuid väljundid saadakse pööratud bittidega aadressi järjekorras;

- nii sisendid kui ka väljundid on lineaarses järjekorras (sealjuures kaob algoritmi struktuuri regulaarsus);
- kõik sammud on realiseeritavad sama struktuuri abil.

Peale *Decimation-In-Time*-algoritmi on veel muid kiireid DFT-algoritme. Näiteks *Decimation-In-Frequency*-algoritmi puhul poolitatakse DFT-blokkide pikkust sageduse tasandi poolelt. Samuti on algoritme, mis ei nõua, et teisendatava signaalilõigu pikkus oleks $N = 2^k$. See ei osutu siiski praktiliselt piiranguks ka *Decimation-In-Time*-algoritmi puhul, sest nagu varem vaatlesime, võime alati signaalilõiku jätkata nullidega lähima 2 astmeni, ilma et spektri kuju muutuks.

4 LTI-süsteemide analüüs z-tasandil

4.1 Z-teisendus ja selle koonduvuspiirkond

Fourier' teisendus diskreetsete mitteperioodiliste signaalide puhul omab järgnevat kuju (valemi 3.8 kordus):

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \quad (4.1)$$

Siin on baasfunktsioonide $e^{j\omega}$ väärtused piiratud kompleksarvudega, mille magnituud on 1. Komplekstanalüüsi (seda kutsutakse diskreetsete signaalide ja süsteemide analüüsi z-tasandiks) paiknevad sellised arvud ühikringil.

Z-teisendus on Fourier' teisenduse üldistus, mille puhul baasfunktsioonide hulka on laiendatud katma kogu komplekstanalüüsi: $e^{j\omega} \rightarrow z = re^{j\omega}$. Z-teisendus näeb seega välja järgmiselt:

$$X(z) = \sum_{n=-\infty}^{\infty} x[n](re^{j\omega})^{-n} = \sum_{n=-\infty}^{\infty} x[n]z^{-n} \quad (4.2)$$

Seega on Fourier' teisendus Z-teisenduse erijuht, kui $r = 1$, ehk teiste sõnadega, Fourier' teisenduse saame, kui vaatleme Z-teisendust piki ühikringi. Diskreetse signaali Fourier' teisenduse perioodilisus avaldub siin selles, et ühikring kordub iga 2π järel.

Z-teisenduse puhul on oluline selle koonduvus ehk see, millistele signaalidele ja milliste r väärtuste puhul valem 4.2 annab lõpliku tulemuse.

Näide 4.1 *Funktsiooni $x[n] = a^n u[n]$ Z-teisenduse koondumine*

Vaatleme funktsiooni $x[n] = a^n u[n]$ Z-teisenduse koonduvust. Selle funktsiooni

Z-teisenduse võime leida järgnevalt:

$$\begin{aligned}
 X(z) &= \sum_{n=-\infty}^{\infty} a^n u[n] z^{-n} \\
 &= \sum_{n=0}^{\infty} (az^{-1})^n \\
 &= \frac{1 - \left(\frac{a}{z}\right)^{\infty}}{1 - az^{-1}} \tag{4.3}
 \end{aligned}$$

$$\stackrel{|z| > |a|}{=} \frac{1}{1 - az^{-1}} \tag{4.4}$$

Avaldises 4.3 olev summa koondub, kui $|z| > |a|$, seetõttu hõlmab funktsiooni $a^n u[n]$ koonduvuspiirkond (ROC, ingl.k Range of Convergence) kõiki neid kompleksmuutuja z väärtusi, mis on absoluutväärtuselt suuremad kui $|a|$ ehk paiknevad z -tasandil koordinaatide alguspunktit kaugemal kui a .

LTI-süsteemide puhul on olulisel kohal sellised kompleksmuutuja z funktsioonid, mis on avaldatavad z polünoomide suhtena: $X(z) = \frac{P(z)}{Q(z)}$. Sealjuures kutsutakse neid kompleksmuutuja z väärtusi, mille puhul $Q(z)$ on võrdne nulliga, $X(z)$ -i poolusteks. Kompleksmuutuja z väärtusi, mille puhul $P(z)$ on võrdne nulliga, kutsutakse $X(z)$ -i nullideks.

Näite 4.1 puhul leidsime, et funktsiooni $a^n u[n]$ Z-teisendus on $\frac{1}{1-az^{-1}}$. Selle funktsiooni nimetaja on 0 (ja seega $X(z)$ -il on poolus) kohas $z = a$. Näites leidsime, et antud juhul on $X(z)$ -i koonduvuspiirkonnaks piirkond $|z| > |a|$. Funktsioon $a^n u[n]$ on parempoolne funktsioon, kuna tal on kindel alguspunkt kohas $n = 0$, kuid lõpp-punkt puudub (s.t lõpp-punkt on lõpmatuses). On võimalik näidata, et *parempoolsete funktsioonide Z-teisenduse koonduvuspiirkond ulatub alati suurima magnituudiga poolusest lõpmatusse*.

Sarnaselt näitega 4.1 on võimalik näidata, et funktsiooni $-a^n u[-n - 1]$ Z-teisendus on samuti $\frac{1}{1-az^{-1}}$, kuid koonduvuspiirkonnaga $|z| < |a|$. Seega ei ole funktsioon aja tasandil üheselt määratud pelgalt tema Z-teisendusega, vaid on vaja teada ka koonduvuspiirkonda. Funktsioon $-a^n u[-n - 1]$ algab lõpmatuses ja lõpeb kohas $n = -1$, olles seega vasakpoolne funktsioon. *Vasakpoolsete funktsioonide Z-teisenduse koonduvuspiirkonna moodustavad alati z-tasandi punktid, mis paiknevad koordinaatide alguspunktile lähemal kui kõige väiksema magnituudiga poolus*. Sealjuures võib punkt $z = 0$ jääda koonduvuspiirkonnast välja.

Kui funktsioon ulatub miinus lõpmatuses pluss lõpmatusse, on tema Z-teisenduse koonduvuspiirkond rõngakujuline, piiratud mõlemalt poolt

(absoluutväärtuse osas) poolusega. Kui aga funktsioon on lõplik, on tema Z -teisenduse koonduvuspiirkonnaks kogu z -tasand, välja arvatud ehk punktid $z = 0$ ja/või $z = \infty$. Koonduvuspiirkond piirneb alati ringiga, kuna koonduvus sõltub kompleksmuutuja magnituudist, mitte faasist. Koonduvuspiirkond ei sisalda pooluseid ja ta on alati ühtne, s.t koonduvuspiirkonna sisse ei või jääda punkte või piirkondi, kus antud funktsiooni Z -teisendus ei koonu.

LTI-süsteemide puhul on süsteemi väljundsignaali Z -teisendus sisendsignaali Z -teisenduse ja süsteemi impulsskaja Z -teisenduse korrutis:

$$Y(z) = X(z) \cdot H(z)$$

Funktsiooni $H(z)$ nimetatakse süsteemi süsteemifunktsiooniks ehk ülekandefunktsiooniks.

4.2 Diferentsvõrrandi abil esitatavad süsteemid z -tasandil

Selles lõigus vaatleme, kuidas avaldub süsteemifunktsioon selliste süsteemide puhul, mis on aja tasandil esitatavad diferentsvõrrandi abil (valemi 2.11 kordus):

$$\sum_{k=0}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k] \quad (4.5)$$

Selliste süsteemide puhul võime süsteemifunktsiooni $H(z)$ leida otse diferentsvõrrandist, ilma et peaksime avaldama impulsskaja $h[n]$.

Võtame diferentsvõrrandi mõlemast poolest Z -teisenduse, kasutades järgmisi tähistusi ja Z -teisenduse omadusi:

- $x[n] \xrightarrow{Z} X(z)$
- $y[n] \xrightarrow{Z} Y(z)$
- $ax_1[n] + bx_2[n] \xrightarrow{Z} aX_1(z) + bX_2(z)$; see omadus väljendab Z -teisenduse linearsust
- $x[n - n_0] \xrightarrow{Z} z^{-n_0} X(z)$

Saame

$$\sum_{k=0}^N a_k Y(z) z^{-k} = \sum_{k=0}^M b_k X(z) z^{-k} \quad (4.6)$$

Liikmed $Y(z)$ ja $X(z)$ võime tuua summa ette ja seejärel saame $X(z)$ -ga läbi jagades

$$\begin{aligned} Y(z) \sum_{k=0}^N a_k z^{-k} &= X(z) \sum_{k=0}^M b_k z^{-k} \\ H(z) = \frac{Y(z)}{X(z)} &= \frac{\sum_{k=0}^M b_k z^{-k}}{\sum_{k=0}^N a_k z^{-k}} \end{aligned} \quad (4.7)$$

Tulemus näitab, et aja tasandil diferentsvõrrandi abil esitatava süsteemi süsteemifunktsioon on alati ratsionaalne funktsioon (s.t esitatav kompleksmuutuja z polünoomide suhte kujul). Valemis 4.7 esinevad süsteemifunktsiooni $H(z)$ lugejas ja nimetajas kompleksmuutuja z negatiivsete astmete polünoomid. Kompleksmuutuja z positiivsetele astmetele võime üle minna, korrutades murru lugejat ja nimetajat avaldisega $z^{\max\{M,N\}}$.

Süsteemifunktsiooni $H(z)$ lugeja ja nimetaja polünoomid võime esitada nende nullkohtade abil vastavate tegurite korrutisena. Sel juhul omandab süsteemifunktsioon järgmise kuju:

$$H(z) = \frac{\prod_{k=1}^M (1 - c_k z^{-1})}{\prod_{k=1}^N (1 - d_k z^{-1})} \quad (4.8)$$

Siinkohal on väärtused c_k ja d_k vastavalt süsteemifunktsiooni nullid ja poolused.

4.3 LTI-süsteemide omadused z -tasandil

Järgnevalt vaatleme, kuidas LTI-süsteemide omadused, sealhulgas stabiilsus ja kausaalsus, avalduvad kompleksmuutuja tasandil.

Lõigus 2.4 nägime, et *kausaalne* LTI-süsteemi puhul $h[n] = 0$, $n < 0$. See aga tähendab, et süsteemi impulsskaja $h[n]$ on parempoolne funktsioon ja

seega ulatub tema Z -teisenduse (ehk $H(z)$ -i) koonduvuspiirkond suurima magnituudiga poolusest lõpmatusse.

Stabiilse LTI-süsteemi puhul leidsime, et $\sum_{n=-\infty}^{\infty} |h[n]| < \infty$ (samuti lõigus 2.4). Uurides funktsiooni $h[n]$ Fourier' teisenduse koonduvust, on võimalik osutada, et

$$\begin{aligned} |H(\omega)| &= \left| \sum_{n=-\infty}^{\infty} h[n] e^{-j\omega n} \right| \\ &\leq \sum_{n=-\infty}^{\infty} |h[n]| |e^{-j\omega n}| \\ &\leq \sum_{n=-\infty}^{\infty} |h[n]| \leq \infty \end{aligned} \quad (4.9)$$

See aga tähendab, et kui süsteemi impulsskaja on absoluutselt summeeruv, siis tema Fourier' teisendus (ehk sageduskarakteristik $H(\omega)$) koondub. Fourier' teisendus on Z -teisenduse erijuht, kui $|z| = 1$ (väärtused ühikringil). Võime seega järeldada, et stabiilse LTI-süsteemi puhul sisaldab süsteemifunktsiooni $H(z)$ koonduvuspiirkond ühikringi.

Üldjuhul soovime disainida süsteeme, millel oleks nii kausaalsuse kui ka stabiilsuse omadus. Eelnevast võimegi teha olulise järelduse: *stabiilse ja kausaalse LTI-süsteemi kõik poolused peavad paiknema ühikringi sees!*

Lõigus 2.4 oli juttu **pöördsüsteemist** – kui süsteemid $h[n]$ ja $h_i[n]$ on teineteise pöördsüsteemid, siis $h[n] * h_i[n] = \delta[n]$. Kompleksmuutuja tasandil näeb vastav tingimus välja $H(z) \cdot H_i(z) = 1$ ja järelikult $H_i(z) = \frac{1}{H(z)}$. Valemist 4.8 on näha, et kui väärtused c_k on süsteemi $H(z)$ nullideks, siis pöördsüsteemi $H_i(z)$ puhul peavad samad väärtused olema poolusteks. Samuti peavad $H(z)$ -i poolused d_k olema $H_i(z)$ -i nullideks.

Diferentsvõrrandi abil esitatava süsteemi süsteemifunktsiooni $H(z)$ on võimalik lahutada osamurdudeks.¹ Juhul kui $H(z)$ sisaldab vaid ühekordseid pooluseid, väljendub ta osamurdude kaudu järgnevalt:

$$H(z) = \sum_{r=0}^{M-N} B_r z^{-r} + \sum_{k=1}^N \frac{A_k}{1 - d_k z^{-1}} \quad (4.10)$$

Siinkohal on B_r ja A_k funktsiooni $H(z)$ parameetrid. Avaldisest 4.10 on võimalik võtta Z -pöördteisendus ja avaldada $h[n]$:

$$h[n] = \sum_{r=0}^{M-N} B_r \delta[n-r] + \sum_{k=1}^N A_k d_k^n u[n] \quad (4.11)$$

¹Osamurdudeks lahutamisest on lähemalt juttu materjali harjutuste osas.

Siit on näha, et taoliselt avaldatuna impulsskaja koosneb kahest summast – esimene summa on lõplik ja teine on lõpmatu. Teine summa tuleb arvesse vaid juhul, kui $H(z)$ -il esineb pooluseid kohtades $0 < |z| < \infty$. Kui pooluseid ei ole või kui neid on ainult kohas $z = 0$ või $z = \infty$, siis impulsskaja, süsteemifunktsioon ja diferentsvõrrand saavad vastavalt järgmised kujud:

$$h[n] = \sum_{k=0}^M b_k \delta[n-k]; \quad H(z) = \sum_{k=0}^M b_k z^{-k}; \quad y[n] = \sum_{k=0}^M b_k x[n-k] \quad (4.12)$$

Võime järeldada, et kui süsteemi süsteemifunktsioon $H(z)$ sisaldab pooluseid kohtades $0 < |z| < \infty$, siis tema impulsskaja on lõpmatu (süsteem on IIR). Lõpliku impulsskajaga süsteemi (FIR-süsteemi) $H(z)$ on alati võimalik esitada ilma poolusteta (v.a võimalikud poolused kohas $z = 0$ ja $z = \infty$).

Mitmekordsete pooluste puhul annab osamurdudeks lahutamine erineva süsteemifunktsiooni kuju, kuid lõppjärgeldus on sama.

4.4 Nullide ja pooluste paiknemise seos sageduskarakteristikuga

Lineaarsete süsteemide disainimine toimub sageduse tasandil etteantavate kriteeriumite alusel. Z -teisendus on siinkohal süsteemide analüüsi ja disaini abivahendiks, mille kaudu on võimalik leida süsteemi kordajad, kui sageduskarakteristiku kriteeriumid on ette antud. Seega on süsteemide analüüsi ja disainimise seisukohalt oluline, kuidas z -tasandi nullid ja poolused on seotud sageduskarakteristikuga (ehk süsteemifunktsiooni $H(z)$ käitumisega ühikringil).

Osutub, et iga $H(z)$ -i null/poolus (välja arvatud nullid ja poolused kohtades $z = 0$ ja $z = \infty$) põhjustab amplituudikarakteristikus miinimumi/maksimumi sagedusel, mis vastab antud nulli/pooluse faasinurgale. Miinimum/maksimum on seda teravam, mida lähemal on vastav null/poolus ühikringile. Ühikringi suhtes vastandlikult paiknevad (sama faasiga, kuid kaugus koordinaatide alguspunktist vastavalt r ja $\frac{1}{r}$) nullid/poolused põhjustavad amplituudikarakteristikus ühesuurused miinimumid/maksimumid.

Ülaltoodud seaduspärasusi on illustreeritud joonisel 4.1, kus on ära toodud nelja süsteemi nullide paiknemine (vasakul) koos vastavate amplituudi- ja faasikarakteristikutega (paremal). Amplituudikarakteristikud on esitatud logaritmilises skaalas. Sagedustelg on normaliseeritud nii, et vahemik $0 \dots 1$ vastab sagedusvahemikule $0 \dots \pi$ rad. Ühikringil paiknevad

nullid põhjustavad lõpmatu sumbuvuse vastaval sagedusel, kusjuures faasikarakteristiku väärtus on selles punktis määramata. Nullid, mille kaugus koordinaatide alguspunktist on 0.9 ja $\frac{1}{0.9}$ (joonisel süsteemid b ja d), põhjustavad amplituudikarakteristikus identseid muutusi, faasikarakteristik on taas kummalgi juhul erinev.

Joonisel 4.1 esitatud süsteemide nullid paiknevad kaaskomplekspaaridena. See valik ei ole juhuslik – osutub nimelt, et vaid süsteemid, mille kõik nullid ja poolused moodustavad kaaskomplekspaare, omavad reaalseid diferentsvõrrandi kordajaid. Erandiks on z -tasandi reaalteljel paiknevad nullid ja poolused.

4.5 All-pass-süsteemid

All-pass-süsteemideks nimetatakse süsteeme, mille amplituudikarakteristik on konstantne. Selline süsteem laseb kõiki sagedusi võrdselt läbi. LTI-süsteemil on all-pass-omadus, kui ta süsteemifunktsioon moodustub järgmisel kujul väljendatavatest teguritest:

$$H_{ap}(z) = \frac{z^{-1} - a^*}{1 - az^{-1}} \quad (4.13)$$

Vaatleme tegurile $H_{ap}(z)$ vastavat sageduskarakteristikut $H_{ap}(\omega)$. Sageduskarakteristiku saame, kui vaatleme süsteemifunktsiooni käitumist ühikringil ehk asendame $H_{ap}(z)$ avaldises $z \rightarrow e^{j\omega}$:

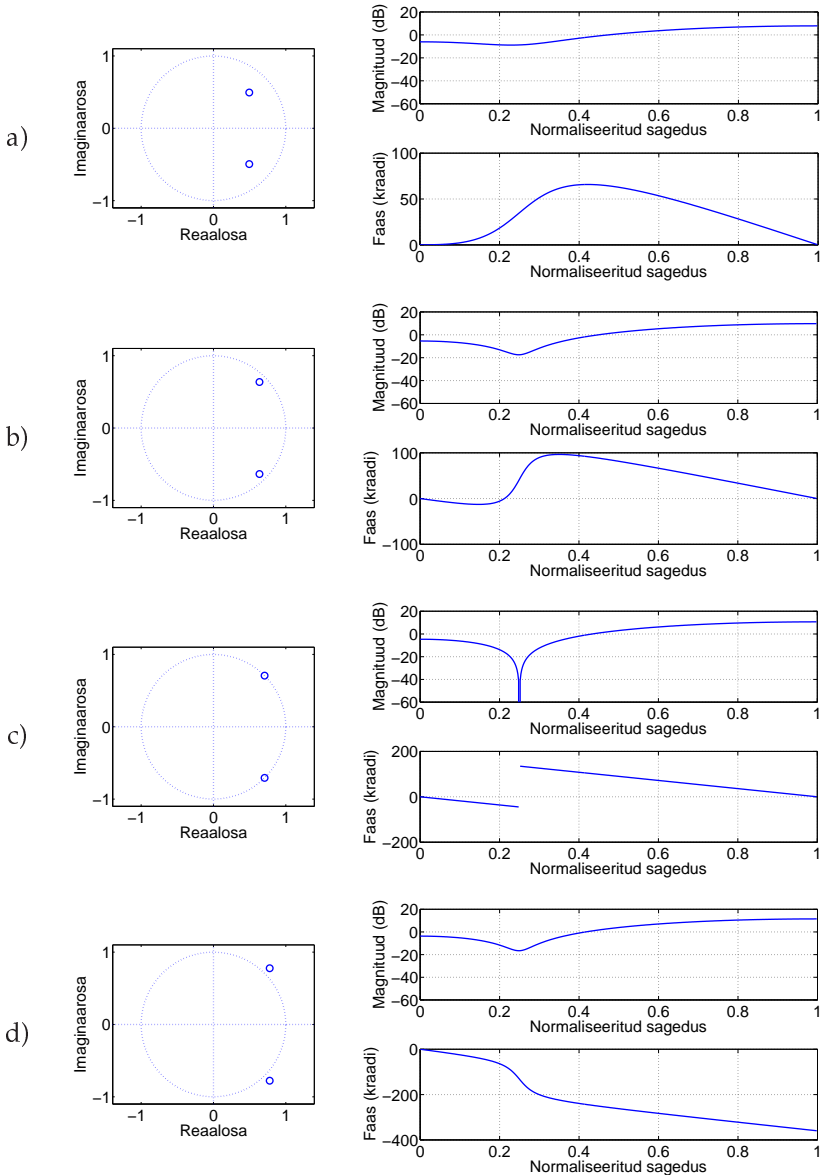
$$\begin{aligned} H_{ap}(\omega) &= \frac{e^{-j\omega} - a^*}{1 - ae^{-j\omega}} \\ &= e^{-j\omega} \frac{1 - a^* e^{j\omega}}{1 - ae^{-j\omega}} \end{aligned} \quad (4.14)$$

Teguri $e^{-j\omega}$ magnituud $|e^{-j\omega}|$ on võrdne ühega. Valemis 4.14 esineva murru lugeja ja nimetaja on üksteise kaaskompleksarvud ja seega on nende magnituudid võrdsed. Järelikult võime kirjutada

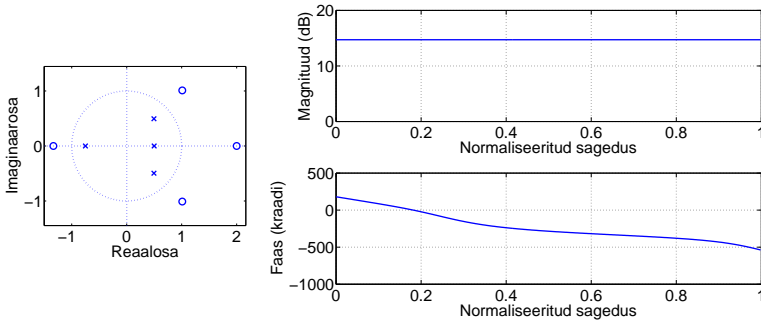
$$\left| e^{-j\omega} \frac{1 - a^* e^{j\omega}}{1 - ae^{-j\omega}} \right| = 1 \quad (4.15)$$

ja oleme näidanud, et $H_{ap}(z)$ on tõepoolest all-pass-süsteem.

Järgnevalt uurime, kuidas paiknevad teguri $H_{ap}(z)$ null ja poolus. Nimetatud teguril on poolus kohas a , mille võime kirjutada lahti magnituudi ja



Joonis 4.1 Sageduskarakteristiku seos nullide paiknemisega ühikringi suhtes. Nullide täpsed asukohad on järgmised: a) $0.7e^{\pm j\pi/4}$; b) $0.9e^{\pm j\pi/4}$; c) $e^{\pm j\pi/4}$ ning d) $\frac{1}{0.9}e^{\pm j\pi/4}$. Kõik esitatud amplituudikarakteristikud on skaalatud vahemikku $-60 \dots 20$ dB; eri süsteemide faasikarakteristikud on aga eri skaalas



Joonis 4.2 Näide *all-pass*-süsteemi nullide ja pooluste paiknemisest

faasi kaudu järgmiselt: $a = re^{j\varphi}$. $H_{ap}(z)$ null paikneb seega kohas

$$\begin{aligned}
 z^{-1} - a^* &= 0 \\
 z^{-1} - re^{j\varphi} &= 0 \\
 z &= \frac{1}{r}e^{-j\varphi}
 \end{aligned} \tag{4.16}$$

Siit on näha, et $H_{ap}(z)$ nulli ja pooluse faasinurk on sama. Kui pooluse magnituud on r , siis nulli magnituud on $\frac{1}{r}$.

Joonisel 4.2 on toodud näide *all-pass*-süsteemi nullide ja pooluste paiknemisest. Kui lisaks *all-pass*-omadusele soovime ka, et süsteemi kordajad oleksid reaalsed, peab nii poolusel kui ka nullil olema vastav kaaskompleksne poolus ja null. Seega paiknevad *all-pass*-süsteemi nullid ja poolused üldjuhul neljastes rühmades. Joonisel moodustavad taolise rühma nullid kohtades $\frac{1}{0.7}e^{j\frac{\pi}{4}}$ ja $\frac{1}{0.7}e^{-j\frac{\pi}{4}}$ ning poolused kohtades $0.7e^{j\frac{\pi}{4}}$ ja $0.7e^{-j\frac{\pi}{4}}$. Reaalteljel paiknevad nullid ja poolused võivad esineda paarikaupa. Jooniselt on näha ka see, et esitatud süsteemil on tõepoolest *all-pass*-omadus, s.t et selle süsteemi amplituudikarakteristik on konstantne.

Oluline *all-pass*-süsteemide omadus on faasikarakteristiku laskuv iseloom. Täpsemalt öeldes, *all-pass*-süsteemi faasikarakteristiku tuletis sageduse järgi (rühmahilistus) ei ole kunagi negatiivne.

All-pass-süsteemide abil on võimalik kompenseerida faasikarakteristiku käitumist, ilma et amplituudikarakteristik muutuks. Samuti on võimalik sobivat faasikarakteristikut omavate *all-pass*-süsteemide abil moodustada kõrgpääs-madalpääs-filtrite paare.

4.6 Amplituudi- ja faasikarakteristiku omavahe- line sõltuvus

Eespool mainisime, et süsteemifunktsiooni nullid (poolused), mis paiknevad ühikringi suhtes vastandkohtades, mõjutavad amplituudikarakteristikut samamoodi. Diferentsvõrrandi abil esitatavate LTI-süsteemide puhul ongi sama amplituudikarakteristikut (kuid erinevaid faasikarakteristikuid) omavaid süsteeme võimalik realiseerida täpselt nii palju, kui on erinevaid võimalusi paigutada kaaskompleksseid nullide paare vastandkohtadesse kummalegi poole ühikringi. Stabiilsuse ja kausaalsuse säilimiseks peavad poolused paiknema alati ühikringi sees. Joonisel 4.3 on esitatud neli kaaskomplekssete nullide paariga süsteemi, millel on sama amplituudikarakteristik, kuid erinev faasikarakteristik.

4.7 Minimaalse faasiga süsteemid

Eelnevas lõigus käsitletu põhjal on kerge märgata, et kõikide sama amplituudikarakteristikut omavate süsteemide hulgas leidub vaid üks süsteem, mille kõik nullid on ühikringi sees (joonisel 4.3 süsteem a). Sellist süsteemi nimetatakse minimaalse faasiga süsteemiks. Sellise nimetuse põhjus selgub järgnevast.

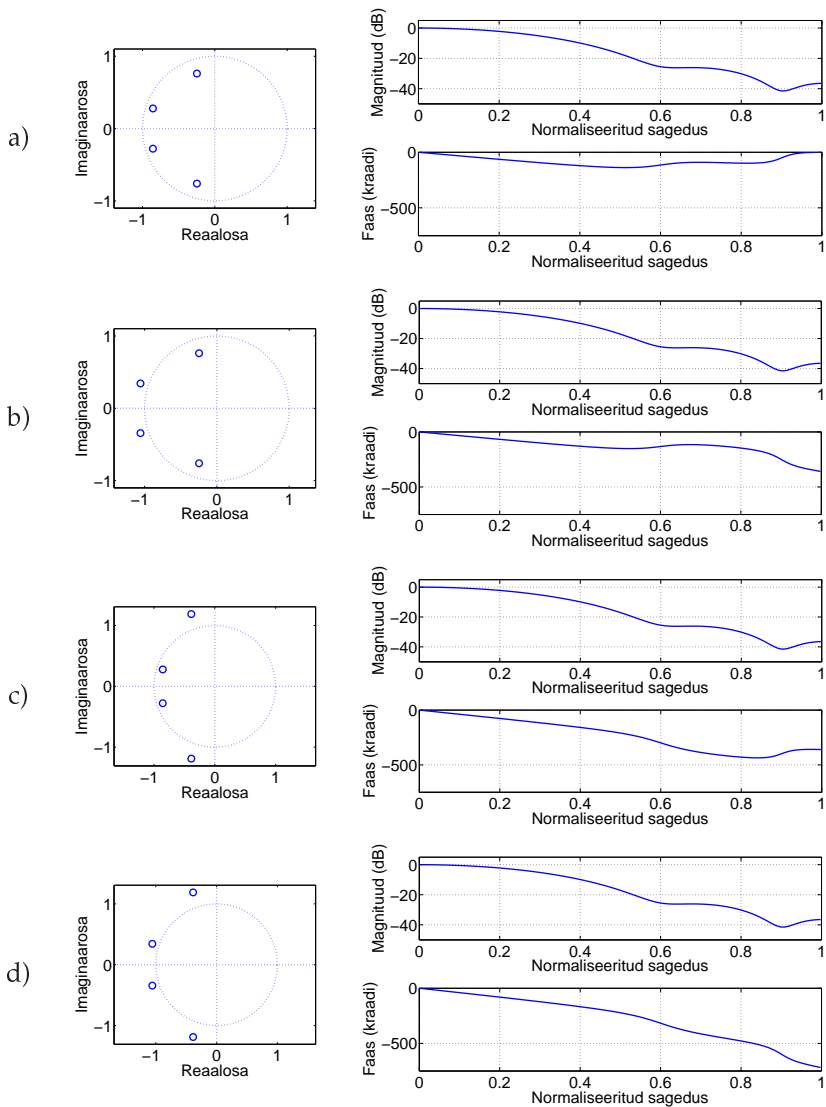
Mis tahes diferentsvõrrandiga väljendatavat LTI-süsteemi on võimalik esitada minimaalse faasiga süsteemi ja *all-pass*-süsteemi jadaühendusena. Vaatleme süsteemi, mille kõik nullid peale ühe paiknevad ühikringi sees (joonis 4.4 a). Sellise süsteemi süsteemifunktsiooni võime väljendada järgmiselt:

$$H(z) = H_1(z)(z^{-1} - c^*) \quad (4.17)$$

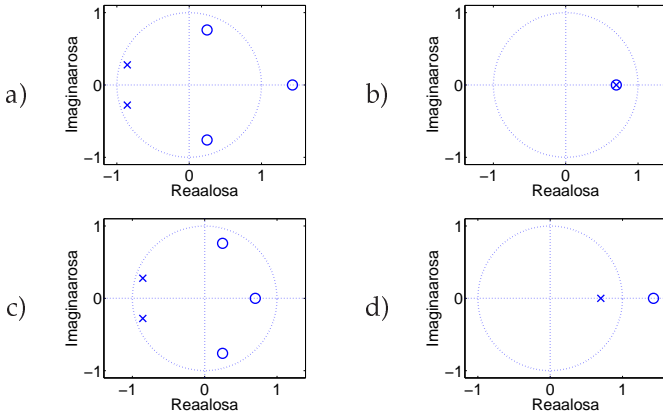
Siinkohal kuuluvad süsteemi $H_1(z)$ kõik süsteemi $H(z)$ poolused ja ühikringi sees paiknevad nullid, seega $H_1(z)$ on definitsiooni põhjal minimaalse faasiga süsteem. Ainus süsteemi $H(z)$ ühikringist väljaspool paiknev null on kohas $z = \frac{1}{c^*}$.

Kui süsteemile lisada null ja poolus, mis paiknevad samas kohas ja seega elimineerivad teineteise mõju, süsteemifunktsioon ei muutu. Lisame süsteemile $H(z)$ nulli ja pooluse kohta $z = c$ ehk $H(z)$ -i väljaspool ühikringi paikneva pooluse suhtes vastandkohta (joonis 4.4 b):

$$H(z) = H_1(z)(z^{-1} - c^*) \frac{1 - cz^{-1}}{1 - cz^{-1}} \quad (4.18)$$



Joonis 4.3 Näide sama amplituudikarakteristik, kuid erinevat faasikarakteristikut omavatest süsteemidest. Süsteem a on minimaalse faasiga ja süsteem d on maksimaalse faasiga



Joonis 4.4 Näide süsteemi $H(z)$ esitamisest minimaalse faasiga süsteemi ja *all-pass*-süsteemi abil. a) Süsteemi $H(z)$ nullid ja poolused, b) süsteemile $H(z)$ lisatud null ja poolus, mis kompenseerivad teineteise, c) minimaalse faasiga tegur $H(z)$ -i esituses, d) *all-pass*-tegur $H(z)$ -i esituses

Järgnevalt rühmitame nullid ja poolused nii, et lisatud null moodustab koos $H_1(z)$ -i nullide ja poolustega minimaalse faasiga süsteemi ja lisatud poolus moodustab koos väljaspool ühikringi oleva nulliga *all-pass*-süsteemi (joonis 4.4 c ja d):

$$\begin{aligned}
 H_{\text{minfaas}}(z) &= H_1(z)(1 - cz^{-1}) \\
 H_{\text{ap}}(z) &= \frac{z^{-1} - c^*}{1 - cz^{-1}}
 \end{aligned}
 \tag{4.19}$$

Juhul kui süsteemil on rohkem kui üks väljaspool ühikringi paiknev null, võime sarnaselt toimida kõigi selliste nullidega ja seega esitada mis tahes süsteemi minimaalse faasiga süsteemi ja *all-pass*-süsteemi jadaühenduse-na.

All-pass-süsteemide juures mainisime, et nende süsteemide rühmahilistus on alati positiivne või võrdne nulliga. Kuna jadamisi ühendatud süsteemide faasikarakteristikud ja rühmahilistused summeeruvad, võime järelda, et *kõigi sama amplituudikarakteristikut omavate süsteemide hulgas minimaalse faasiga süsteemi rühmahilistus on väikseim*. Seega hilineb signaal minimaalse faasiga süsteemis kõige vähem. Süsteemi, mille kõik nullid on ühikringist väljaspool, nimetatakse *maksimaalse faasiga süsteemiks* ja kõigist sama amplituudikarakteristikut omavatest süsteemides hilineb signaal sellises süsteemis kõige rohkem.

4.8 Lineaarne faasikarakteristik

Faasikarakteristiku linearsus on oluline LTI-süsteemide disainimise kriteerium. Kui faasikarakteristik on lineaarne, on selle tuletis (ehk rühmahilistus) konstantne ja seega kõik sisendsignaali sageduskomponendid hilistuvad süsteemis võrdselt. Seega säilitavad lineaarse faasikarakteristikuga süsteemid signaali kuju nende sageduste osas, mida süsteem laseb läbi. See on oluline omadus füsioloogiliste signaalide analüüsil. Oletame näiteks, et eesmärgiks on disainida filter, mis eemaldab EKG-signaalist müra. Kuna arst teeb otsuse EKG-signaali kuju põhjal, on oluline, et filter signaali kuju ei moonutaks. Seetõttu on sellisel juhul otstarbekas kasutada lineaarse faasikarakteristikuga filtrit. Samas näiteks arvutades EEG-signaali võimsuste suhet eri sagedusribadel, ei ole kasutatava filtri faasikarakteristiku linearsus tingimata vajalik.

Digitaalsete süsteemide disainimismeetoditest tulenevalt on lineaarse faasikarakteristikuga süsteemid üldjuhul lõpliku impulsskajaga (FIR-süsteemid). Järgnevalt vaatleme, millal LTI-süsteemil on lineaarne faasikarakteristik ja kuidas lineaarse faasikarakteristikuga LTI-süsteeme liigitatakse.

Kui süsteemil on lineaarne faasikarakteristik, võime tema sageduskarakteristiku esitada järgmisel kujul:

$$H(\omega) = |H(\omega)| e^{-j\alpha\omega} \quad (4.20)$$

Faasikarakteristik avaldub seega $\angle\{H(\omega)\} = -\alpha\omega$.

Üldiselt käsitletakse lineaarse faasikarakteristiku mõistet siiski laiemalt, arvates selliste süsteemide hulka ka süsteemid, mille faasikarakteristik avaldub $\angle\{H(\omega)\} = -\alpha\omega + \beta$.

Selgub, et süsteemil on lineaarne faasikarakteristik (laiemas mõttes), kui tema impulsskaja on paaris- või paaritusümmeetriline. Selleks et süsteem oleks kausaalne, peab tema impulsskaja kohta kehtima $h[n] = 0$, $n < 0$. Eeldades, et impulsskaja kestus on $M + 1$ diskreeti, avaldub sellise süsteemi impulsskaja paarissümmeetria järgmisel kujul:

$$h[n] = \begin{cases} h[M - n], & 0 \leq n \leq M \\ 0, & \text{mujal} \end{cases} \quad (4.21)$$

Sageduskarakteristik on sel juhul väljendatav järgmiselt:

$$H(\omega) = A_e(e^{j\omega})e^{-j\omega\frac{M}{2}}, \quad (4.22)$$

kus $A_e(e^{j\omega})$ on reaalne paarissümmeetriline sageduse funktsioon.

Impulsskaja paaritu sümmeetria avaldub järgmisel kujul:

$$h[n] = \begin{cases} -h[M-n], & 0 \leq n \leq M \\ 0, & \text{mujal} \end{cases} \quad (4.23)$$

ja sellise süsteemi sageduskarakteristiku võime väljendada järgmiselt:

$$H(\omega) = A_o(e^{j\omega})e^{-j\omega\frac{M}{2}+j\frac{\pi}{2}} \quad (4.24)$$

Siin on $A_o(e^{j\omega})$ reaalne paaritu sümmeetriaga sageduse funktsioon.

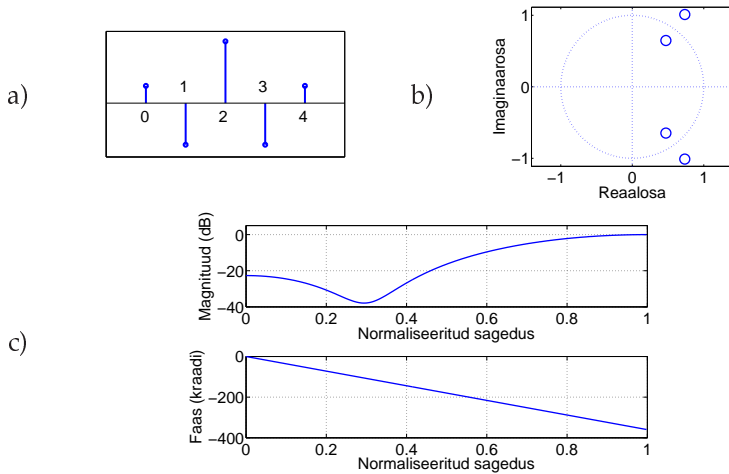
Lisaks paaris- ja paaritule sümmeetriale on süsteemi omaduste seisukohalt tähtis ka impulsskaja diskreetide arvu $M + 1$ paarilisus. Lineaarse faasikarakteristikuga FIR-süsteemid jagataksegi tabelis 4.1 esitatud moel nelja kategooriasse.

Tabel 4.1 Lineaarse faasikarakteristikuga FIR-süsteemide jaotus nelja kategooriasse

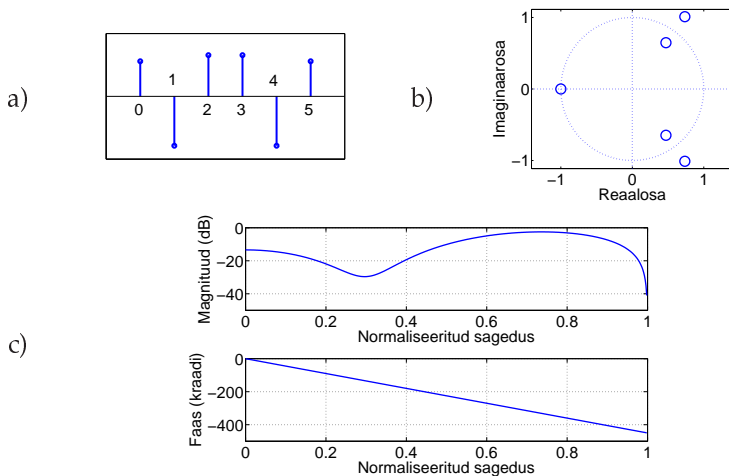
	M paaris	M paaritu
$h[n] = h[M-n]$	I kategooria FIR-süsteem	II kategooria FIR-süsteem $H(z) _{z=-1} = 0$
$h[n] = -h[M-n]$	III kategooria FIR-süsteem $H(z) _{z=-1,1} = 0$	IV kategooria FIR-süsteem $H(z) _{z=1} = 0$

Joonistel 4.5 . . . 4.8 on esitatud näited lineaarse faasikarakteristikuga FIR-süsteemidest. Kui M on paaris (s.t impulsskaja diskreetide arv on paaritu), satub impulsskaja sümmeetriatelg diskreedi $\frac{M}{2}$ kohale. Kui M on paaritu, jääb impulsskaja sümmeetriatelg kahe diskreedi vahele. III kategooria süsteemi impulsskaja keskmise diskreedi väärtus on alati 0.

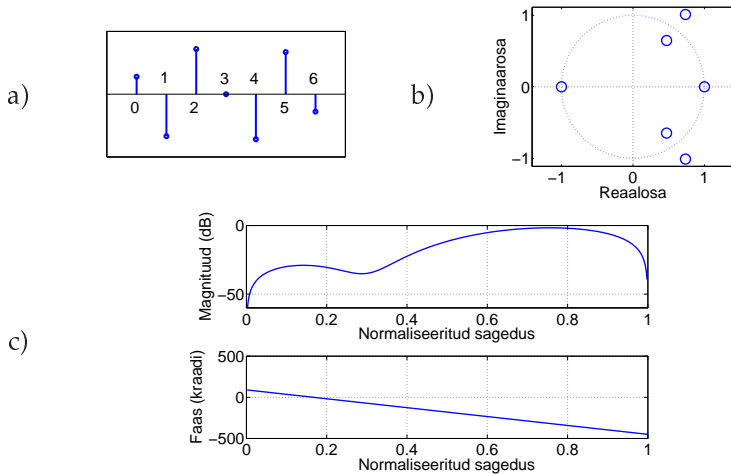
Kasutades impulsskaja sümmeetriaomadusi Z -teisenduse valemis, on võimalik näidata, et kui lineaarse faasikarakteristikuga FIR-süsteemil on null kohas $re^{j\varphi}$, siis peab tal olema null ka kohas $\frac{1}{r}e^{-j\varphi}$. Kui nõuda süsteemilt ka reaalseid kordajaid, peavad sellise süsteemi nullid paiknema neljastes



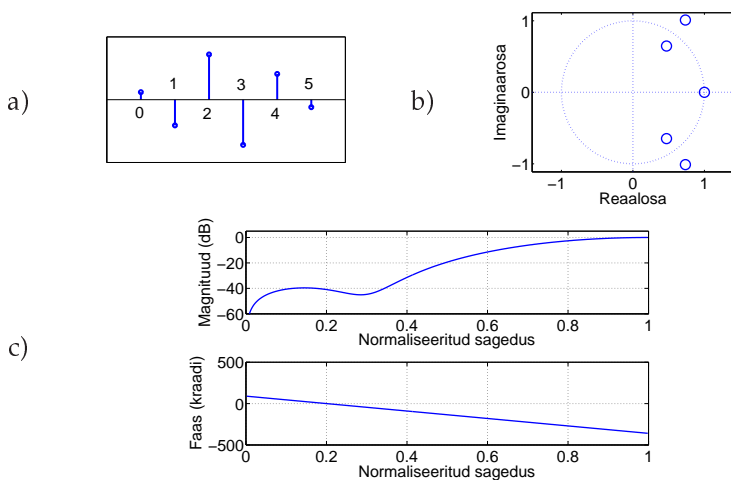
Joonis 4.5 Näide I kategooria lineaarse faasikarakteristikuga FIR-süsteemist: a) impulsskaja, b) nullide diagramm, c) sageduskarakteristik



Joonis 4.6 Näide II kategooria lineaarse faasikarakteristikuga FIR-süsteemist: a) impulsskaja, b) nullide diagramm, c) sageduskarakteristik



Joonis 4.7 Näide III kategooria lineaarse faasikarakteristikuga FIR-süsteemist: a) impulsskaja, b) nullide diagramm, c) sageduskarakteristik



Joonis 4.8 Näide IV kategooria lineaarse faasikarakteristikuga FIR-süsteemist: a) impulsskaja, b) nullide diagramm, c) sageduskarakteristik

rühmades (lisaks mainitud kohtadele veel kohtades $re^{-j\varphi}$ ja $\frac{1}{r}e^{j\varphi}$). See on näha ka joonistelt 4.5 . . . 4.8.

Lisaks sellele osutub, et

- II kategooria süsteemil on alati null kohas $z = -1$ (vastavalt sageduskarakteristiku jaoks $H(\omega)|_{\omega=\pi} = 0$);
- III kategooria süsteemil on alati null kohtades $z = -1$ ja $z = 1$ (vastavalt $H(\omega)|_{\omega=0,\pi} = 0$);
- IV kategooria süsteemil on alati null kohas $z = 1$ (vastavalt $H(\omega)|_{\omega=0} = 0$).

5 Filtrite disainimine

Diskreetsete lineaarsete süsteemide disainimise puhul on kõigepealt vaja kindlaks teha, kas antud kasutusvaldkond eeldab süsteemilt lineaarset faasikarakteristikut. Lineaarse faasikarakteristiku eeliseks on, et kõik sageduskomponendid hilistuvad ajas ühepalju ja seega signaali kuju ei moonutu. Samas on ebalineaarse faasikarakteristikuga süsteeme võimalik samade kriteeriumide juures realiseerida oluliselt väiksema arvutusliku mahuga. Disainimismetodoloogias tingitult on lineaarse faasikarakteristikuga süsteemid üldjuhul FIR-süsteemid ja nad realiseeritakse mitterekursiivselt.

Pidevate signaalide ja analoogsete süsteemide puhul pole lõplik impulsskaja võimalik, küll aga on olemas põhjalik teoreetiline taust analoogsete lõpmatu impulsskajaga (IIR) filtrite disainimisel. Diskreetsed IIR-filtrid disainitaksegi üldjuhul vastavate analoogsete prototüüpfiltrite abil.

5.1 FIR-filtrite disainimine

5.1.1 FIR-filtrite disainimine akna meetodil

FIR-filtrite disainimisel akna meetodi abil lähtutakse ideaalsest sageduskarakteristikust $H_d(\omega)$. Enamasti valitakse ideaalseks sageduskarakteristikuks nelinurkne funktsioon, mille väärtus on 1 läbilaskeribal ja 0 tõkkeribal. Oluline on, et ideaalsest sageduskarakteristikust oleks võimalik võtta Fourier' pöördteisendus. Filtri disainimist võib kirjeldada järgmiste sammude abil:

- Ideaalsele sageduskarakteristikule vastava impulsskaja $h_d[n]$ leidmine Fourier' pöördteisenduse abil. Kui $H_d(\omega)$ sisaldab hüppeid, siis $h_d[n]$ on lõpmatu kestusega.
- Et filter oleks realiseeritav mitterekursiivselt, peame ideaalset impulsskaja piirama, korrutades ta läbi aknafunktsiooniga $w[n]$: $h[n] = h_d[n] \cdot w[n]$. Samuti peame impulsskaja nihutama paremale (hilistama), et disainitav süsteem oleks kausaalne.

- Saadud filtri amplituudikarakteristikut võime analüüsida, võttes arvesse, et korrutamise aknafunktsiooniga aja tasandil vastab konvolutsioonile aknafunktsiooni Fourier' teisendusega $W(\omega)$ sageduse tasandil. Osutub, et saadud filtri amplituudikarakteristik erineb ideaalse filtri amplituudikarakteristikust kahel moel:
 - saadud filtri üleminekuriba on lauge,
 - saadud filtri läbilaske- ja tõkkeribadel esineb laineid (ingl.k *ripples*).

Akna meetodi puhul saame filtri amplituudikarakteristiku kuju mõjutada kahel moel:

- valides sobiva filtri järgu ehk akna laiuse,
- valides sobiva aknafunktsiooni kuju.

Filtri järgu kasvatamine muudab üleminekuriba järsumaks; aknafunktsiooni valikuga on võimalik vähendada ka vea suurust (lainete amplituudi) läbilaske- ja tõkkeribal.

Tuntumad aknafunktsioonid on *hamming*, *hanning*, *blackman*, *bartlett*. Aknafunktsiooni kuju mõju filtri amplituudikarakteristikule võime hinnata aknafunktsiooni Fourier' teisenduse põhjal. Mida laiem on aknafunktsiooni Fourier' teisenduse ($|W(\omega)|$) pealaine, seda laugem on disainitava filtri üleminekuriba. $|W(\omega)|$ kõrvallainete kõrgus määrab ära disainitava filtri vea ülemineku- ja tõkkeribadel.

FIR-filtri disainimisel akna meetodi abil on järgmisi puudusi:

- disainimine toimub n -ö katse-eksituse meetodil – pole võimalik leida optimaalset aknafunktsiooni ja filtri järku etteantud kriteeriumite põhjal;
- filtri vead läbilaske- ja tõkkeribal on omavahel seotud.

5.1.2 Kaiseri aknafunktsioonide pere

Kaiser töötas välja aknafunktsioonide pere, mille puhul on võimalik optimaalsele lähedase aknafunktsiooni ja filtri järgu leidmine etteantud kriteeriumite põhjal. Kaiseri aknafunktsiooni võime arvutada järgmise valemi abil:

$$w[n] = \begin{cases} \frac{I_0[\beta(1-[(n-\alpha)/\alpha^2]^2)^{0.5}]}{I_0(\beta)} & 0 \leq n \leq M \\ 0 & \text{mujal} \end{cases} \quad (5.1)$$

kus $I_0(\cdot)$ on nullindat järku Besseli funktsioon, $\alpha = \frac{M}{2}$ (M on filtri järk) ja β on parameeter, mis reguleerib aknafunktsiooni kuju.

Kaiser leidis empiiriliselt valemid β ja M leidmiseks filtri kriteeriumite järgi. Filtri kriteeriumiteks on

- tõkkeriba lõikesagedus ω_s ,
- läbilaskeriba lõikesagedus ω_p ,
- suurim lubatud viga läbilaske- ja tõkkeribal δ .

Kasutades parameetreid

$$\Delta\omega = \omega_s - \omega_p \quad \text{ja} \quad A = -20 \log_{10} \delta, \quad (5.2)$$

võime aknafunktsiooni parameetri β leida järgmiselt:

$$\beta = \begin{cases} 0.1102(A - 8.7) & A > 50 \\ 0.5842(A - 21)^{0.4} + 0.07886(A - 21) & 21 \leq A \leq 50 \\ 0 & A < 21 \end{cases} \quad (5.3)$$

Filtri järgu võime leida valemiga

$$M = \frac{A - 8}{2.285\Delta\omega} \quad (5.4)$$

Need valemid on siiski vaid ligikaudsed ja filtri vastavust kriteeriumitele tuleb kontrollida, analüüsides disainitud filtrit.

Kaiseri aknafunktsiooni puhul kaob vajadus disainida katse-eksituse meetodil. Vead läbilaske- ja tõkkeribal on aga endiselt teineteisest sõltuvad.

Eespool eeldasime, et soovitud filter on madalpääskarakteristikuga. Muud filtrite tüübid (kõrgpääs-, riba- ja tõkkefilter) saame, lahutades omavahel erinevate madalpääsfiltrite amplituudikarakteristikuid (aja tasandil vastavalt impulsskajasid).

5.1.3 FIR-filtrite disainimine Remezi meetodil

Disainitud filtri vastavust ideaalsele filtrile võime hinnata ruutkeskmise veaga ideaalse ja disainitud filtri amplituudikarakteristikute vahel:

$$\epsilon = \sqrt{\int_{-\pi}^{\pi} |H_d(\omega) - H(\omega)|^2 d\omega} \quad (5.5)$$

Akna meetod nelinurkse aknafunktsiooni puhul minimeerib selle vea. Ruutkeskmine viga ei ole tihti siiski otstarbekas veakriteerium, sest akna meetodil disainides saame üldjuhul filtri, mille amplituudikarakteristikus on suured lained üleminekuriba läheduses (Gibbsi efekt). Need ülelöögid muutuvad kitsamaks filtri järgu kasvades, kuid nende amplituud ei vähene. Seega võimendab akna meetodi abil disainitud filter sagedusi kitsas ribas filtri üleminekuriba läheduses.

Filtrite disainimise puhul on otstarbekam minimeerida maksimaalne viga:

$$E(\omega) = V(\omega)(|H(\omega) - H_d(\omega)|) \quad (5.6)$$

Siin on $V(\omega)$ vea kaalufunktsioon, mis võimaldab anda eri sagedusribadele erinevaid veakriteeriume.

FIR-filtrite disainimine kaalutud maksimaalse vea minimeerimise teel toimub iteratiivsete algoritmide abil (Parks-McClellani algoritm). Tulemuseks on filter, mille amplituudikarakteristikus on võrdse kõrgusega lained (*equi-ripple design*) läbilaske- ja tõkkeribal. Samas võivad kaalufunktsioonist tulenevalt olla kummalgi ribal lainete kõrgused erinevad, teisisõnu, viga eri ribadel on võimalik ette anda sõltumatult. Osutub, et sellise disaini puhul esineb maksimaalne viga kõige rohkem $\frac{M}{2} + 3$ sagedusel. Maksimaalne viga esineb alati sagedustel ω_s ja ω_p ehk vastavalt tõkkeriba ja läbilaskeriba piirsagedustel.

Filtri järgu suurust on võimalik hinnata filtri kriteeriumite põhjal järgmise valemi abil:

$$M = \frac{-10 \log_{10}(\delta_1 \delta_2) - 13}{2.324 \Delta \omega}, \quad (5.7)$$

kus δ_1 ja δ_2 on maksimaalsed lubatud vead läbilaske- ja tõkkeribal.

5.2 IIR-filtrite disainimine

Diskreetsed IIR-filtrid disainitakse analoogsete prototüüpfiltrite kaudu, seetõttu tutvume järgnevalt analoogsete filtrite põhiklassidega. Seejärel vaatleme kahte meetodit diskreetsete IIR-filtrite disainimisel analoogsete prototüüpfiltrite põhjal.

5.2.1 Analoogete filtrite põhiklassid

Analoogsed "maailmas" kehtivad sarnaselt diskreetse "maailmaga" järgmised reeglid:

$$\begin{aligned} h(t) &\xrightarrow{\text{Fourier' teiesndus}} H(\Omega) \\ h(t) &\xrightarrow{\text{Laplace'i teisendus}} H(s) \end{aligned} \quad (5.8)$$

Kompleksmuutuja on siis analoogsed "maailmas" $s = \sigma + j\Omega$ ja kompleks-tasandit kutsutakse s -tasandiks. Erinevalt diskreetse "maailma" z -tasandist

- vastab analoogete süsteemide sageduskarakteristikule süsteemifunktsiooni $H(s)$ käitumine s -tasandi imaginaarteljel ($s = j\Omega$);
- peavad süsteemifunktsiooni $H(s)$ poolused paiknema vasakul pooltasandil, et analoogne süsteem oleks stabiilne ja kausaalne.

Järgnevalt vaatleme nelja analoogete filtrite põhiklassi.

Butterworthi filter

Butterworthi filtri sageduskarakteristiku magnituudi ruut avaldub valemist

$$|H(\Omega)|^2 = \frac{1}{1 + \left(\frac{j\Omega}{\Omega_c}\right)^{2N}}, \quad (5.9)$$

kus Ω_c on filtri lõikesagedus. Süsteemifunktsioonile kehtib seega

$$H(s)H(-s) = \frac{1}{1 + \left(\frac{s}{j\Omega_c}\right)^{2N}} \quad (5.10)$$

Avaldise $H(s)H(-s)$ poolused paiknevad s -tasandil ringjoonel, mille raadius on Ω_c , üksteisest võrdsel kaugusel:

$$s_k = \Omega_c e^{(j\frac{\pi}{2N})(2k+N-1)}, \quad k = 0, 1, \dots, 2N-1 \quad (5.11)$$

Nendest poolustest valitakse filtri $H(s)$ süsteemifunktsiooni poolused, mis paiknevad s -tasandi vasakul poolel.

Butterworthi filtri amplituudikarakteristik on monotoonne ja seda järsuma üleminekuribaga, mida kõrgem on filtri järk N .

1. tüüpi Chebyshevi filter

1. tüüpi Chebyshevi filtri sageduskarakteristiku magnituudi ruut avaldub valemist

$$|H(\Omega)|^2 = \frac{1}{1 + \varepsilon^2 V_N^2\left(\frac{\Omega}{\Omega_c}\right)}, \quad (5.12)$$

kus $V_N(\cdot)$ on N -ndat järku Chebyshevi polünoom.

1. tüüpi Chebyshevi filtri puhul

- paiknevad avaldise $H(s)H(-s)$ poolused ovaalil sümmeetriliselt kummagi s -tasandi telje suhtes;
- on amplituudikarakteristik läbilaskeribal laineline, kuna tõkkeribal käitub ta monotoonselt.

2. tüüpi Chebyshevi filter

2. tüüpi Chebyshevi filtri sageduskarakteristiku magnituudi ruut avaldub valemist

$$|H(\Omega)|^2 = \frac{1}{1 + (\varepsilon^2 V_N^2\left(\frac{\Omega_c}{\Omega}\right))^{-1}} \quad (5.13)$$

2. tüüpi Chebyshevi filtri puhul on amplituudikarakteristik läbilaskeribal monotoonne, kuna tõkkeribal esineb võrdsete amplituudidega laineid.

Elliptiline filter

Elliptilise filtri sageduskarakteristiku magnituudi ruut avaldub valemist

$$|H(\Omega)|^2 = \frac{1}{1 + \varepsilon^2 U_N^2(\Omega)}, \quad (5.14)$$

kus $U_N(\cdot)$ on Jacobi elliptiline funktsioon.

Elliptilise filtri puhul

- paiknevad avaldise $H(s)H(-s)$ poolused ellipsil sümmeetriliselt kummagi s -tasandi telje suhtes;
- on amplituudikarakteristik nii läbilaske- kui ka tõkkeribal laineline.

Üldjuhul annab sama filtri järgu juures Butterworthi filter kõige väiksema ebalineaarsusega faasikarakteristiku. Samas nõuab Butterworthi filter suurimat järku, et rahuldada samad kriteeriumid. Elliptilise filtri faasikarakteristik on kõige ebalineaarsem, kuid samas on järk kõige madalam.

5.2.2 IIR-filtrite disainimine impulsskaja diskreetimise teel

Diskreetseid IIR-filtreid on võimalik disainida analoogsete prototüüpfiltreite põhjal, diskreetides prototüüpfiltri impulsskaja

$$h[n] = T_d h_c(nT_d), \quad (5.15)$$

kus T_d on vastav diskreetimissamm. Alaindeks c rõhutab seda, et avaldise paremal poolel on tegemist pideva funktsiooniga.

Praktikas toimub kogu disainimisprotsess kompleksmuutujate (s ja z) tasandil; järgneva eesmärgiks on uurida, millise mäpingu analoogse filtri impulsskaja diskreetimine annab s -tasandi ja z -tasandi vahel.

Lõigus 3.3 tõdesime, et pideva funktsiooni diskreetimine põhjustab sageduse tasandil selle funktsiooni sagedusspektri perioodilise kordumise ja sellest tulenevalt on diskreetsete funktsioonide sagedusesitus alati perioodiline. Kuna filtri sageduskarakteristik on impulsskaja Fourier' teisendus, moodustub ka antud meetodil disainitud diskreetse filtri sageduskarakteristik analoogse filtri sageduskarakteristiku koopiaste summast:

$$H(\omega) = \sum_{k=-\infty}^{\infty} H_c \left(j \frac{\omega}{T_d} + j \frac{2\pi}{T_d} k \right) \quad (5.16)$$

Selles valemis on arvesse võetud sagedustelje skaala muutus diskreetimisel: $\omega = \Omega T_d$.

Kuna analoogses "maailmas" on sagedustelg s -tasandi imaginaartelg (mis ulatub $-\infty \dots \infty$), aga diskreetises "maailmas" on sagedustelg z -tasandi ühikring (mis on lõplik ja ulatub $-\pi \dots \pi$), siis kujutab diskreetse filtri disainimine analoogse prototüüpfiltri põhjal endast s -tasandi imaginaartelje mäpingut z -tasandi ühikringile. Antud disainimismeetodi puhul on see mäping lineaarne ($\omega = \Omega T_d$). Seega, selleks et analoogse prototüüpfiltri sageduskarakteristik ei moonutuks, peaks ta olema piiratud ribaga ja T_d peaks olema piisavalt suur (vt lõik 3.3). Praktiliselt pole ühegi analoogse filtri sageduskarakteristik piiratud sagedusribaga ja seega põhjustab selline mäping sageduskarakteristikus moonutusi, teisisõnu, saadud diskreetse filtri sageduskarakteristik ei vasta täpselt prototüüpfiltri sageduskarakteristikule.

Järgnevalt vaatleme, mis toimub disainimisprotsessis kompleksmuutuja tasandil. Analoogse filtri süsteemifunktsiooni $H_c(s)$ on võimalik esitada

osamurdude kujul järgmiselt:

$$H_c(s) = \sum_{k=1}^N \frac{A_k}{s - s_k} \quad (5.17)$$

Võttes osamurdudest Laplace'i pöördteisenduse, saame analoogse filtri impulsskaja:

$$h_c(t) = \begin{cases} \sum_{k=1}^N A_k e^{s_k t}, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad (5.18)$$

Nüüd võime rakendada diskreetimist:

$$\begin{aligned} h[n] &= T_d h_c(nT_d) \\ &= \sum_{k=1}^N T_d A_k e^{s_k n T_d} u[n] \\ &= \sum_{k=1}^N T_d A_k \left(e^{s_k T_d} \right)^n u[n] \end{aligned} \quad (5.19)$$

Sellest summast aga on võimalik võtta Z -teisendus, mis annab meile vastava diskreetse filtri süsteemifunktsiooni:

$$H(z) = \sum_{k=1}^N \frac{T_d A_k}{1 - e^{s_k T_d} z^{-1}} \quad (5.20)$$

Seega on analoogse filtri poolused s_k kandunud üle diskreetse filtri poolusteks $z_k = e^{s_k T_d}$.

Vaatleme diskreetse filtri pooluseid, mis on saadud analoogse filtri s -tasandi vasakul poolel paiknevate pooluste mäpingul:

$$e^{(\sigma_k + j\Omega_k)T_d} = e^{\sigma_k T_d} e^{j\Omega_k T_d}, \quad \sigma_k < 0 \quad (5.21)$$

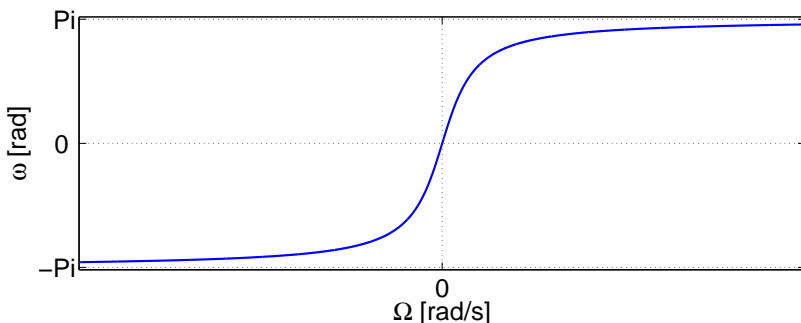
Kuna T_d on alati positiivne, siis antud tingimus ($\sigma_k < 0$) annab

$$e^{\sigma_k T_d} < 1 \quad (5.22)$$

ja seega kehtib diskreetse filtri poolustele

$$\left| e^{s_k T_d} \right| < 1 \quad (5.23)$$

Järelikult kannab antud meetodi poolt loodav mäping analoogse filtri s -tasandi vasakul poolel olevad poolused alati üle diskreetse filtri z -tasandi ühikringi sisse. Seega saame stabiilse prototüüpfiltri põhjal alati stabiilse diskreetse filtri.



Joonis 5.1 Bilineaarne teisendus sagedustelgede Ω ja ω vahel

5.2.3 IIR-filtrite disainimine bilineaarse teisenduse meetodil

Bilineaarse teisenduse meetodi puhul lähtutakse otse mäpingust s - ja z -tasandite vahel. Mäpingut nimetatakse bilineaarseks teisenduseks:

$$s = \frac{2}{T_d} \left(\frac{1 - z^{-1}}{1 + z^{-1}} \right); \quad z = \frac{1 + \frac{T_d}{2}s}{1 - \frac{T_d}{2}s}. \quad (5.24)$$

Ka selle mäpingu puhul kandub s -tasandi vasak pool üle z -tasandi ühikringi sisse. Mäping sagedustelgede vahel pole aga enam lineaarne. Kui panna teisenduse valemisse sisse $s \rightarrow j\Omega$ ja $z \rightarrow e^{j\omega}$ ehk vaadelda mäpingut sagedustelgede vahel, saame

$$\Omega = \frac{2}{T_d} \tan\left(\frac{\omega}{2}\right); \quad \omega = 2 \arctan\left(\Omega \frac{T_d}{2}\right) \quad (5.25)$$

Jooniselt 5.1 on näha, et bilineaarse teisenduse puhul kandub s -tasandi lõpmatu imaginaartelg üle z -tasandi lõplikule ühikringile, surudes esimest kokku seda enam, mida kõrgemad on sagedused. 0-punkti ümbruses on mäping ligikaudu lineaarne.

Tänu ebalineaarsusele bilineaarse teisenduse meetodi puhul analoogse prototüüpfiltri sageduskarakteristiku koopiate ülekatte probleemi ei teki, samas moonutub analoogse prototüüpfiltri sageduskarakteristik mäpingu ebalineaarsuse tõttu, seda eriti kõrgetel sagedustel.

Näide 5.1 IIR-filtrite disainimine

Olgu filtril järgmised kriteeriumid:

- läbilaskeriba piirsagedus 0.2π ,
- tõkkeriba piirsagedus 0.25π ,
- maksimaalne viga läbilaskeribal 0.05 ,
- maksimaalne viga tõkkeribal 0.01 (ehk -40 dB).

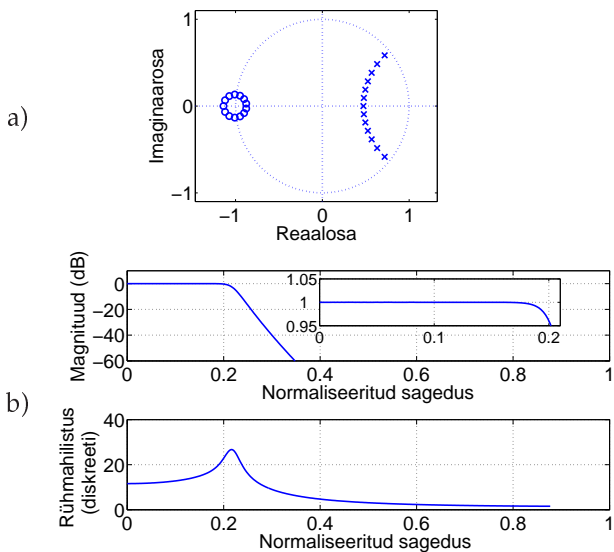
Joonistel 5.2 ... 5.5 on esitatud nende kriteeriumite järgi disainitud diskreetsed filtrid, kusjuures prototüüpfiltritena on kasutatud kõikidesse eespool toodud analoogsete filtrite põhiklassidesse kuuluvaid filtreid. Joonistel on ära toodud filtrite nullide ja pooluste diagramm, amplituudikarakteristik ning rühmahilistus. Amplituudikarakteristiku juures on eraldi näidatud lineaarses skaalas selle käitumine läbilaskeribal. Butterworthi filtri puhul on rühmahilistuse kõver jäetud lõpuni välja joonistamata, kuna kõrgetel sagedustel on filtri sumbuvus sedavõrd suur, et numbriline resolutsioon ei ole rühmahilistuse korrektseks arutamiseks piisav.

Eespool toodud kriteeriumite rahuldamiseks vajalikud filtrite järgud on järgmised:

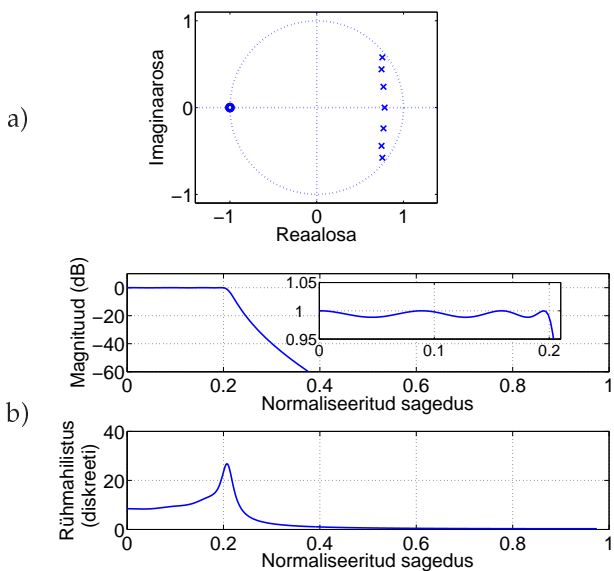
- Butterworthi filter: $N = 13$;
- Chebyshevi 1. ja 2. tüüpi filtrid: $N = 7$;
- elliptiline filter: $N = 5$.

Praktikas leiab ehk diskreetsetest IIR-filtritest kõige rohkem kasutamist Butterworthi filter, kuna tema eeliseks on monotoonne amplituudikarakteristik.

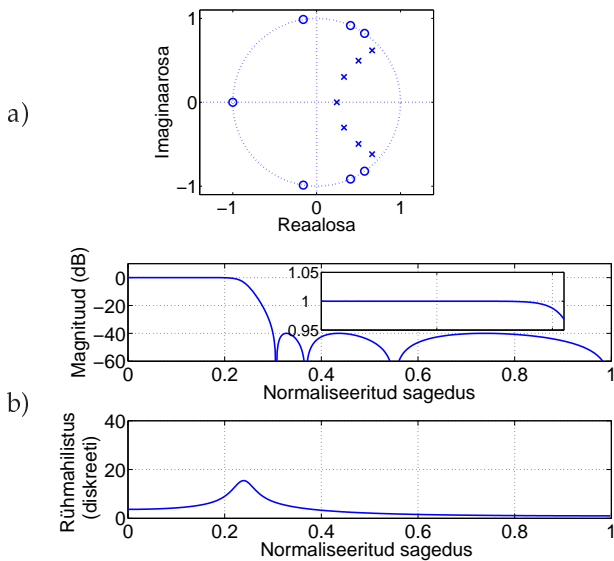
Eespool vaatlesime madalpääsfiltrite disainimist. Kõrgpääs-, riba- ja tõkkefiltrite puhul disainitakse eelnevalt vastav madalpääsfilter ja seejärel kasutatakse sagedusteisendusi. Sagedusteisendused on üldjuhul filtrite disainimiseks kasutatavatesse tarkvarapakettidesse (näiteks MatLab) sisse integreeritud.



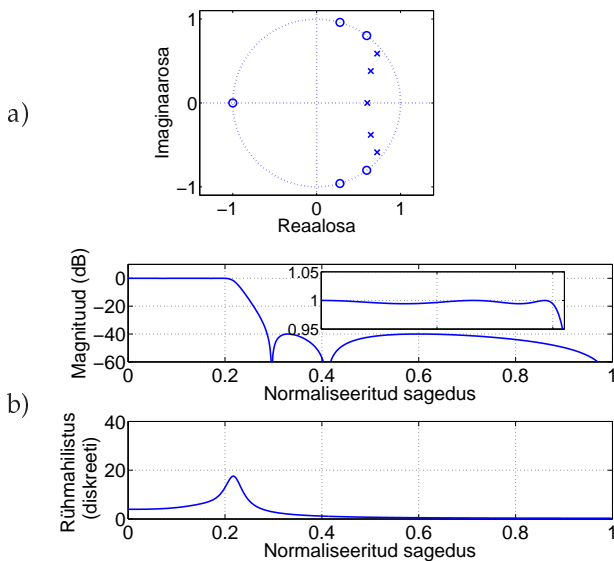
Joonis 5.2 Etteantud kriteeriumite järgi disainitud Butterworthi filter



Joonis 5.3 Etteantud kriteeriumite järgi disainitud Chebyshevi 1. tüüpi filter



Joonis 5.4 Etteantud kriteeriumite järgi disainitud Chebyshevi 2. tüüpi filter



Joonis 5.5 Etteantud kriteeriumite järgi disainitud elliptiline filter

6 Meditsiinilise kujutise töötlemine

Käesoleva peatüki eesmärgiks on anda lugejale põgus ülevaade kujutise töötlemise vahenditest. Kui eelmistes petükkides vaatlesime diskreetsete süsteemide analüüsi ja disainimist teoreetilisest seisukohast, siis kujutise töötlemisel lähtume praktilistest vahenditest. Selline lähenemisviis on otsustav, kuna meditsiinilise kujutise töötlemisel kasutatakse enamasti valmis operaatoreid (näiteks *sobel*, *prewitt*, *laplacian*), selle asemel et disainida etteantud kriteeriumite järgi süsteeme. Teisest küljest ei mahu kahedimensiooniliste süsteemide disainimise teooria käesoleva materjali raamidesse.

See peatükk lähtub suures osas MatLabi programmeerimiskeskonna võimalustest kujutise töötlemisel. Mitmeid kujutisega teostatavaid operatsioone on suurepäraselt illustreeritud materjali harjutuste osas.

6.1 Kujutise formaadid

Kujutis koosneb kujutise elementidest ehk *pikslitest* (tuleneb sõnadest *picture element*). 3-D kujutiste puhul kasutatakse ka terminit *voksel*. 2-D kujutist käsitletakse üldjuhul kahedimensioonilise maatriksina. MatLabi puhul paikneb maatriksi element (1,1) kujutise vasakus ülanurgas, esimene indeks viitab reale ja teine veerule.

Värviliste kujutiste salvestamiseks kasutatakse MatLabis kahte erinevat formaati:

- *RGB-formaadi* puhul esitatakse kujutis maatriksi kujul, mille mõõtmed on $M \times N \times 3$. Viimane koordinaat viitab põhivärvile (*red-green-blue*). RGB-formaat on sobiv kujutisega opereerimisel, kuid mälu suhtes on ta ebaefektiivne.
- *Indekseeritud formaat*: iga piksel on määratud indeksiga, mille abil viidatakse värvikaardile (*colormap*). Värvikaart on $(L \times 3)$ maatriks, mille veerud vastavad põhivärvide intensiivsusele vahemikus $0 \dots 1$, ja

L on värvikaardis esitatud värvitoonide arv. Indekseeritud formaat võimaldab säästa kujutise salvestamiseks vajalikku mälu hulka, kuid samas on ta arvutuslikeks operatsioonideks ebasobiv.

Halltoon-kujutise (*grayscale image*) esitamiseks kasutatakse *intensity*-formaati, mis kujutab endast $M \times N$ maatriksit, kus iga väärtus esitab vastava piksli intensiivsust halltoon-skaalal. Samuti on olemas oma formaat mustvalgete (binaarsete) kujutiste esitamiseks.

Kujutise väärtused võivad MatLab'i keskkonnas olla salvestatud kolme erinevat andmetüüpi kasutades: 8-bitiste täisarvudena (*uint8*), 16-bitiste täisarvudena (*uint16*) või kahekordse täpsusega ujukoma-formaadis (*double*). Viimasel puhul on iga väärtuse salvestamiseks vaja 32 bitti.

Eri formaatide vahel on võimalik kujutist teisendada näiteks järgmiste käskude abil:

- eri formaatidest indekseeritud formaati: `gray2ind`, `grayslice`, `rgb2ind`;
- eri formaatidest *intensity*-formaati: `ind2gray`, `mat2gray`, `rgb2gray`;
- indekseeritud formaadist RGB-formaati: `ind2rgb`;
- eri formaatidest binaarformaati: `im2bw`.

Kujutisi võib MatLab'i keskkonnas failist lugeda, faili kirjutada ja monitoril esitada vastavalt käskude `imread`, `imwrite` ja `imshow` abil.

Kujutistest võib moodustada montaaže ja neid võib monitoril esitada `montage`-käsu abil. Kaadritest on võimalik koostada filmilõik `imovie`-käsu abil ja seda on võimalik vaadata monitoril `movie`-käsuga.

Kõige lihtsamad operatsioonid kujutistega toimivad $n \times n$ piksel-pikselilt. MatLab võimaldab järgmisi operatsioone:

- `imabsdiff`: kahe kujutise vahe absoluutväärtus,
- `imcomplement`: kujutise komplement,
- `imadd`: kahe kujutise summa,
- `imsubtract`: kahe kujutise vahe,
- `imddivide`: kahe kujutise jagatis,
- `immultiply`: kahe kujutise korrutis,
- `imlincomb`: kujutiste lineaarne kombinatsioon.

6.2 Kujutise filtreerimine

Nii nagu peatükis 5 esitatud diskreetsete filtrite puhul saadakse väljundsignaali väärtused filtri aknasse jäävate sisendsignaali diskreetide kaudu, nii on ka kujutise puhul väljundpiksli väärtuse arvutamisel enamasti otstarbekas arvesse võtta vastava sisendpiksli naabruse jäävate pikslite väärtused. Kujutise puhul sisaldab filtri aken antud piksli suhtes nii horisontaal- kui ka vertikaalsuunas olevaid naaberpikseid. Samuti on kujutist filtreerides vaja akent edasi nihutada nii horisontaal- kui ka vertikaalsuunas.

Sarnaselt 1-D signaalidega tekib ka kujutise puhul probleem, kui akna keskpunkt on kujutise serval ja seega osa aknast jääb kujutisest välja. Sel juhul lisab enamik MatLab'i funktsioone kujutisest väljapoole jäävate pikslite kohale aknasse nulle. Näiteks `imfilter`-käsuga on aga võimalik määrata erinevaid mooduseid kujutisest väljapoole jäävate pikslite täitmiseks.

MatLab'i keskkonnas on mitmeid käsk 2-D signaalide töötlemiseks jooksvas aknas ehk filtreerimiseks. Üks selline käsk on `nlfilter`, mille sisendmuutujateks on töödeldav kujutis, akna mõõtmed, töötlev funktsioon ning vajaduse korral töötleva funktsiooni parameetrid. Akna keskmine piksel ehk piksel, mille väärtust väljundkujutises antud akna asukohta puhul arvutatakse, on MatLab'is määratud akna mõõtmete kaudu järgnevalt: `floor(([M N] + 1)/2)`. Näiteks 3×3 akna puhul on keskmiseks pikslik teises reas ja teises veerus paiknev piksel ja 4×6 akna puhul teises reas ja kolmandas veerus paiknev piksel. Üldjuhul kasutatakse paaritu arvu ridade ja veergudega aknaid, sest sel juhul on akna kuju keskpunkti suhtes sümmeetriline. Töötleva funktsiooni võib `nlfilter`-käsule ette anda kolmel moel:

- 1) valides funktsiooni MatLab'is olemasolevatest funktsioonidest (näiteks kujutise jooksvat keskväärtust on võimalik arvutada, kasutades MatLab'is olemasolevat `mean2`-funktsiooni, mille puhul filtreerimiskäsk 5×5 aknaga näeb välja järgmiselt: `Iout = nlfilter(Iin, [5 5], 'mean2')`);
- 2) määraes failisanga (*file handle*); töötlev funktsioon on antud failis;
- 3) kasutades *inline*-funktsioone (jooksva keskväärtuse 5×5 aknaga võib näiteks *inline*-operaatorit kasutades arvutada järgmise käsu abil: `Iout = nlfilter(Iin, [5 5], 'sum(sum(x))/25')`).

`nlfilter` võimaldab näiteks arvutada kujutise keskväärtust (*average*), kujutist diferentseerida, detekteerida kujutisel esinevaid piirjooni jne. Kui

`imfilter` käsu puhul väljundkujutise piksli väärtus arvutatakse alati aknasse jäävate sisendkujutise pikslite väärtuste lineaarkombinatsioonina, siis `nlfilter` võimaldab mitmekülgsemat operaatorite valikut.

Käsu `imfilter` puhul määratakse filter kahedimensioonilise kerneli k abil, mis kujutab endast aknasse jäävate sisendkujutise pikslite kordajate maatriksit. Filtreerimiskäsk näeb lihtsaimal kujul välja järgmiselt: $I_{out} = imfilter(I_{in}, k)$. `imfilter`-käsust ja kerneli määramisest tuleb juttu veel lõigus 6.3.

Siiamaani oleme kujutise filtreerimisega seoses käsitlenud erinevaid mooduseid filtreerimise realiseerimiseks, pööramata tähelepanu filtrite disainimisele. Kuigi ka kahedimensioonilisi filtreid on võimalik disainida sageduse tasandil etteantud kriteeriumite põhjal, kasutatakse kujutise puhul enamasti väljakujunenud kerneleid. Kõige kasutatavamate filtrite hulka kuuluvad näiteks järgmised filtrid:

- `'gaussian'`: selle filtri puhul saadakse kernel kahedimensioonilise Gaussi jaotuse põhjal. Filter kujutab endast madalpääsfiltrit ja selle parameeter σ määrab, kui järsult kerneli kordajate väärtused vähenevad, liikudes eemale filtri akna keskkohast.
- `'disk'`: ringikujuline keskmistav filter.
- `'sobel'`: Sobeli filtrit kasutatakse kujutisel esinevate objektide piirjoonte (servade) esiletoomiseks.
- `'prewitt'`: samuti filter objektide piirjoonte esiletoomiseks.
- `'laplacian'`: antud filter vastab kujutise teise tuletise võtmisele.
- `'log'`: kerneli kordajad saadakse Laplace'i operaatori rakendamisel Gaussi kernelile. Selle kerneliga filtreerides võetakse kujutisest tuletis.
- `'average'`: nelinurkse aknaga keskmistav filter. Selle filtri kerneli kõik kordajad on võrdsed.
- `'unsharp'`: filter, mis muudab kujutise teravamaks, rõhutades pikslitevahelist kontrastsust.

Mainitud kernelite loomiseks on MatLabis käsk `fspecial`, mille sisendparameetriteks on kerneli tüüp ning võimalikud kerneli parameetrid.

Kujutist võib töödelda ka blokikaupa. Sel juhul peab töötleva funktsiooni väljundiks olema kogu blokita is pikslite väärtusi. Käsk, mis võimaldab kujutist töödelda blokikaupa, on `blkproc`. Näiteks blokikaupa keskmistades

asendatakse väljundkujutise kõik filtri aknasse jäävad pikslid sisendkujutise vastavate pikslite keskväärtusega. Seejärel tõstetakse akent edasi terve akna ulatuse võrra.

Kujutise filtreerimisest on mitmeid näiteid kursuse harjutuste materjalis. Täpsemat informatsiooni vastavate MatLabi käskude kasutamise ning kernelite disainimise kohta on võimalik saada MatLabi keskkonna dokumentatsioonist.

6.3 Kahedimensiooniline Fourier' teisendus ja konvolutsioon

Kujutisi ning nende töötusalgoritme on võimalik vaadelda sageduse tasandil. On oluline märkida, et filtrite analüüsil sageduse tasandil on mõte vaid lineaarsete filtrite puhul, s.t kui väljund arvutatakse kerneliga antud kordajate ja sisendkujutise pikslite lineaarkombinatsioonina. Näiteks kõiki `imfilter` käsu abil realiseeritavaid filtreid on võimalik analüüsida sageduse tasandil. Samas võimaldab käsk `nfilter` realiseerida ebalineaarseid operaatoreid, mille analüüsil sageduse tasandil ei ole mõtet.

Kujutise teisendamine sageduse tasandile toimub kahedimensioonilise Fourier' teisenduse abil:

$$F(\omega_1, \omega_2) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f[m, n] e^{-j(\omega_1 m + \omega_2 n)} \quad (6.1)$$

Vastav pöördteisendus on

$$f[m, n] = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} F(\omega_1, \omega_2) e^{j(\omega_1 m + \omega_2 n)} d\omega_1 d\omega_2 \quad (6.2)$$

Esitatud teisenduse puhul on sagedusesitus $F(\omega_1, \omega_2)$ pidev. Et saada diskreetset sagedusesitust, kasutatakse diskreetse Fourier' teisenduse valemid:

$$\begin{cases} F[k, l] &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f[m, n] e^{-j2\pi(\frac{km}{M} + \frac{ln}{N})} \\ f(m, n) &= \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} F[k, l] e^{j2\pi(\frac{km}{M} + \frac{ln}{N})} \end{cases} \quad (6.3)$$

Kahedimensioonilist diskreetset Fourier' teisendust on MatLabis võimalik arvutada käsu `fft2` abil.

Nii nagu ühedimensiooniliste signaalide puhul, vastab ka kujutise filtreerimisele lineaarse filtriga kujutise konvolutsioon filtri impulsskajaga. Impulsskaja on antud juhul kahedimensiooniline ($h[m, n]$). 2-D konvolutsioonil on järgmine kuju:

$$y[m, n] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[m, n] h[m - k, n - l] \quad (6.4)$$

Nagu valemist 6.3 on näha, seisneb kahedimensioonilise konvolutsiooni arvutamine sisendkujutise pikslite väärtuste korrutamises mõlema telje suhtes ümberpööratud impulsskajaga ja summeerimises üle mõlema dimensiooni. Sealjuures nihutatakse impulsskaja piksel-pikslilt edasi üle kogu kujutise.

Lõigus 6.2 vaatlesime kujutise filtreerimist kui kujutise ja kerneli väärtuste lineaarkombinatsiooni arvutamist. Kernel ongi filtreerimise seisukohalt 2-D impulsskaja mõlema telje suhtes ümberpööratud versioon. Oletades, et kernel/impulsskaja sisaldab $0 \dots M - 1$ rida ja $0 \dots N - 1$ veergu, võime kerneli impulsskaja kaudu avaldada järgmiselt:

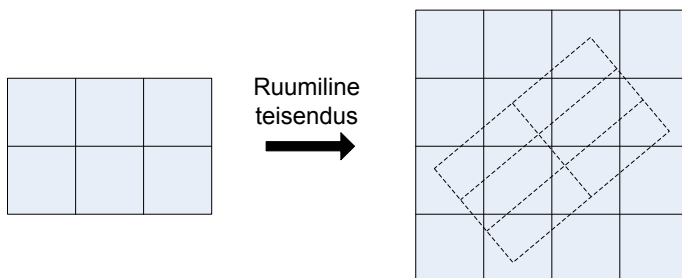
$$k[m, n] = h[M - m - 1, N - n - 1] \quad (6.5)$$

Nagu varem mainisime, kasutatakse kujutise töötleses enamasti kerneleid/impulsskajasid, mille ridade ja veergude arv on paaritu. Samuti on suurem osa kõige kasutatavamatest kernelitest kummagi telje suhtes sümmeetrilised, mille puhul pole vahet, kas filtreerimisel kasutada kernelit või impulsskaja. Erandi moodustavad filtrid, mille otstarve on rõhutada või eemaldada kujutisel teatud suunas olevaid objekte või mustreid.

MatLab'i käsk `imfilter` tõlgendab etteantud filtri kordajate maatriksit vaiki-misi kernelina. Kui soovitakse kordajad ette anda impulsskaja kujul, tuleb kasutada lisaparaametrit `'conv'`. Samuti on võimalik kahedimensioonilist konvolutsiooni arvutada käsu `conv2` abil.

6.4 Kujutise ruumilised teisendused

Kujutise ruumiline teisendus kujutab endast kujutise pikslite mäpingut etteantud algoritmi järgi. Ruumilise teisenduse tulemus võib näiteks olla alge- se kujutise teatud suunas venitatud ja/või pööratud versioon (vt joonis



Joonis 6.1 Kujutise ruumilise teisenduse põhimõte

6.1). Uue kujutise pikslitele on sealjuures vaja leida väärtused algse kujutise pikslite põhjal. See operatsioon sisaldab üldjuhul interpoleerimist.

Kujutise interpoleerimisel kasutatakse põhiliselt kolme strateegiat:

- *nearest neighbor*: uue kujutise piksli väärtuseks võetakse teisendatud algse kujutise selle piksli väärtus, mis paikneb kõige lähemal uue kujutise antud pikslile;
- *bilinear interpolation*: uue kujutise piksli väärtus arvutatakse teisendatud algse kujutise 2×2 akna pikslite kaalutud summana. Valitakse selline akna positsioon, mis hõlmab arvutatava piksli lähimat naabrust;
- *bicubic interpolation*: sarnane eelmisega, kasutatakse 4×4 akent.

6.4.1 Afiinsed teisendused

Ruumilist teisendust, mille puhul sirged jäävad sirgeteks ning sirgete paralleelsus säilib, nimetatakse afiinseks teisenduseks. Afiinne teisendus pöörab kujutist ning kas venitab või surub seda kokku horisontaal- ja/või vertikaalsuunas.

Afiinseks teisenduseks võib kasutada järgmisi käske:

- *imresize*: selle käsuga on võimalik muuta kujutise mõõtmeid, seega hõlmab see käsk vaid afiinsete teisenduste kitsast erandjuhtu. Käsu *imresize* sisendparameetriteks on uue kujutise mõõtmed ($[M N]$) ning interpolatsiooni strateegia. Kui kujutise mõõtmeid vähendatakse, teostab *imresize* automaatselt madalpääsfiltrereerimise ülekatte

vältimiseks sageduse tasandil (välja arvatud juhul, kui interpolatsiooni strateegia on *nearest neighbor*, mida kasutatakse põhiliselt indekseeritud formaadi puhul).

- `imcrop`: selle käsuga on võimalik kujutisest "välja lõigata" nelinurkseid piirkondi. Kasutaja määrab soovitud piirkonna hiirega. Ka sellist operatsiooni on võimalik vaadelda kui ruumilise teisenduse erandjuhtu.
- `imrotate`: võimaldab kujutist pöörata. Antud käsu sisendparameetriteks on pööramisnurk ja interpolatsioonistrateegia.
- Laia hulka ruumilisi teisendusi on võimalik teostada `imtransform`-käsu abil. See käsk kasutab teisenduse määramiseks `Tform`-formaati. Selle formaadi võib luua `maketform`-käsuga, mille parameetriteks on teisenduse tüüp, sisendkujutise geomeetria ning vastav geomeetria teisendatud kujutisel. Afiinne teisendus on defineeritud kolme punkti abil, järelikult piisab, kui algse ja teisendatud kujutise geomeetriad on asetatud vastavusse kolmes punktis. Lisaks teisenduse formaadile `Tform` võib käsule `imtransform` anda lisaparameetreid, mille abil on võimalik määrata teisendatav ala algsel kujutisel, teisendatud kujutise mõõtmed, muuta piksli mõõtmeid teisenduse käigus ning määrata, kuidas täidetakse teisenduse tulemusel tühjaks jäävad pikslid.

Näide 6.1 Kujutise afiinne teisendus `imtransform`-käsu abil

Nagu eespool mainisime, on kõigepealt vaja defineerida teisendus, kasutades `Tform`-formaati. Seda saab teha `maketform`-käsuga. Viimasele on võimalik ette anda sisendparameetreid mitmel eri moel. Selles näites vaatleme teisenduse defineerimist piisava arvu vastavate punktide määramise teel sisend- ja väljundkujutisel. Afiinse teisenduse jaoks piisav punktide arv on 3.

Määrame teisenduse nii, et kolmnurk, mille kaks nurka on sisendkujutise alumistes nurkades ja kolmas nurk sisendkujutise ülemise serva keskel, teisenduks kolmnurgaks, mille ülemine nurk on nihkunud kujutise paremasse ülemisse nurka. Kolmnurga alumised nurgad püsivad teisenduses paigal. Eeldades, et kujutise ridade arv on M ja veergude arv N ning võttes arvesse, et MatLabis paikneb piksel $[1, 1]$ kujutise ülemises vasakus nurgas, on eespool kirjeldatud kolmnurga nurgad sisendkujutisel punktides

```
>> U = [N/2 1; 1 M; N M];
```

ja teisendatud kujutisel punktides

```
>> X = [N 1; 1 M; N M];
```

Afiinse teisenduse võime nüüd defineerida järgmiselt:

a)



b)



Joonis 6.2 Sisendkujutis (a) ja selle näites 6.1 defineeritud afiinne teisendus (b)

```
>> Tform = maketform('affine', U, X);
```

ja teisenduse teostada järgmise käsu abil:

```
>> Iaffine = imtransform(Iin, Tform);
```

Joonisel 6.2 on esitatud nii originaalne kujutis kui ka eespool määratud teisenduse tulemusel saadud kujutis. Nagu näha, jäävad afiinne teisenduse puhul sirged sirgeteks ja paralleelsus säilib.



Joonis 6.3 Joonisel 6.2 a toodud kujutise projektiivne teisendus

6.4.2 Projektiivsed teisendused

Kujutise projektiivse teisenduse puhul jäävad sirged sirgeteks, kuid paralleelsus ei säili. Projektiivseid teisendusi võib teostada samuti *imtransform*-käsu abil. Teisenduse defineerimiseks on nüüd vaja nelja punkti.

Näide 6.2 *Kujutise projektiivne teisendus imtransform-käsu abil*

Projektiivse teisenduse puhul on vaja seada vastavusse neli punkti originaalsel ja teisendatud kujutisel. Analoogselt näitega 6.1 määrame

```
>> U = [1 1; N 1; 1 M; N M];  
>> X = [N/5 1; 4*N/5 1; 1 M; N M];  
>> Tform = maketform('projective', U, X);
```

Projektiivse teisenduse teostame järgmiselt:

```
>> Iprojective = imtransform(Iin, Tform);
```

Joonisel 6.3 on esitatud joonisel 6.2 a toodud originaalkujutise projektiivne teisendus eespool defineeritud punktide järgi.

imtransform-käsk võimaldab lisaks aafiinsele ja projektiivsele teisendusele veel mitmete keerulisemate teisenduste realiseerimist. Nende kohta, nii nagu kõikide käesolevas lõigus mainitud käskude kohta, on võimalik lisainformatsiooni saada MatLab'i keskkonna dokumentatsioonist.

6.5 Kujutise registreerimine

Kujutise registreerimise all mõistetakse kujutise sobitamist referentskujutisega. See on tüüpiline ülesanne meditsiiniliste kujutiste puhul, kui eri aegadel või eri meetodite abil salvestatud kujutised on vaja asetada vastavusse.

Kujutisi võib MatLabi abil registreerida kahel moel:

- 1) Automaatne kujutise registreerimine. Selline kujutise registreerimine põhineb iteratiivsel kujutise teisendamisel ja korrelatsiooni arvutamisel referents- ja teisendatud kujutiste vahel. Eesmärgiks on leida teisendus, mille puhul korrelatsioon on maksimaalne. Nii nagu tihti iteratiivsete maksimeerimismeetodite puhul, on ka siin oht lõpetada otsing lokaalses maksimumis. Automaatse kujutise registreerimise jaoks on MatLabis käsk `fminsearch`, mille argumentideks on kasutatav teisendus (näiteks `rescale`) ning teisenduse algparameetrid, väljundiks optimaalne teisendus ning korrelatsiooni väärtus. On oluline märkida, et `fminsearch` ei liitu tingimata kujutise registreerimisega, vaid seda võib kasutada mis tahes funktsiooni miinimumi või maksimumi leidmiseks.
- 2) Interaktiivne kujutise registreerimine. Sel puhul määrab kasutaja interaktiivselt hulga vastavaid punkte referentskujutisel ja registreeritava kujutisel. Käsu `cp2tform` abil on võimalik moodustada teisendusformaati `Tform` interaktiivselt määratud punktide alusel.

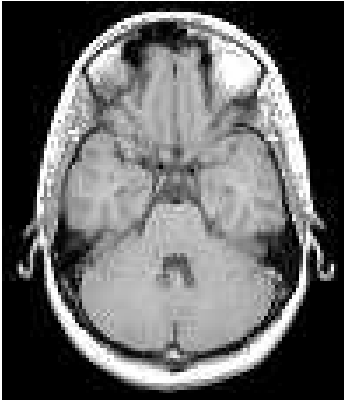
Näide 6.3 Kujutise registreerimine

Järgnevalt vaatleme kujutise registreerimist interaktiivselt MatLabi keskkonna vahendite abil. Joonisel 6.4 on esitatud referentskujutis ja registreeritav kujutis.andes MatLabi keskkonnas käsu

```
>> cpselect(I_reg, I_ref);
```

avaneb joonisel 6.5 esitatud töövahend. Alumistes akendes on kujutised, mille vahel soovitakse määrata vastavaid punkte. Ülemistes akendes on suurendatult mõlema kujutise valitud piirkonnad. Vastavate punktide määramine toimub, valides hiirega kordamööda kummagi kujutise vastavaid punkte. Punktide salvestatakse vektoritesse `input_points` (registreeritava kujutise jaoks) ja `base_points` (referentskujutise jaoks). Järgnevalt on võimalik käsu

```
>> Tform = cp2tform(input_points, base_points, 'projective');
```



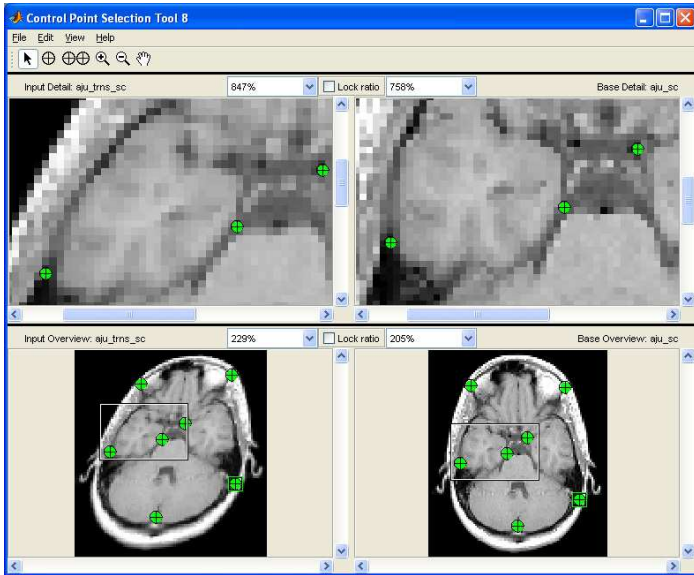
Joonis 6.4 Kujutise registreerimine: referentskujutis (vasakul) ja registreeritav kujutis (paremal)

abil leida ruumiline teisendus, mis registreerib registreeritava kujutise referentskujutisele. Teisenduse tüüp valitakse selle järgi, kuidas kujutised on oletatavasti omavahel seotud. Registreerimine toimub *imtransform*-käsu abil:

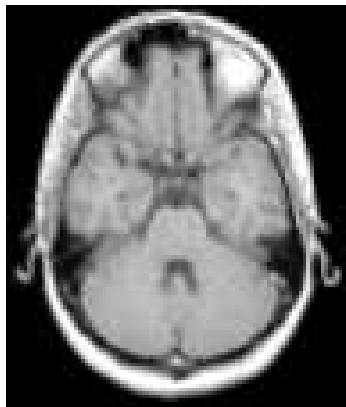
```
>> I_out = imtransform(I_reg, Tform);
```

Joonisel 6.6 on esitatud joonisel 6.5 näidatud punktide järgi registreerimise tulemus, kui teisenduse tüübiks valiti *'projective'*.

Tuleb veel märkida, et toodud näide ei hõlma sugugi kõiki registreerimisprotseduuri aspekte. Lähemalt on võimalik tutvuda mainitud käskude kasutamise võimalustega MatLabi keskkonna dokumentatsiooni abil.



Joonis 6.5 Kujutise registreerimine: MatLabi keskkonna vahend vastavuspunktide määramiseks graafiliselt



Joonis 6.6 Kujutise registreerimine: registreeritud kujutis