

Elektroakustika

osa6:

Helisüntees

Helisüntees lainetabelite abil

Sämplimine

Summeeriv süntees

Vähendav süntees

FM-süntees

Pillide mudelid

Digitaalse helisünteesi eesmärk

Digitaalse helisünteesi eesmärk võib olla:

- mingi akustilise instrumendi võimalikult täpne 'matkimine', kusjuures mitmetest akustilise instrumendi 'piirangutest' on võimalik loobuda
- genereerida helisid, mida akustiliste instrumentidega pole võimalik toota, nö. laiendada 'helide maailma'

Max Mathews (1963) *'Sound from numbers ... Any perceivable sound can be produced'*

(Digitaalse) helisünteesi puhul muusikaline heli toodetakse nö. 'tühjast'. Vaja on üldjuhul:

- lähtesignaali (näiteks müra, elementaarne perioodiline signaal jne; lähtesignaali üldjuhul töödeldakse heli sünteesimisel)
- kontrollparameetreid (kuidas genereeritud signaali maha mängida, näiteks amplituudi muutumine, ADSR)

Helisünteesi kasutusvaldkonnad

- Muusikas
 - naturaalsed ja sünteetilised pillid
 - sünteetiline lauluhääli
- Multimeedia-seadmed
- Arvutimängud, virtuaaltegelikkus, filmide heli
 - muusikalise heli muutumine vastavalt situatsioonile (näiteks liikumine ühest ruumist teise)
 - heliefektid (näiteks relvade lasud, kokkupõrked)
 - taustahelid (näiteks sadu, tuul, metsa kohin)
- Testsignaalid
 - akustilistel mõõtmistel kasutatavad helisignaalid
 - psühhoakustiline ja kognitiivne testimine

Helisünteesi meetodite klassikaline liigitamine

- Lineaarsed meetodid (sünteesitud heli koosneb samadest sagedustest kui lähtesignaal; lineaarsete meetodite puhul kehtivad homogeensus ja superpositsioon):
 - lainetabel
 - sãmplimine
 - summeeriv süntees
 - vähendav süntees (lähtesignaal -> DSP)
 - raesüntees (*granular synthesis*)
- Ebalineaarsed meetodid (üldjuhul tegemist modulatsiooniga, signaali tekivad uued sageduskomponendid)
 - FM (*frequency modulation* e. sagedusmodulatsioon) – süntees
 - lainekuju vormimine (*waveshaping*)
 - ringmodulatsioon
 - faasimodulatsioon

Lainetabeli-süntees

Lainetabeli-süntees (*wavetable synthesis*) oli esimene helisünteesi meetod (Max Mathews 1950-ndate lõpp)

Genereeritakse (näiteks sinusoidide summa, kolmnurkne perioodiline signaal vms.) või salvestatakse üks periood helisignaali, mida seejärel korratakse. Kui perioodis on N diskreeti, heli põhisageduseks saame:

$$f = \frac{F_s}{N}$$

Erineva põhisageduse saamiseks peame mahamängimisel edenema tabelis eri sammuga. Tabelis edenemise samm sõltub soovitud põhisagedusest:

$$\text{samm} = \frac{N \times f}{F_s}$$

Kui samm ei ole täisarv, on vaja tabelis olevate väärtuste vahel interpoleerida.

Näiteid primitiivsest lainetabeli-sünteesist:

- lainetabelis üks periood sinusoidi
- lainetabelis poolteist perioodi sinusoidi
- lainetabelis müra

Amplituudi mähisjoon lainetabeli-sünteesil

Mängitud noodi amplituud ei ole enamasti konstantne vaid tema mähisjoon koosneb tüüpiliselt järgmistest etappidest:

- *attack*
- *decay*
- *sustain*
- *release*



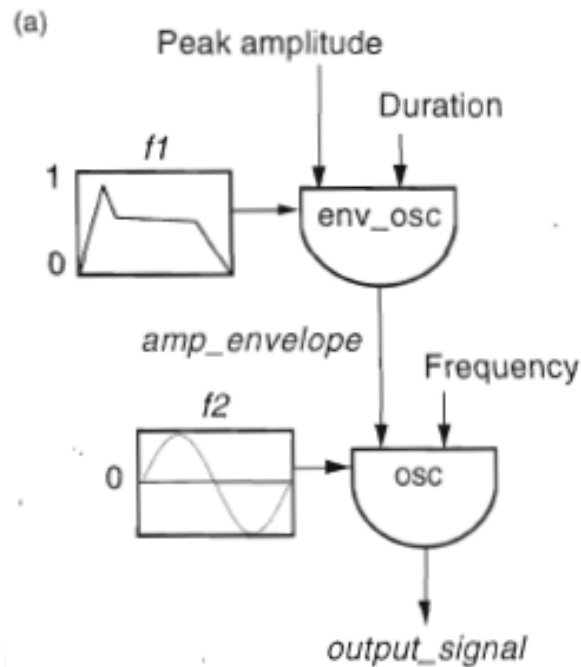
Sellist mähisjoont tähistatakse lühendiga ADSR

Komplitseeritumate süntesaatorite puhul mähisjoone vormimine on keeruline protsess.

Lainetabeli-süntees

Lainetabeli sünteesi puhul ühe heli sünteesimisel määratakse ära järgmised parameetrid:

- maksimaalne amplituud
- heli kestus
- mähisjoone kuju
- heli sagedus (sellest arvutatakse samm)
- laine kuju

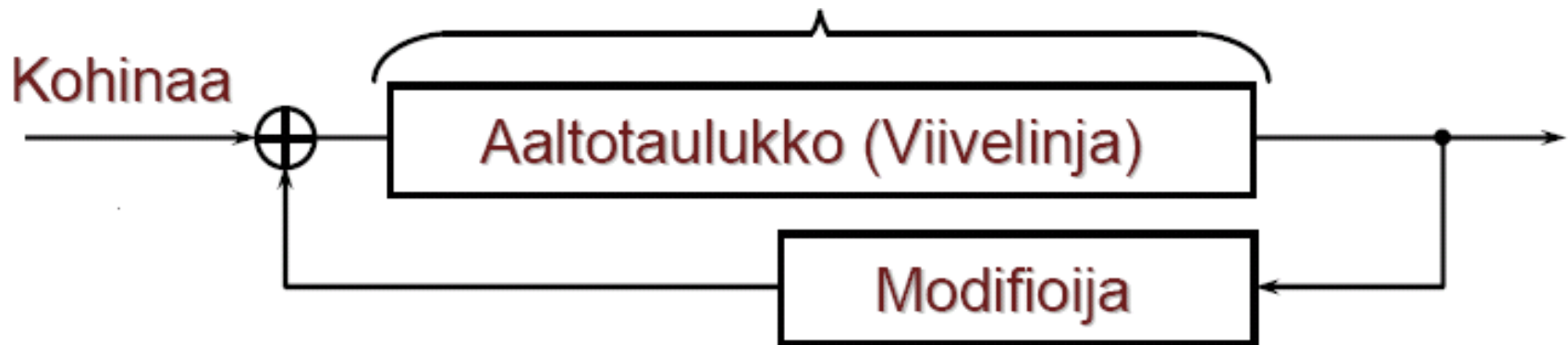


Karplus-Strong algoritm

Karplus ja Strong leiutasid 1970-ndate lõpul helisünteesi meetodi kus lainetabeli asemel kasutatakse hilistusahelat. Hilistusahelas olevat signaalilõiku muudetakse jätkuvalt, mis annab helile loomulikuma kõla.

Üks moodus muuta hilistusahelas olevat lõiku on arvutada jätkuvalt kahe kõrvuti oleva diskreedi keskmist.

Heli sageduse määrab ära hilistusahela pikkus



Sämplimine

Sämplimise all mõeldakse helisünteesi puhul lainetabeli-sünteesile sarnast meetodit, mille puhul lainetabelis on salvestatud (akustiliste pillide) signaalilõigud. Tavaliselt sämplimise puhul tabelis olevad lõigud on oluliselt pikemad kui üks periood. Kui mälu maht ei ole kriitiline, salvestatakse iga noot eraldi.

Sämplitud lainetabeli 'rida' on võimalik enne mahamängimist töödelda (näiteks filtreerida, aeglustada, kiirendada).

Sämplimise abil on võimalik esitada nö. hübriidpille, mille heli algab näiteks klaverina ja lõpeb viulina.



Sämplimine ja ADSR

Sämpler tavaliselt loeb lainetabelis kõigepealt heli *attack+decay* osa, seejärel kordab *sustain*-osa nii kaua et heli kestus on soovitud pikkusega ning lõpuks loeb tabelist *release*-osa.

Korratava osa algus- ja lõpp-punkti määratlemiseks leitakse tihti kõigepealt heli põhisagedus. Tsükli moodustamisel on siiski oht et tsükli vahekohad on kuuldavad. Selle vältimiseks on võimalik

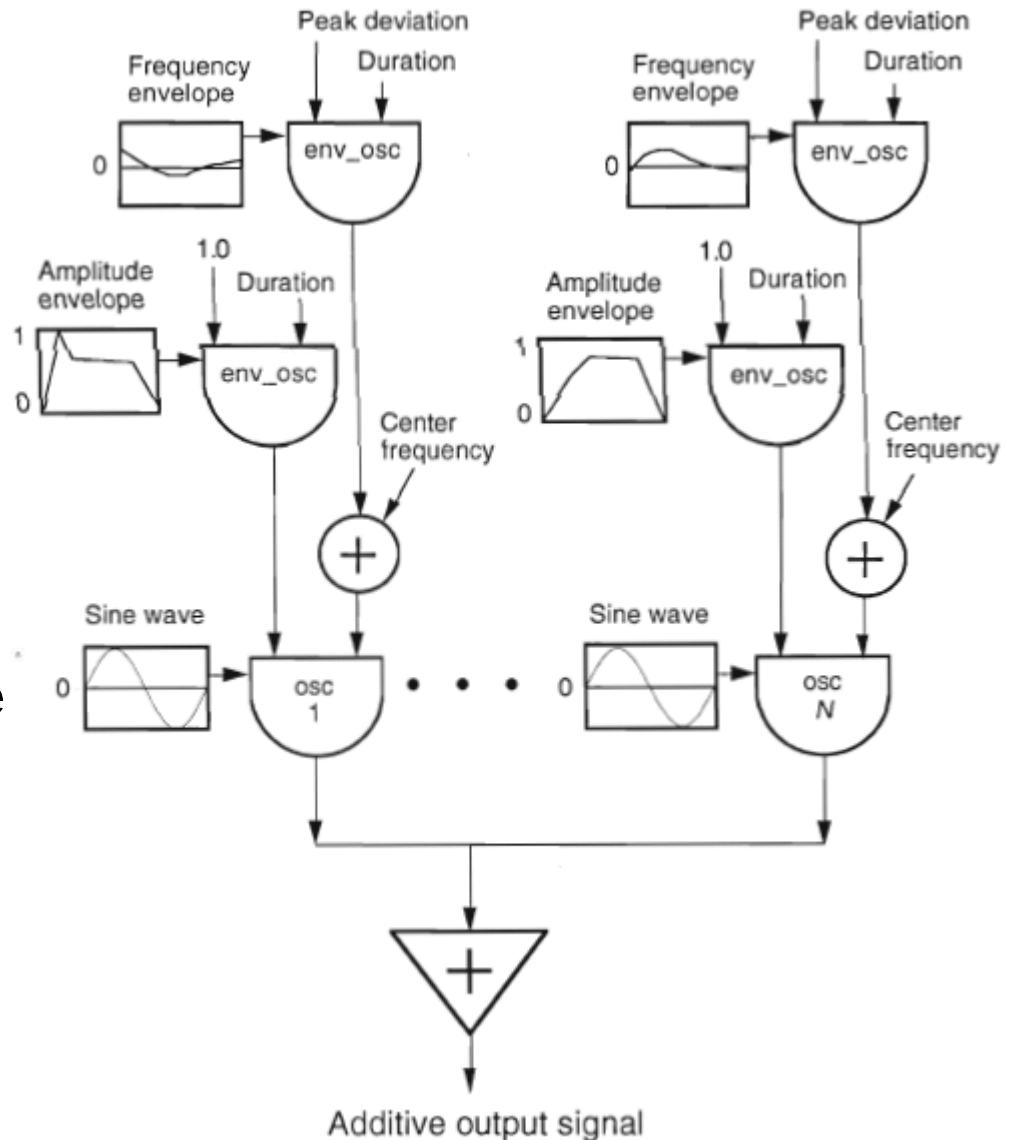
- laine kuju järgi leida koht kus tsüklit oleks kõige sujuvamalt võimalik moodustada (NB! üksikud perioodid ei ole identsed!)
- kasutada nn. *crossfading*-tehnikat
- korrata löiku vaheldumisi eri suundades
- arvutada edaspidi ja tagurpidi loetud löikude keskmine ning korrata seda keskmistatud löiku.

Summeeriv süntees

Summeeriva sünteesi puhul heli toodetakse summeerides sinusoide (iga sagedus-komponent e. harmooniline genereeritakse eraldi).

Sagedus võib noodi kestel muutuda (*Frequency envelope*)

Summeeriv süntees on vastupidine operatsioon võrreldes Fourier' analüüsiga.



Summeeriv süntees: Kawai K5000W


Kawai K5000W (1996-1998): [Kawai K5000 Additive synthesis \(YouTube\)](#)


32 Sources

64 harmonics per Source (#1-64 or #65-128)

5-stage, looping amplitude envelope for each harmonic

123 specially sampled PCM waveforms

Demo 1: 

Demo 2: 



K5000W
Advanced Additive Workstation

Summeeriva sünteesi puudused ja eelised

Põhimõtteliselt võimalik genereerida mistahes omadustega heli. Heli omadusi on võimalik manipuleerida väga detailselt.

Probleemiks on kontrollandmete suur hulk (kõigi sageduskomponentide mähisjoon, sageduse muutus ajas, faas jne.)

Ei sobi kuigi hästi ei-harmonilise (müralaadse) heli sünteesimiseks (spekter on nõ. tihedalt täis komponente)

Modelleerimine sinusoidide ja müra summaga

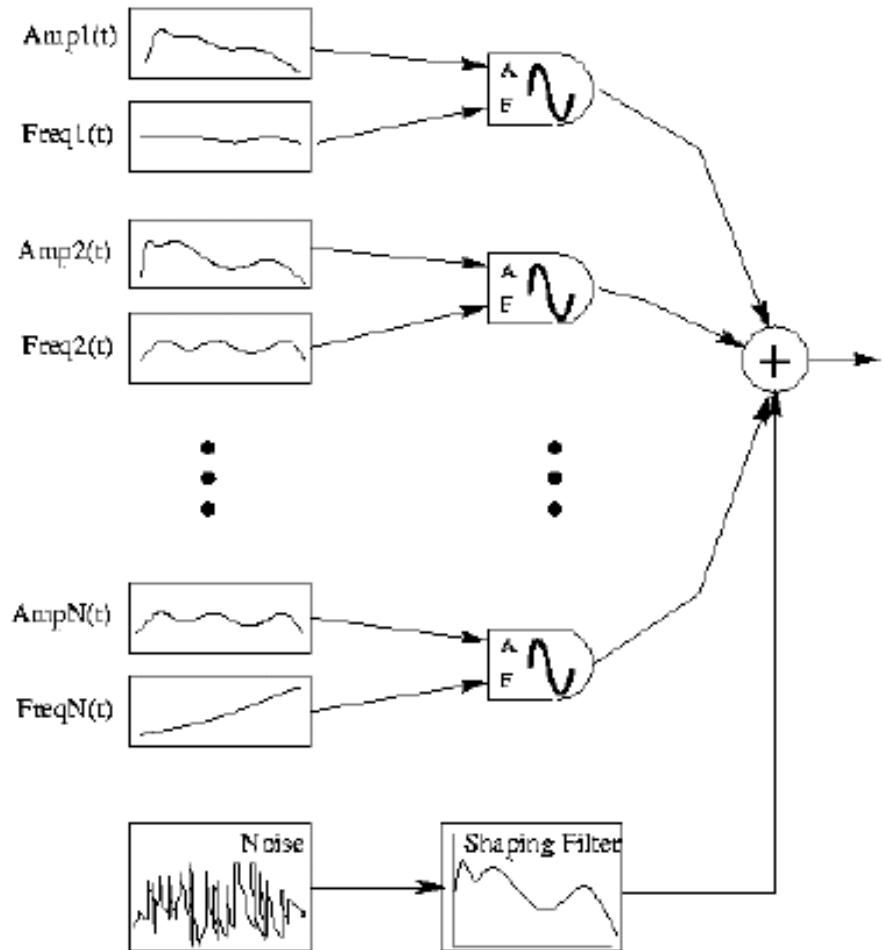
Helisignaali on võimalik modelleerida (ja vastavalt mudeli järgi sünteesida) sinusoidide ja müra summana.

Helisignaali lahutatakse sinusoidideks ja müraks. Sinusoidid modelleeritakse nende määrisjoone ja hetkelise sageduse järgi.

Müra modelleeritakse enamasti filtriga, mis kujundab valge müra spektrit.

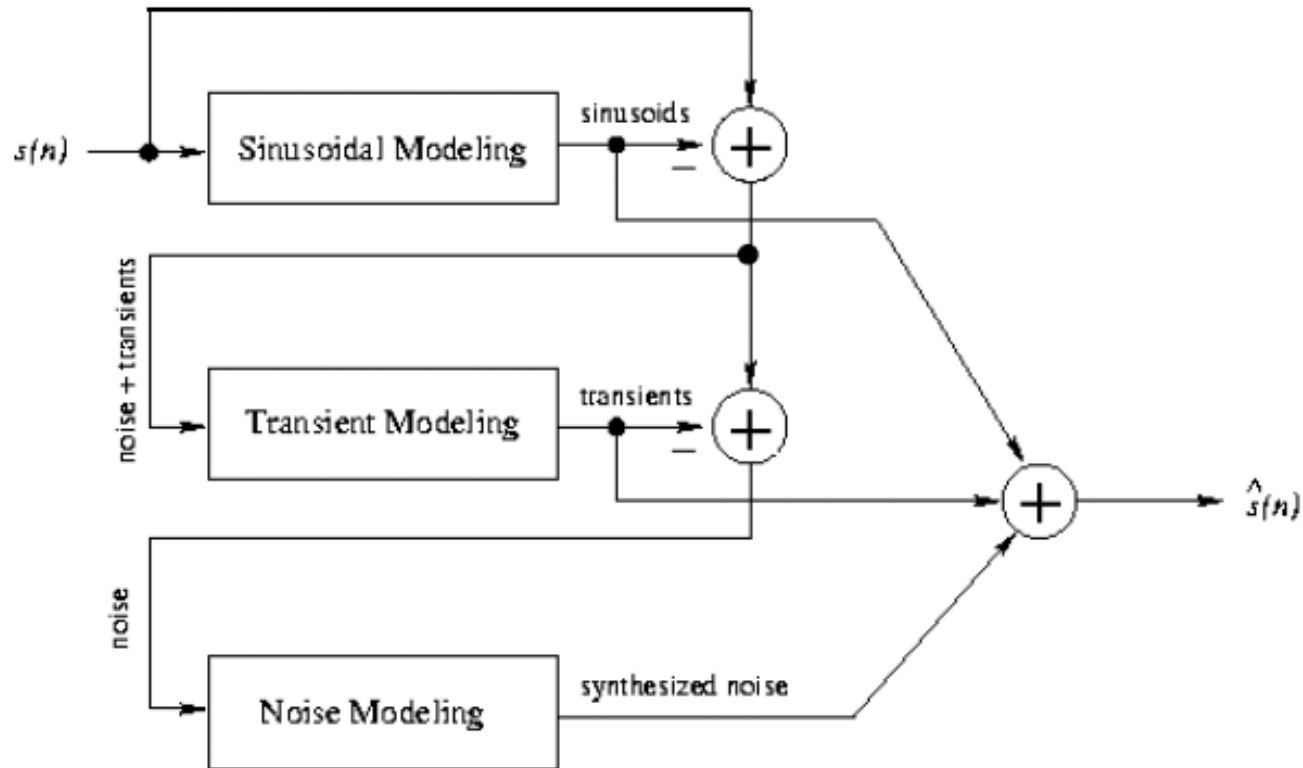
Näiteid:

[Sinusoid+müra: Tuomas Virtanen](#)



Lühiajalised komponendid (*transients*)

Lühiajaliste signaalikomponentide modelleerimiseks ei sobi sinusoidaalsed komponendid ega ka müra. Need on võimalik detekteerida ja modelleerida eraldi.



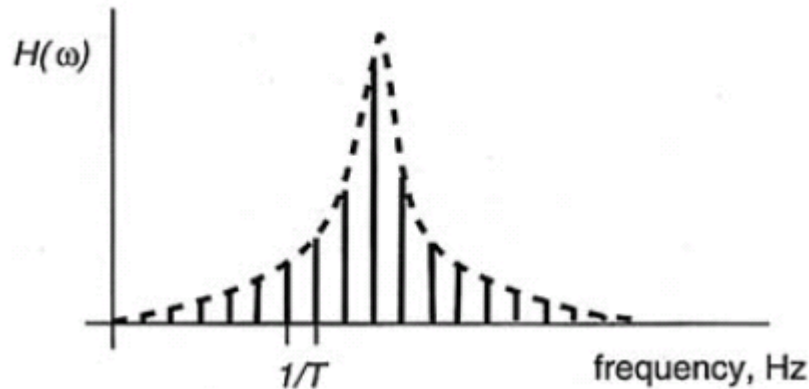
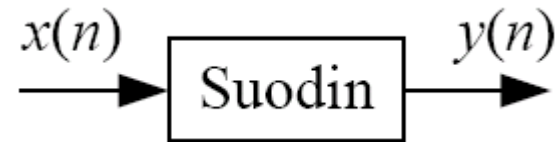
Vähendav süntees

Põhineb allikas -> filter põhimõttel:

genereeritakse mingi elementaarne signaal, mille omadusi muudetakse filtreerimise teel.

Enamasti kasutatakse elementaarseid signaale, mille spektris on tihedalt komponente.

Signaal filtreeritakse kas madalpääsfiltriga või resonaatoriga



Minimoog (1970-1981)

Vähendava sünteesi puudused ja eelised

Arvutuslikult oluliselt lihtsam moodus toota suurt hulka harmoonilisi võrreldes summeeriva sünteesiga

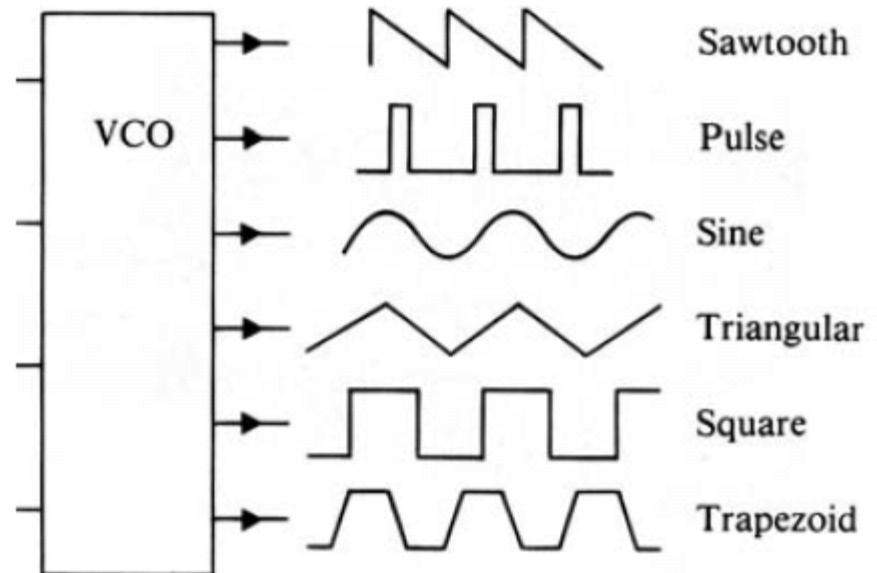
Spektri kuju on võimalik kontrollida filtri kordajate abil

Filter peab olema dünaamiline, ajas muutuva sageduskarakteristikuga (muidu on sünteesitud heli 'igav')

Mitteharmooniliste helide sünteesimine on keeruline

Üksikute harmooniliste kontrollimine õnnestub ainult väga kõrge järguga filtri puhul. See ei ole üldjuhul otstarbekas.

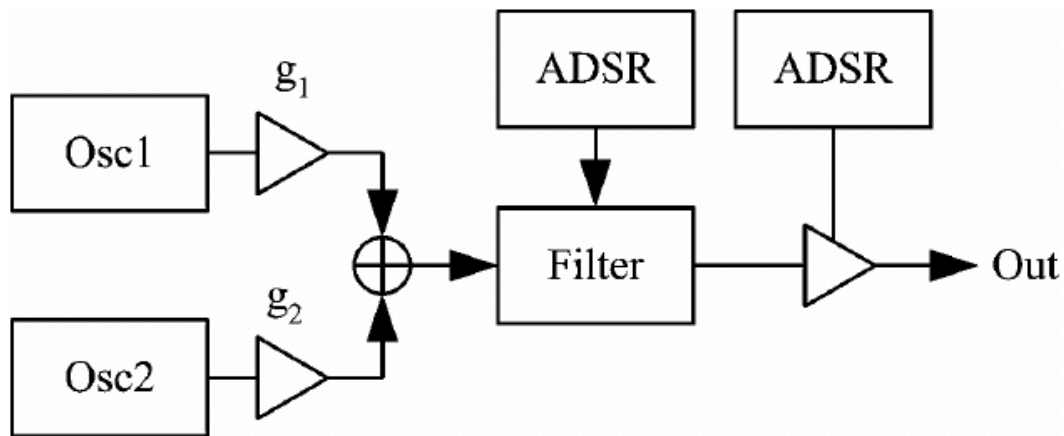
Lähtesignaali on üldjuhul perioodiline signaal:



Vähendava sünteesi implementeerimine

Ostsillaatorid näiteks oktaavis või natuke 'häälest ära'

Üks ADSR kontrollib filtri parameetreid ja teine signaali mähisjoont



Filtrina võib kasutada näiteks teist või neljandat järku resoneerivat madalpääsfiltrit

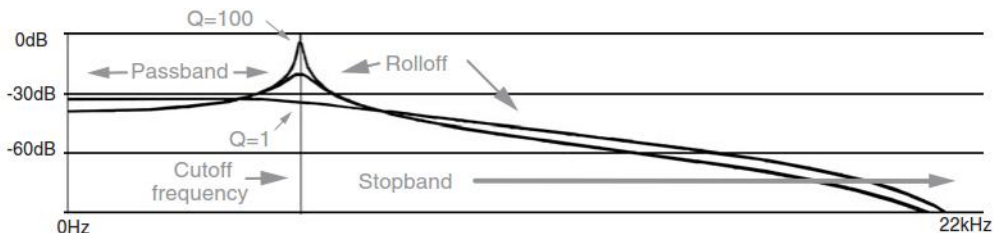


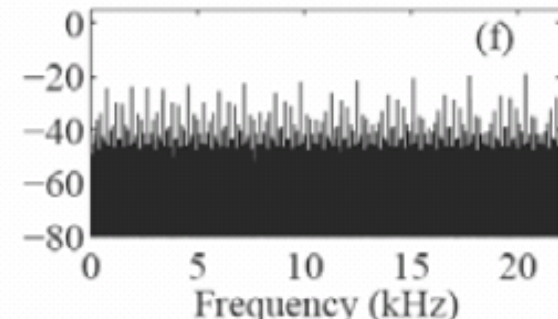
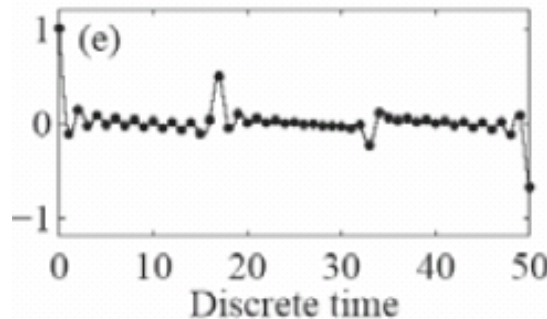
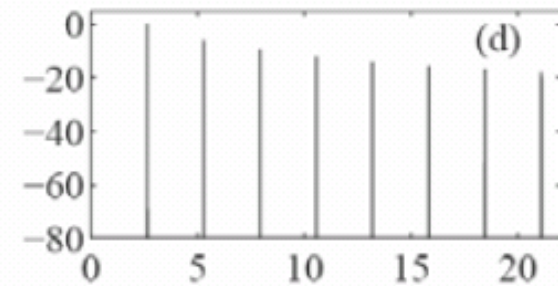
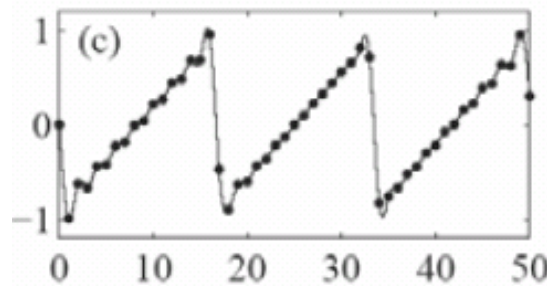
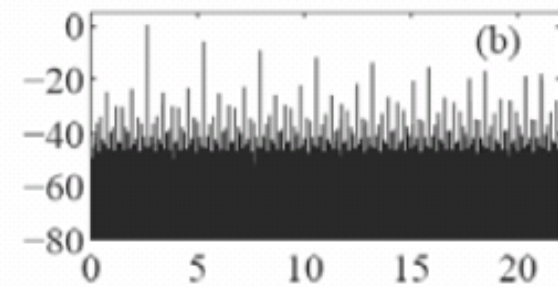
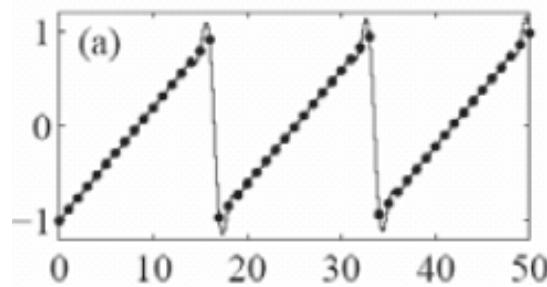
Figure 6.7 LPF low-pass filter frequency response for Q=1, Q=10, and Q=100

Lähtesignaali genereerimine digitaalselt

Digitaalselt genereeritud perioodiliste signaalide spektris esinevad nõ. alias-komponendid (joonise ülemine rida)

Summeeriva sünteesi abil on võimalik genereerida perioodilisi signaale, milles ei ole alias-komponente. Selline generaator on arvtuslikult keeruline (kaotame vähendava sünteesi eelise) ->

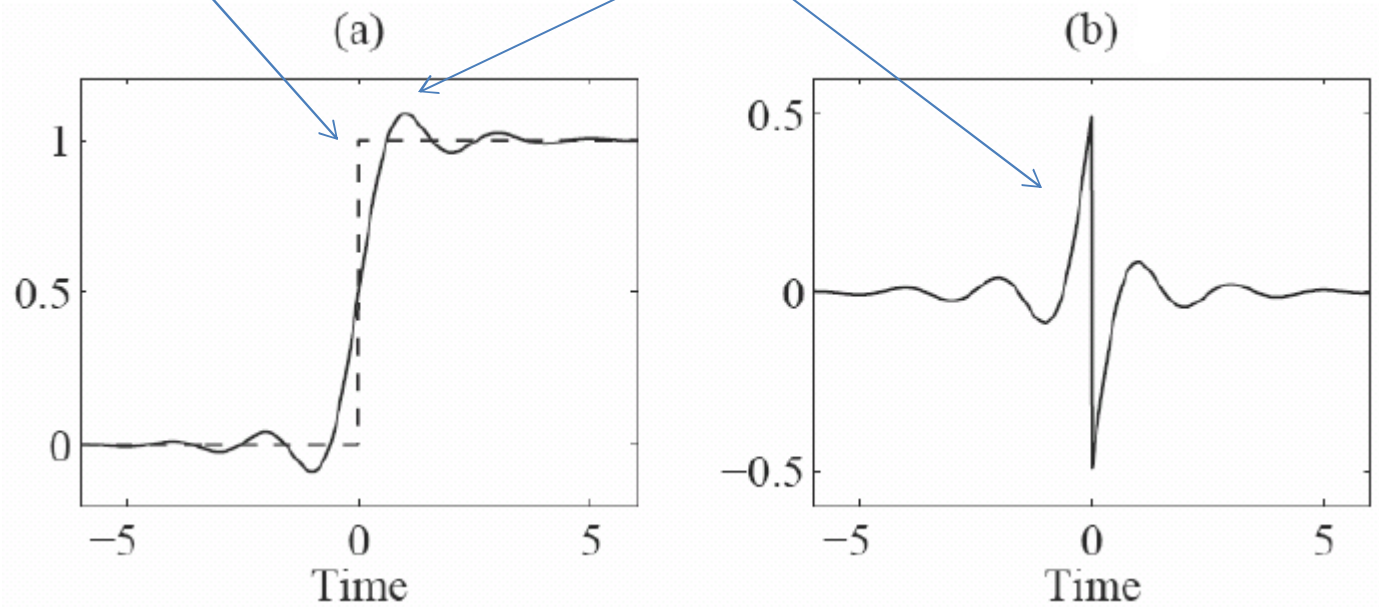
Kahe eelmise signaali vahe ->



BLEP meetod digitaalse lähtesignaali genereerimiseks

BLEP – BandLimited stEP function

Iga järsu hüppe kohale genereeritavas signaali lisatakse madalpääsfiltreeritud ja originaalse ühikhüppe vahe



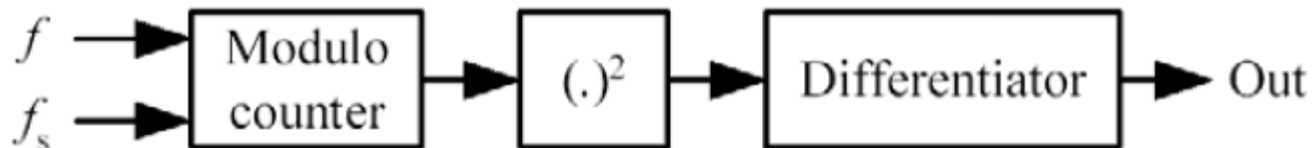
DPW meetod

DPW – Differentiated Parabolic Wave (Välimäki et. al.)

Genereeritud saehamba kujuline signaal võetakse teise astmesse; sealjuures alias-komponentide osakaal väheneb

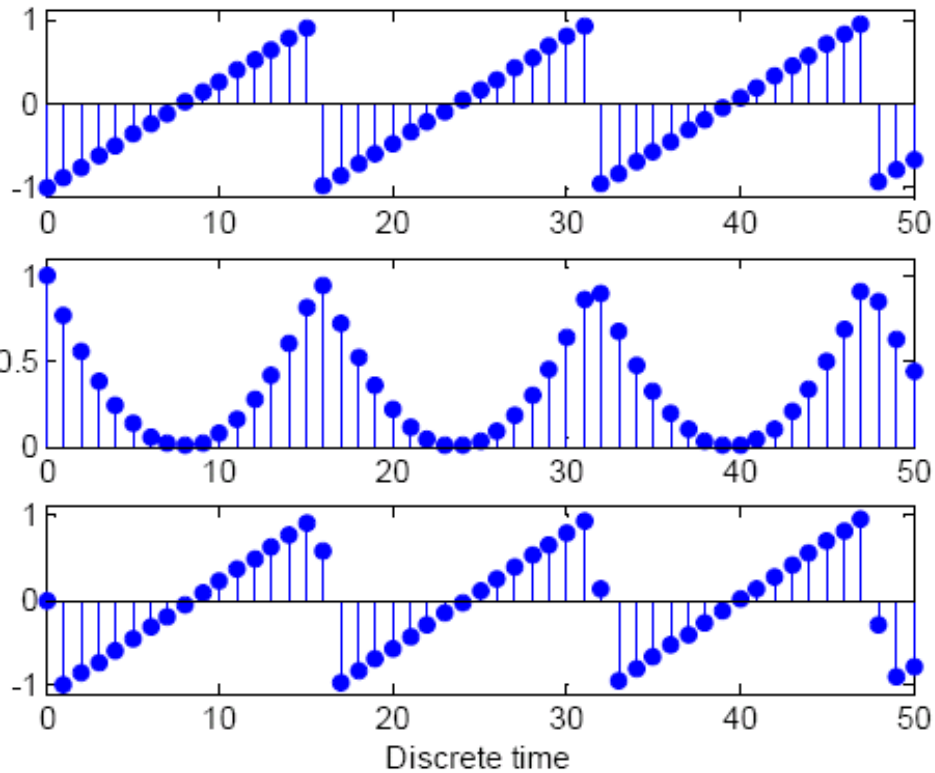
Seejärel differentseeritakse e. arvutatakse kõrvuti olevate diskreetide vahe

$$y = c[x^2(n) - x^2(n-1)] \quad c = f_s/4f$$

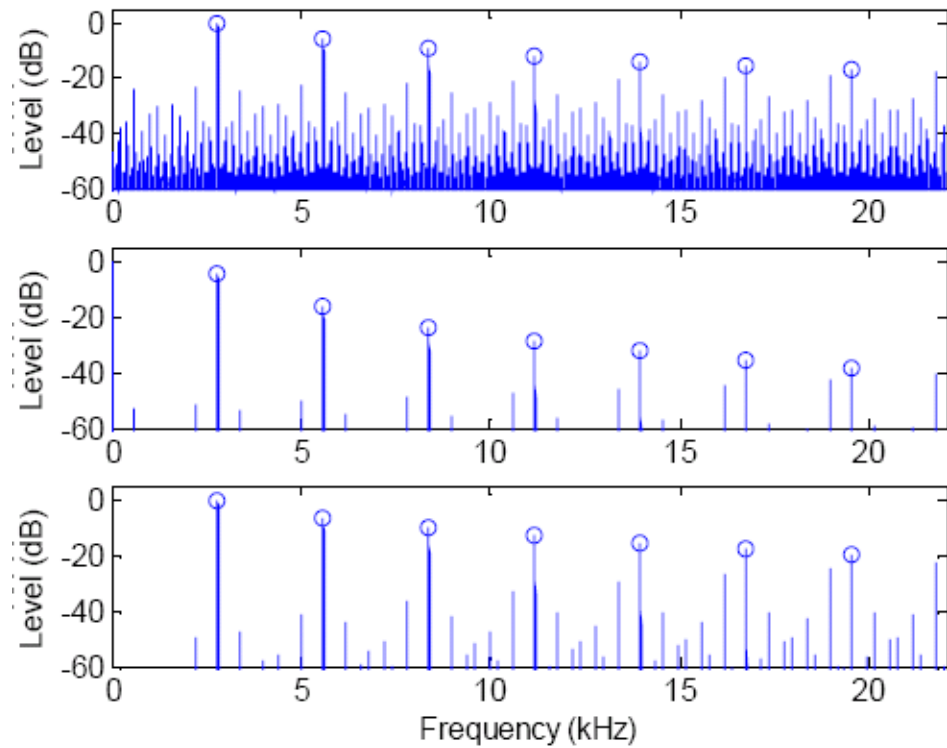


DPW meetod (2)

Aja tasandil:



Sageduse tasandil:



Digitaaalse ostsillaatori algoritme

Algorithm	Computational complexity	Memory consumption	Sound quality	Remarks
Trivial	Very low	No	Poor	Practically useless, except at low f
Lane	Average	Very small	Good	Requires a sine oscillator
DPW	Very low	Very small	Good	Simple and useful
DPW2X	Low	Very small	Very good	Slightly better than DPW
PolyBLEP	Low	Very small	Very good	A division per discontinuity
Wavetable	Average...high	Very large	Very good	Requires interpolation
BLIT-SWS	Average...high	Small	Very good	A division per discontinuity
BLEP	Average	Small	Excellent	A division per discontinuity
Additive	Very high	Very small	Excellent	Computationally intensive

Virtuaal-analoog süntees

Helisünteesi eelpool kirjeldatud meetodil (digitaalne ostsillaator alias-komponentide vähendamisega + digitaalne filter) kutsutakse virtuaal-analoog sünteesiks.

Esimesi tooteid, kus kasutati virtuaal-analoog sünteesi oli Nord Lead (Clavia, Rootsi)

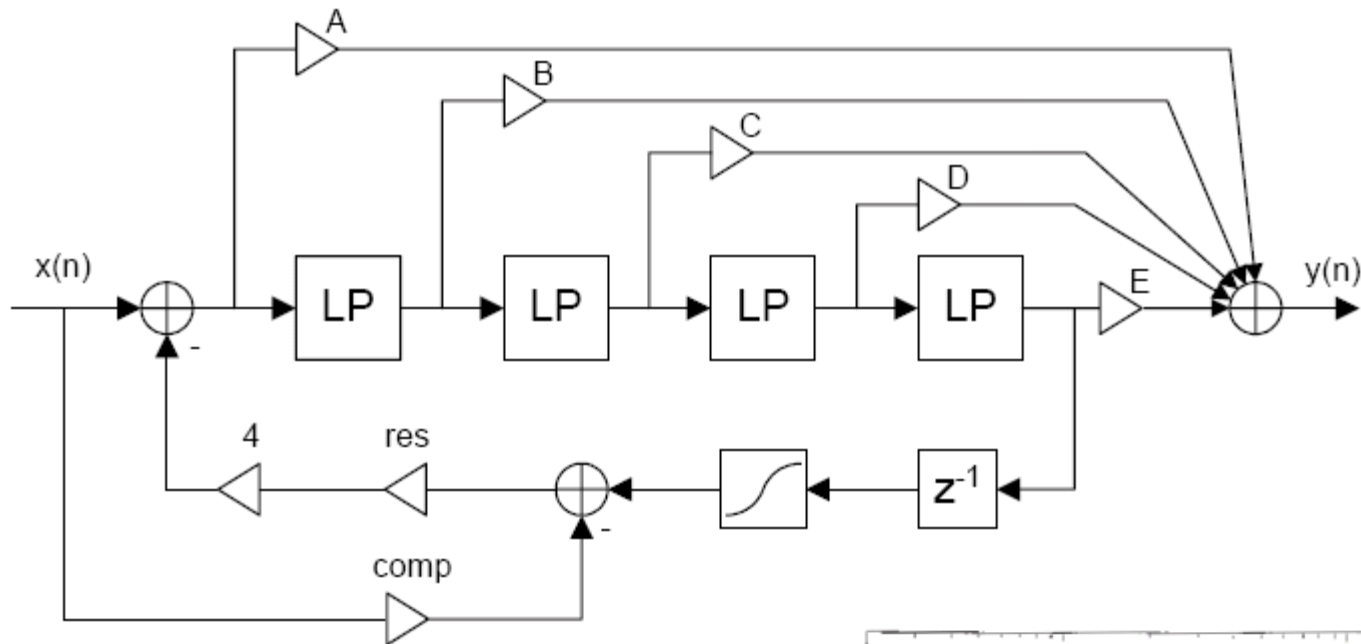
<http://www.nordkeyboards.com/main.asp>

Tänapäeval mh. sellised valmistajad nagu Access, Virus, Novation, Roland kasutavad virtuaal-analoog sünteesi

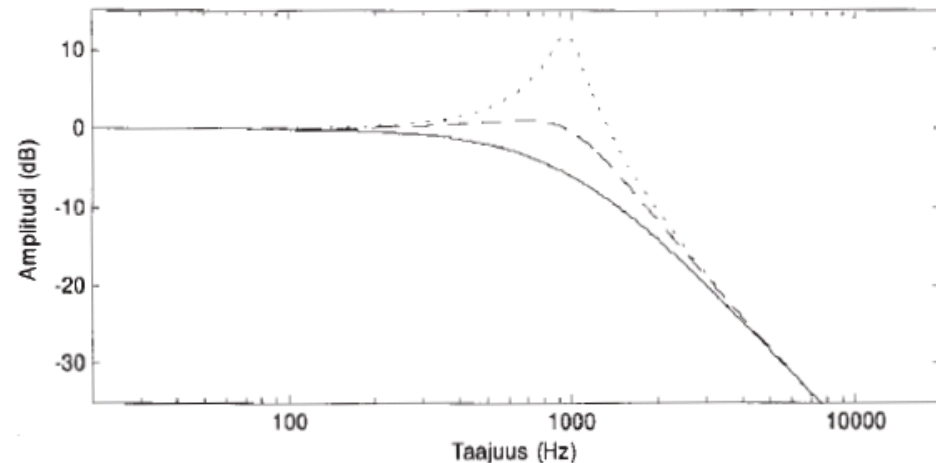


Digitaalne resonantsifilter virtuaal-analoog sünteesi puhul

Joonisel on 4. järku filter.



Madalpääs-variant ->



FM-süntees

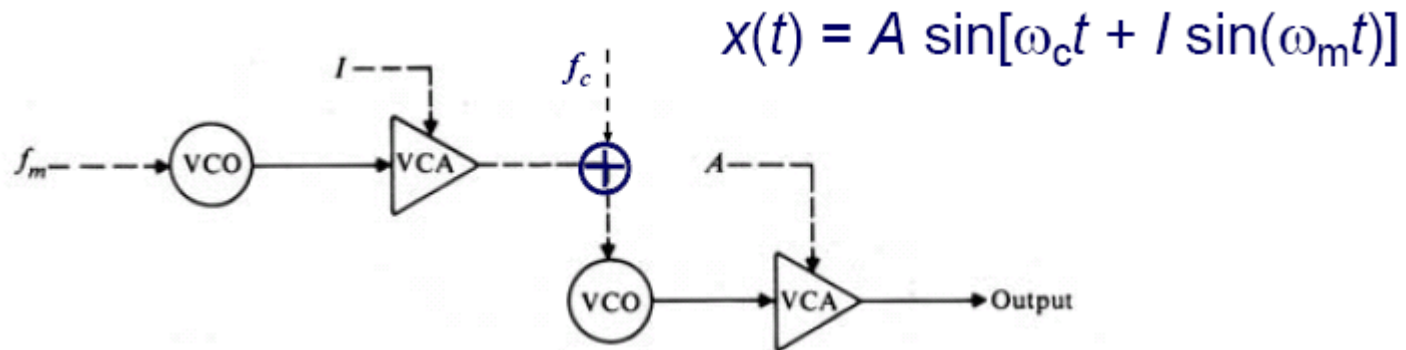
FM – Frequency Modulation (sagedusmodulatsioon)

Leiutas John Chowning 1960:ndatel. Publitseeris ja patenteeris 1970:ndate

lõpul. Meetod jõudis kasutusse Yamaha süntesaatorites 1980:ndatel.

Arvutuslikult lihtne moodus toota palju harmoonilisi.

Joonisel analoogse FM-sünteesi skeem ja valem



VCO – voltage controlled oscillator

VCA – voltage controlled amplifier

Harmoonilised tasaste sagedusvahemike järel mõlemal pool kandesagedust f_c

Harmooniliste hulk sõltub modulatsiooniindeksist I

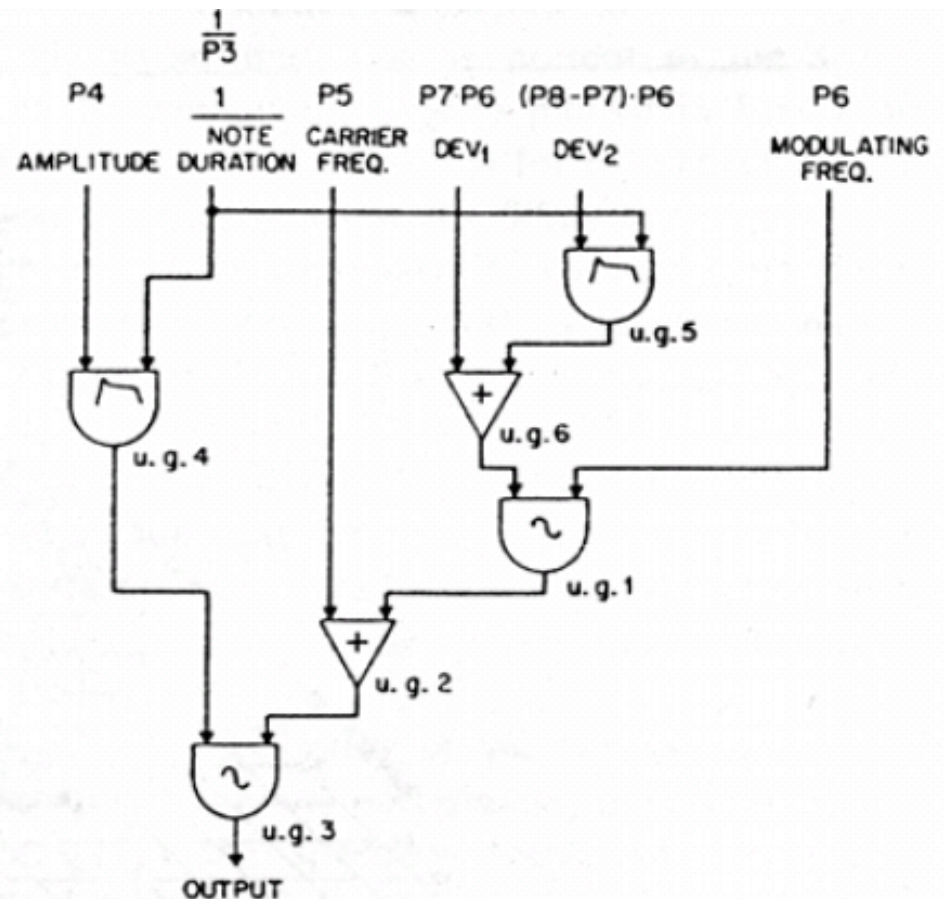
Signaali spekter FM sünteesi puhul

Sageduskomponendid 'peegelduvad' kui nad lähevad Nyqvisti sagedusest kõrgemale

Kui kandesageduse ja modulatsiooni sageduse suhe ei ole täisarv, sünteesitud heli kõlab ebaharmoniliselt. Seda omadust võib kasutada näiteks metalse kõlaga helide tootmiseks.

Chowning'i FM-sünteesi algoritmi ->

Kandesagedus ja modulatsiooniindeks peavad olema ajas muutuvad



FM süntees: Yamaha DX7

Yamaha DX7 oli esimene nõ. Laiatarbe digitaalne süntesaator 1980ndatel.

Sisaldas 6 ostsillaatorit, 10-bitine DA-muundur

Põhines puhtalt FM-sünteesil.

[FM synthesis: Yamaha DX7](#)



Uus helisünteesi meetodite klassifikatsioon (1)

Mitmesugused algoritmid, millega on võimalik toota lihtsalt spektri poolest rikast helisignaali

- FM-süntees, vähendav süntees (alias-filter meetod), Karplus-Strong jne.
- ei põhine füüsilisel mudelil
- üldjuhul on arusaadav, millise meetodiga heli on toodetud

Salvestatud helilõikude töötlemine

- Musique Concrète, sämplimine
- levinuim tehnika kuna mälu on odav
- kasutatakse ka näiteks mobiiltelefonide polüfooniliste helide tootmisel
- näitena [Synful Orchestra: Samples](#)

J. O. Smith, "Viewpoints on the history of digital synthesis," in *Proc. Int. Computer Music Conf. (ICMC'91)*, pp. 1-10, Montreal, Canada, Oct. 1991. A revised version is available at:

<http://www-ccrma.stanford.edu/~jos/kna/>

[J.O.Smith Physical Audio Signal Processing for Virtual Musical Instruments and Audio Effects](#)

Uus helisünteesi meetodite klassifikatsioon (2)

Spektri mudelid

- sinusoid-müra(-transient) mudel, summeeriv süntees
- spekter on heli põhiline omadus kõrva seisukohalt; kõrv on spektrianalüsaator
- põhinevad heli spektri matkimisel, soovitud spektri moodustamisel
- arvutuslikult üldjuhul keerulised, samas võib toota millist heli tahes

Pillide mudelid

- palju erinevaid tehnikaid, näiteks lainejuht-süntees
- püütakse modelleerida akustilise instrumendi helitootmismehhanismi
- põhineb oletusel et toodetud heli kostab loomulikuna kui mudel töötab õieti

Näide laulusünteesist:

<http://www.vocaloid.com/en/lineup/vocaloid4/v4editor.html>

Põhineb sãmplitud signaalil ja selle spektri modelleerimisel sinusoididega

Pillide modelleerimine

Suhteliselt noor helisünteesi valdkond, esimesi tooteid alates 1994

Eesmärgid:

- paremate süntesaatorite loomine
- uute teadmiste saamine akustiliste muusikainstrumentide kohta: millised omaduse mõjutavad heli ja kuidas? milliste heli omaduste järgi instrumente eristada? jne.

Tehnikate liigitus:

- allikas-filter mudelid (modernne vähendav süntees)
- matemaatilisel diferentsvõrrandil põhinevad mudelid
- *mass-spring* (mass-vedru) tehnikad (näiteks CORDIS-ANIMA tarkvara)
- moodisüntees (*Modal synthesis*); näiteks Modalys-tarkvara (IRCAM)
- digitaalne lainejuht-süntees

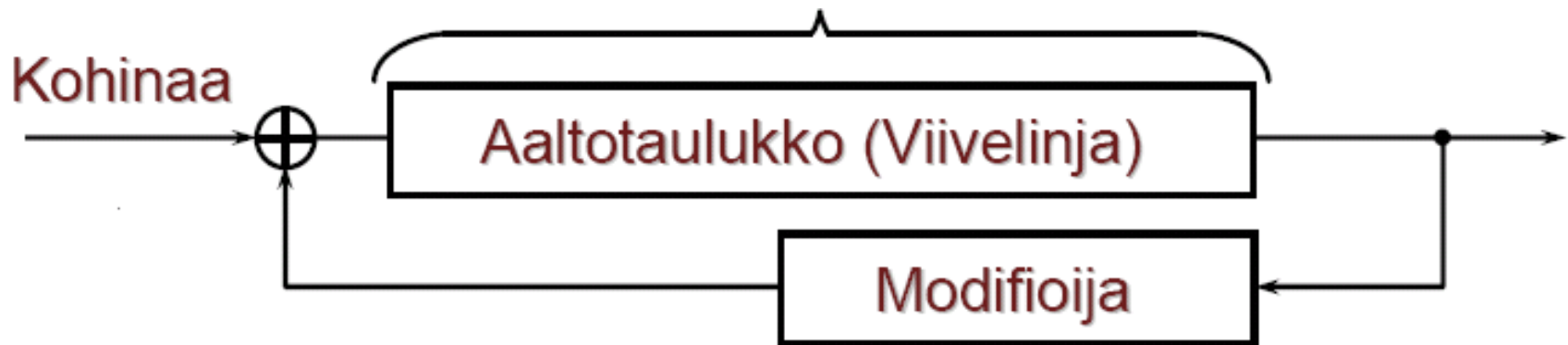
Allikas-filter mudelid

Karplus-Strong algoritm on allikas-filter tüüpi. Selle puudusi on et kõrgemad helid ei 'häälestu'. Heli värv ning sumbumisaeg sõltuvad diskreetimissagedusest.

Probleemi põhjustab see et mida kõrgem sagedus, seda lühem hilistusahel ja seda olulisemaks saab hilistusahela pikkuse 'diskreetsest' iseloomust tulenev sageduse viga.

$$f = f_s / (L + 0,5)$$

Sugugi kõiki põhisagedusi ei ole võimalik toota!



Jaffe-Smith'i mudel

Põhimõtteliselt kujutab endast murdhilistuse kasutamist Karplus-Strong'i algoritmis. Selle abil:

- võib lahendada häälestumisprobleemi
- on võimalik toota glissando

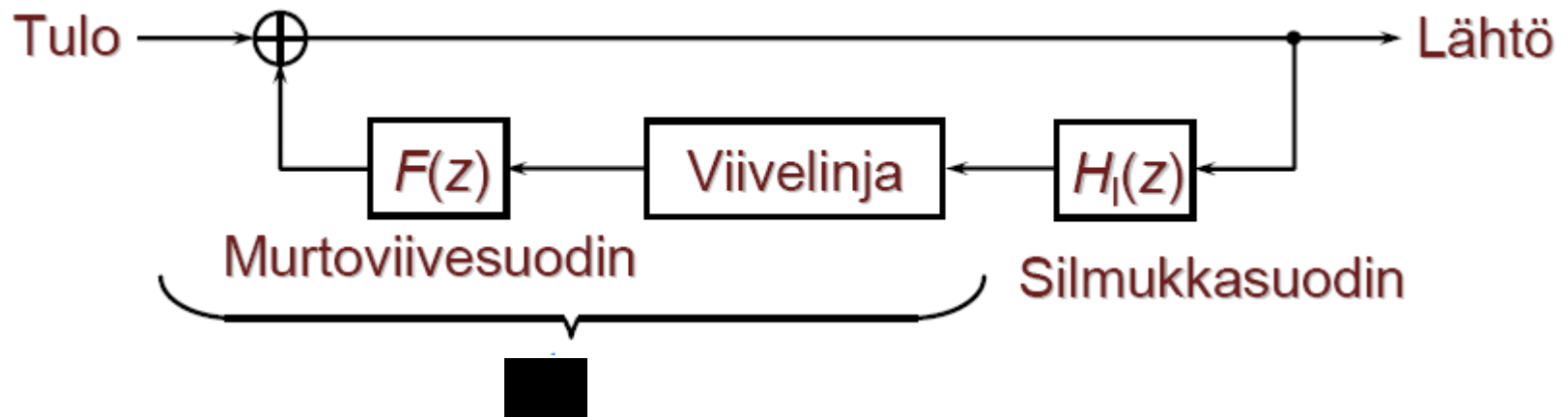
Murdhilistusfiltrina võib kasutada kas *all-pass* filtrit või Lagrange'i interpolaatorit (vähemalt 3. või 4. järku)

Karplus-Strong algoritmi kahe diskreedi keskmist arvutava filtri võime asendada 1. järku IIR filtriga

$$H_1(z) = g(1 + a) / (1 + az^{-1})$$

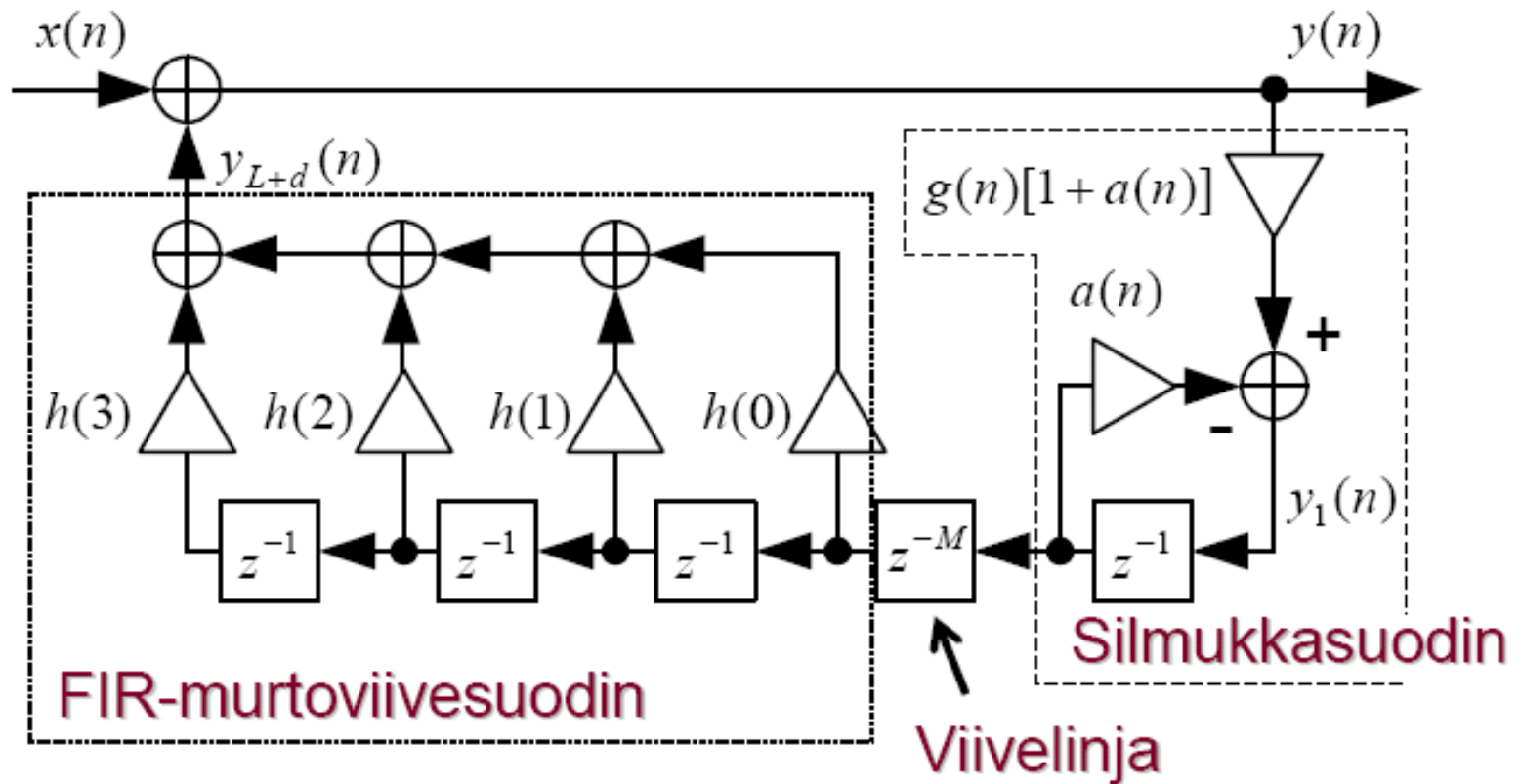
$$g = 1 - \varepsilon \text{ ja } -1 < a < 0$$

Kirjeldatud modifikatsiooni pakkusid välja Jaffe ja Smith:



Jaffe-Smith'i mudel (2)

Joonisel on näide Jaffe-Smith'i mudelist kus heli sumbumiseks on kasutatud 1. järku IIR filtrit ja murdhilistusfiltriks on 3. järku FIR filter



Pillikeele mudeli täiendused

Keele mudelit võib täiendada vastavalt soovile ja modelleeritavale instrumendile. Modelleerimise põhimõte:

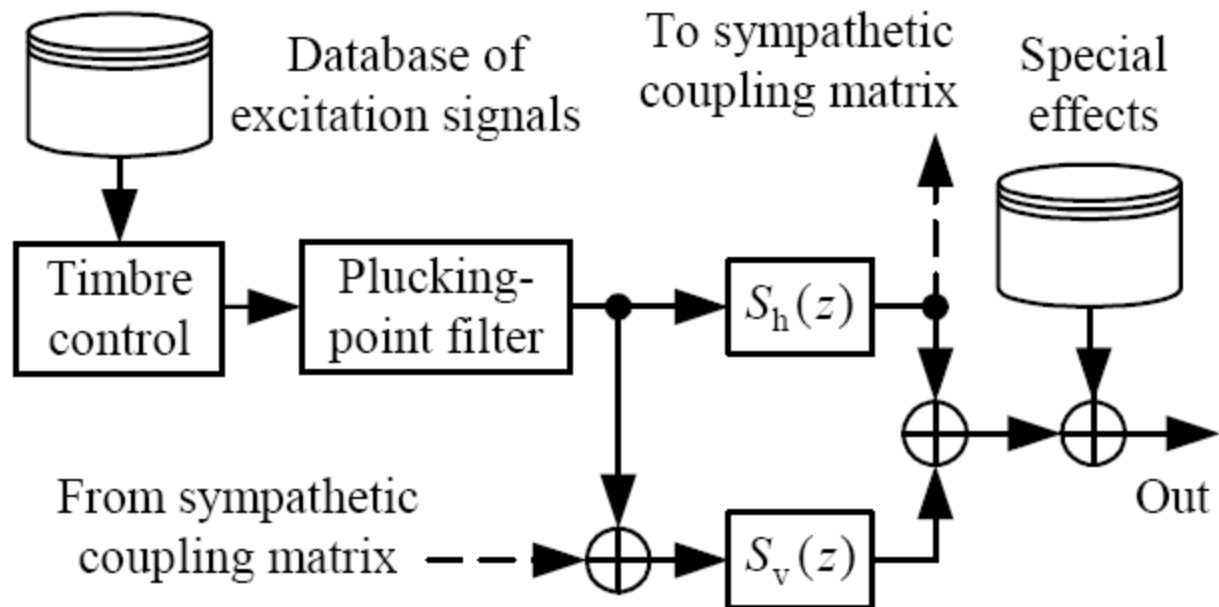
- salvestatakse akustilise keelpilli heli
- analüüsitakse heli Fourier teisenduse abil
- disainitakse keele mudeli filtrid (murdhilistusfilter, sumbuvusfilter)
- eristatakse helist lähtesignaali (keeles 'tõmbamine') näiteks filtreerides keele mudeli pöördfiltriga või kasutades modelleerimist sinusoididega

Joonisel mudel kus on kaks paralleelset keelemudelit. Lähtesignaali ja esitusefektid (näiteks klahvi vabastamine) on salvestatud eelnevalt andmebaasi

<http://www.acoustics.hut.fi/demos/>

Klaverimodelleerimis-tarkvara:

<http://www.pianoteq.com>



Tiibklaveri (*Grand piano*) mudel: *Beating*

Näiteid:

- Üks ülemheli *beating* – efektiga
'a single partial with a beating effect. The partial index varies from 0 (no beating) to 10 in an increasing manner;
Fundamental frequency = 65.4 Hz (C2)
Beating frequency = 1 Hz
 - 0...5 🗣️
 - 6...10 🗣️
- Ühest üheksa ülemhelini *beating* – efektiga
'the number of beating partials is increased from 0 to 9'
Fundamental frequency = 61.7 Hz (B1)
 - 0...9 🗣️
- Ilma *beating* – efektita vs. *beating* – efektiga
'two 15s samples without and with beating. The latter sample has 9 beating partials'
 - 2 näidet 🗣️



[HUT Acoustics Group: Piano beating examples](#)

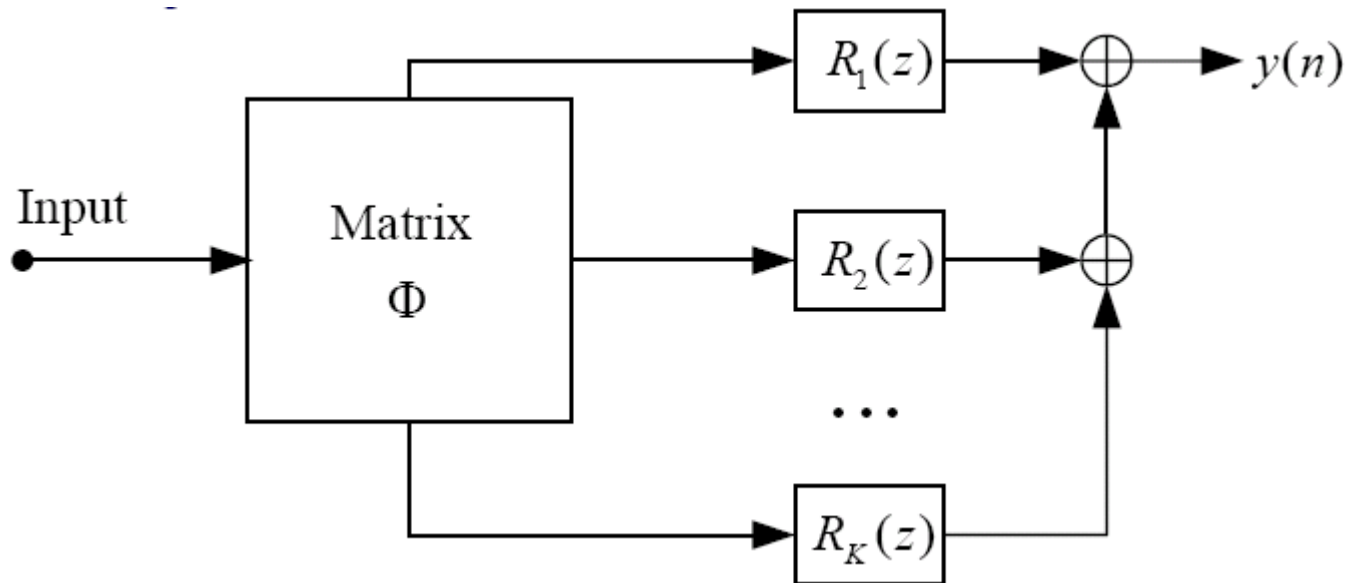
Beating partial indices and frequencies (in order of appearance):

Partial index	Beating frequency
3	0.63 Hz
4	0.67 Hz
5	0.83 Hz
8	0.50 Hz
11	0.80 Hz
12	0.95 Hz
13	0.77 Hz
14	0.20 Hz
16	1.05 Hz

Moodisüntees

Kõik pilli võnkumismoodid modelleritakse eraldi.

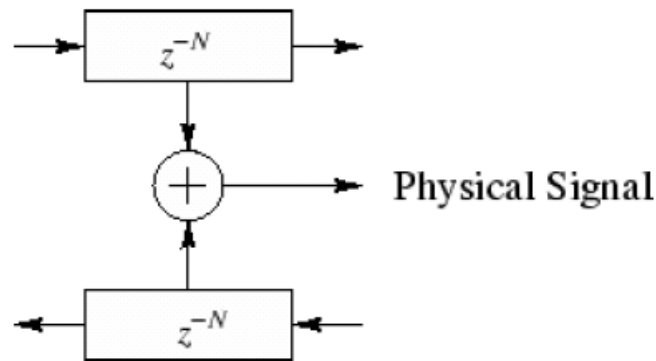
Kasutatakse hulka teist järku digitaalfiltreid



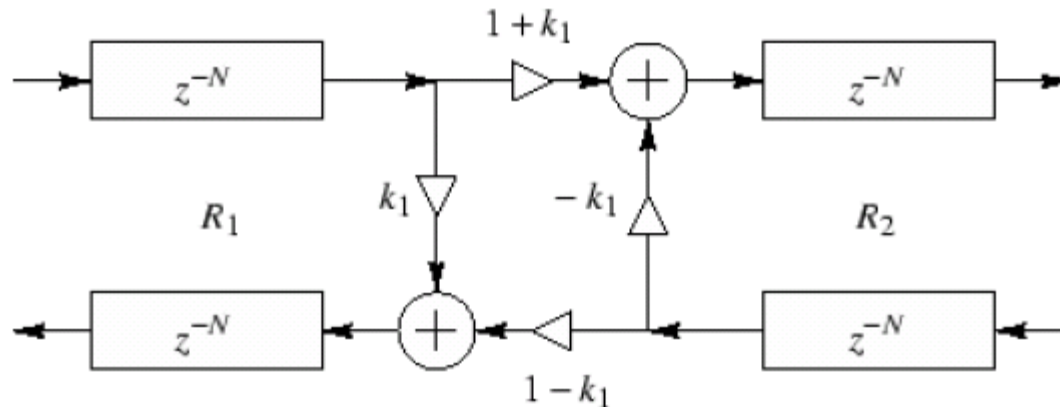
Lainejuht-süntees

Digitaalne lainejuht kujutab endast kahe suunalist hilistusahelat.

Väljundsignaaliks on nende hilistusahelate signaalide summa samas punktis:



Mudel mis võtab arvesse ka peegeldunud laine:



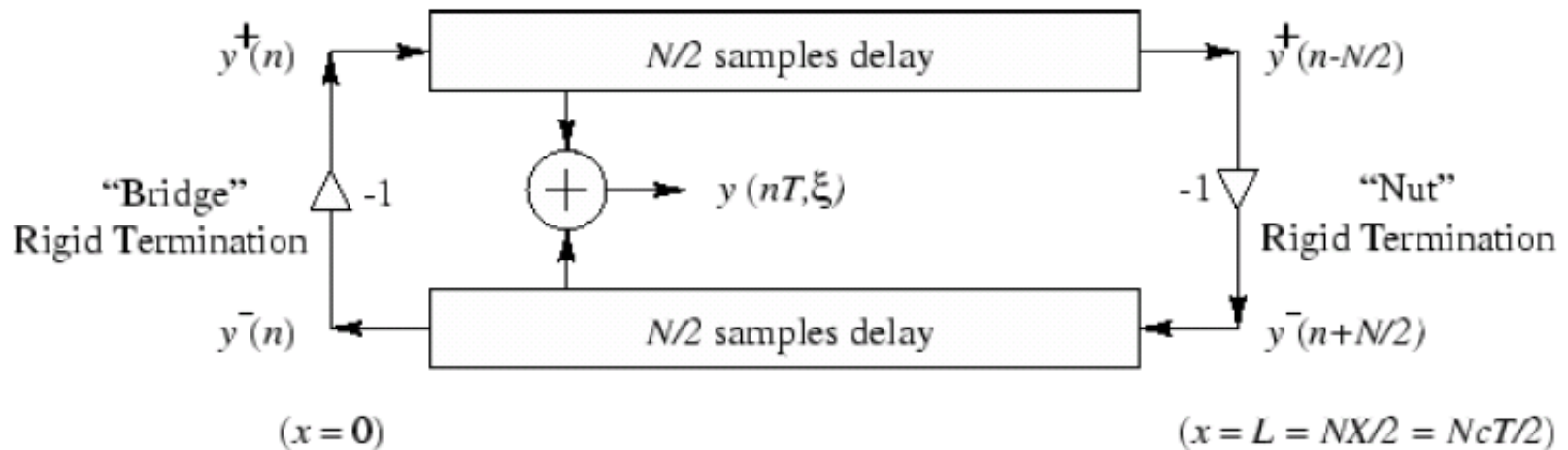
$$k_1 = \frac{R_2 - R_1}{R_2 + R_1}$$

Lainejuht-süntees (2)

Kadudeta lainejuht-mudel:

- peegelduskordaja mõlemas otsas -1
- teoreetiline mudel

Väljundi võib arvutada kas ühes või mitmes punktis

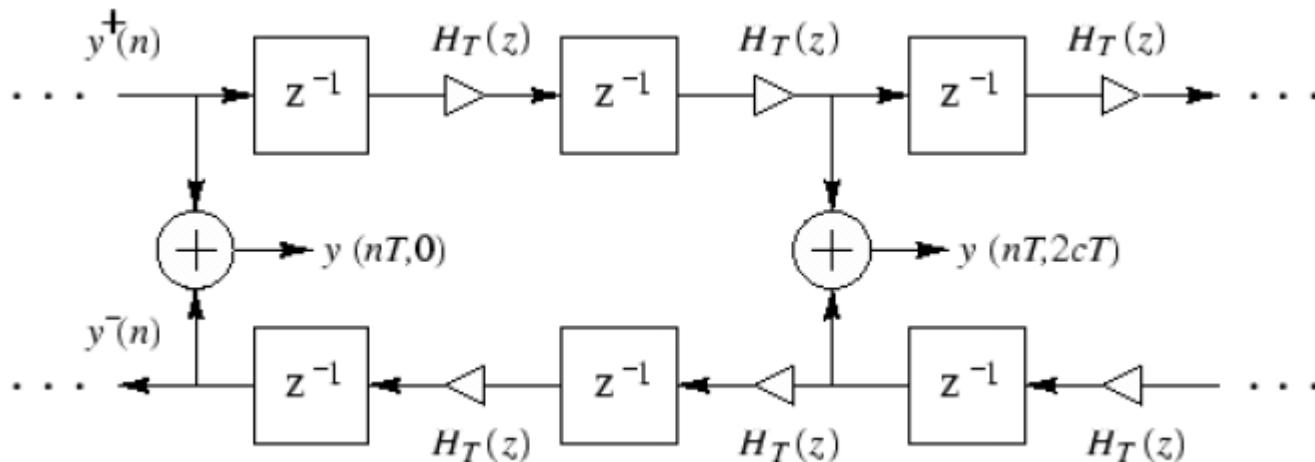


Kadudega lainejuht

Praktikas võnkumine sumbub. Energia kaob kohtades kus keel kinnitub pilli külge. Samuti mõjutab keele sisene ja õhu hõõrdumine

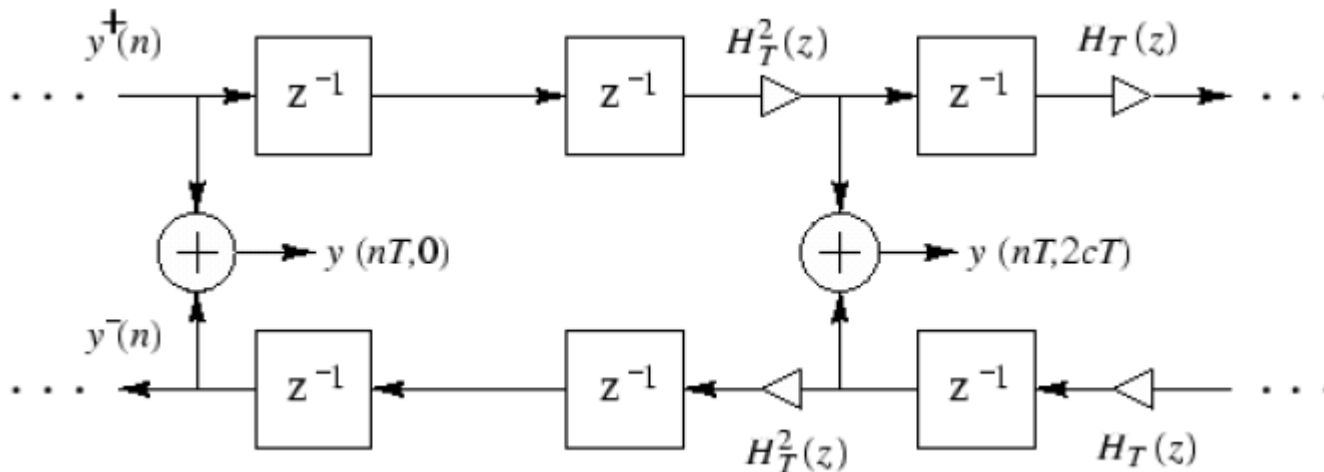
Sumbumine sõltub sagedusest. Looduslikes süsteemides kõrged sagedused sumbuvad rohkem kui madalad sagedused

Kadusid võib modelleerida hilistusega ühendatud madalpääsfiltri abil. Praktikas üsna lauge karakteristikuga

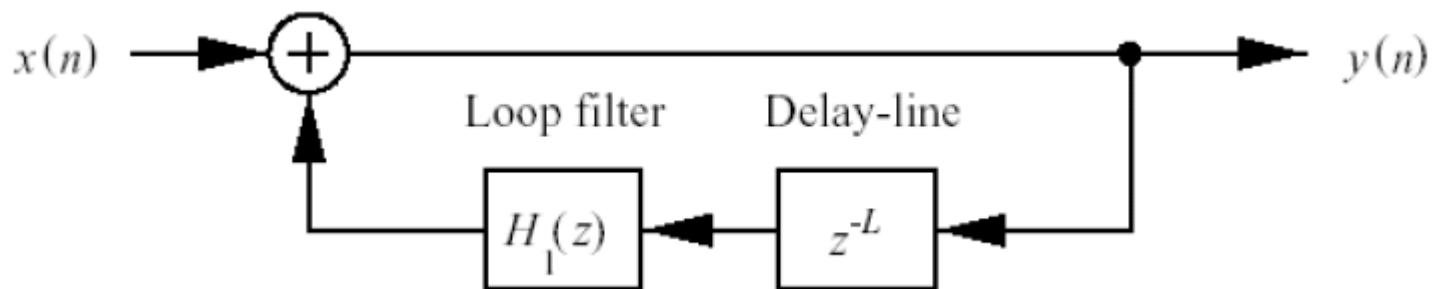


Filtrite ühendamine

Sumbuvusfiltrid võib ühendada nii et iga väljundi kohta on üks filter



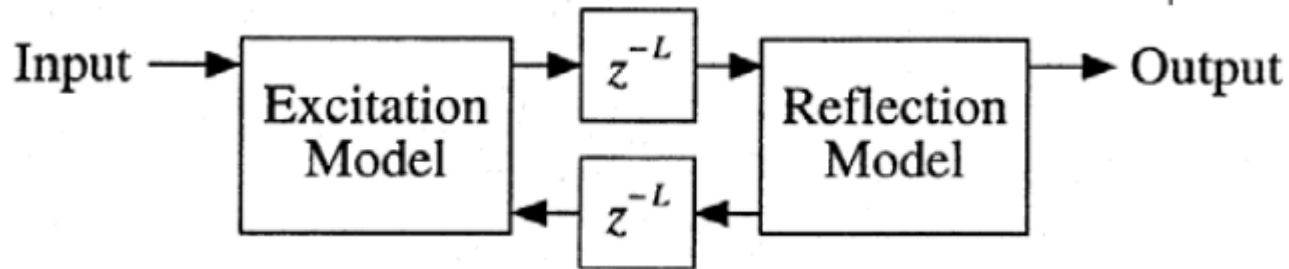
Kui väljundeid ja sisendeid on üks, võib kogu mudeli joonistada:



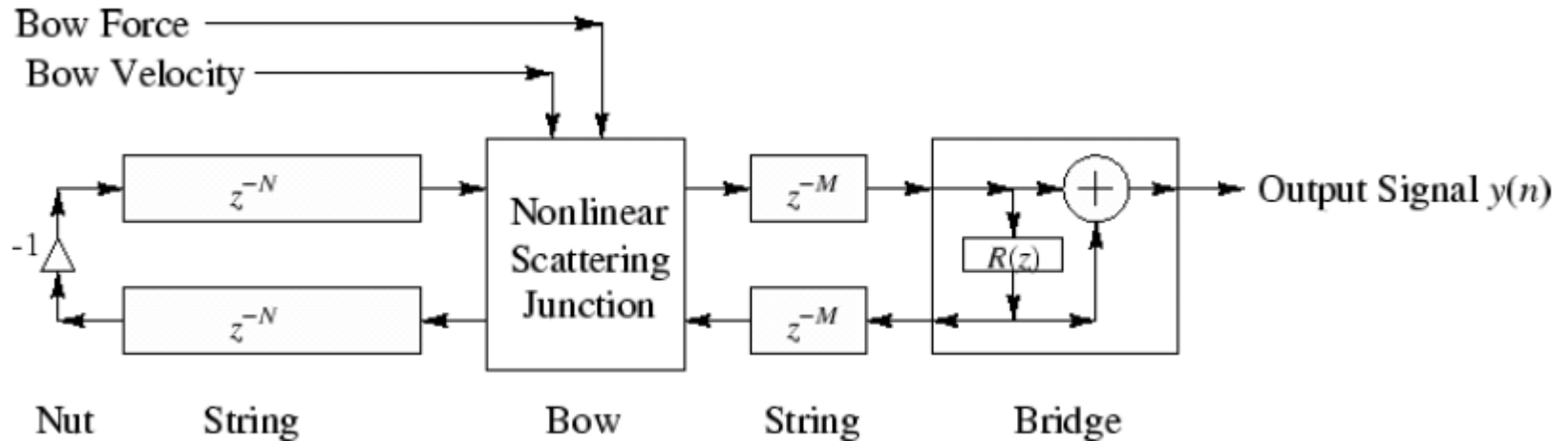
See skeem meenutab Karplus-Strong algoritmi; filtrid ja nende otstarve on erinevad

Puhkpilli ja keelpilli mudelid

Puhkpilli mudelis on lineaarne 'toru' mudel ning ebalineaarne lähtesignaali (*excitation*) ühendumismudel:



Keelpilli mudelis on lineaarne pillikeele mudel ja ebalineaarne poogna ühendumismudel:



[Examples from Smith](#)

Lainejuhil põhinevad süntesaatorid

Lainejuhil põhinevad näiteks:

- Yamaha VL-1 ja VL-7, 1994

<http://www.patchmanmusic.com/yamahaVL1.html>



- Korg Prophecy, 1995

